Face Hallucination via Deep Neural Networks

Xin Yu

A thesis submitted for the degree of
Doctor of Philosophy
The Australian National University

January 2019

Declaration

I hereby declare that this thesis is my own original work. This thesis has not been previously submitted by me in whole or part for a degree or diploma in any university or other tertiary education institution. The content of this thesis is mainly based on the publications during my PhD as listed below:

1. Xin Yu, Fatih Porikli: Ultra-Resolving Face Images by Discriminative Generative Networks. In *European Conference on Computer Vision (ECCV)*, 318-333, 2016.
2. Xin Yu, Fatih Porikli: Face Hallucination with Tiny Unaligned Images by Transformative Discriminative Neural Networks. In *The Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*, 4327-4333, 2017.
3. Xin Yu, Fatih Porikli: Hallucinating Very Low-Resolution Unaligned and Noisy Face Images by Transformative Discriminative Autoencoders. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3760-3768, 2017.
4. Xin Yu, Basura Fernando, Richard Hartley, Fatih Porikli: Super-Resolving Very Low-Resolution Face Images with Supplementary Attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 908-917, 2018.
5. Xin Yu, Fatih Porikli: Imagining the Unimaginable Faces by Deconvolutional Networks. *IEEE Transactions on Image Processing*, 27(6): 2747-2761, 2018.
6. Xin Yu, Basura Fernando, Bernard Ghanem, Fatih Porikli, Richard Hartley: Face Super-Resolution Guided by Facial Component Heatmaps. In *European Conference on Computer Vision (ECCV)*, 217-233, 2018.
7. Fatemeh Shiri, Xin Yu, Piotr Koniusz, Fatih Porikli: Face Destylization. In *Digital Image Computing: Techniques and Applications (DICTA)*, 1-8, 2017.
8. Fatemeh Shiri, Xin Yu, Fatih Porikli, Richard Hartley, Piotr Koniusz: Identity-Preserving Face Recovery from Portraits. In *IEEE Winter Conference on Application of Computer Vision (WACV)*, 102-111, 2018.
9. Fatemeh Shiri, Xin Yu, Richard Hartley, Fatih Porikli, Piotr Koniusz: Recovering Faces from Portraits with Auxiliary Facial Attributes. Accepted by *IEEE Winter Conference on Application of Computer Vision (WACV), 2019*.
10. Xin Yu, Basura Fernando, Richard Hartley, Fatih Porikli: Semantic Face Hallucination: Super-Resolving Very Low-Resolution Face Images with Supplementary Attributes. Submitted to *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

11. Xin Yu, Basura Fernando, Fatih Porikli, Richard Hartley: Hallucinating Unaligned Face Images by Multiscale Transformative Discriminative Networks. Submitted to *International Journal of Computer Vision*.

12. Xin Yu, Fatemeh Shiri, Bernard Ghanem, Fatih Porikli: Can We See More? Joint Frontalization and Hallucination of Non-frontal Tiny Faces by Transformative Adversarial Neural Networks. Submitted to *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Xin Yu
7 January 2019

to my family

Acknowledgments

This thesis would never have been finished without the support of the ANU PhD Scholarship and the ANU HDR Fee Remission Merit Scholarship which were awarded to me in 2016, as well as the Australian Research Council's Discovery Projects funding scheme (project DP150104645). I would like to thank ANU for providing such a beautiful environment where I can focus on my research, as well as for supporting me to attend international conferences which not only broadened my horizons but also encouraged me to aim for the highest.

I really want to thank my supervisors Prof. Richard Hartley, Prof. Fatih Porikli and Dr. Basura Fernando, for guiding me with their erudition and being patient with me. They taught me how to do research and gave great freedom and space to choose the research topics. Without their encouragement and support, I would not have achieved confidence and joys on academic research. I remember that whenever I had a new idea, I always had the support from all my supervisors, which made me believe I can work it out. I feel so lucky to be looked after so well by all my supervisors from idea discussion to paper submission. Despite they are already prestigious, they are extremely nice and always make time for me to discuss my ideas and papers. They set a good example of what an excellent scholar is to me. Furthermore, I also owe special thanks to Prof. Fatih Porikli. Without his DP funding, I may not have the opportunity to receive such great education and pursue my dream of being an excellent researcher.

I would love to thank my dear friend Morgan Hitchcock for proofreading all my papers. He helps me to improve my spoken and written English, and has always been the first reader of all my papers. He is generous with his time to help me and also let me stay with him when I need to find some temporary place in Canberra. I also want to thank my friends at ANU. With your company, the journey never seems long and solitary.

Last but not least, I want to thank my family. Even though they are all in China, they tried their best to support me financially and mentally. I am so blessed to be my parents' son and my grandparents' grandson.

Abstract

Face images can provide vital clues for identity recognition and expression analysis. However, those tasks require face images have sufficient resolutions and clear details. Due to different imaging conditions, face images might be captured in very low resolutions. Obtaining high-resolution (HR) face images plays an important role for the following face analysis tasks. In this thesis, we tackle the face super-resolution problem, also known as face hallucination, and propose our methods to upsample very low-resolution face images as well as recover fine details of the deteriorated faces.

We firstly address aligned low-resolution (LR) face images (*i.e.*, 16×16 pixels) by designing a discriminative generative network, named URDGN. URDGN is composed of two networks: a generative model and a discriminative model. We introduce a pixel-wise ℓ_2 regularization term to the generative model and exploit the feedback of the discriminative network to make the upsampled face images resemble real ones. In our framework, the discriminative network learns the essential constituent parts of the faces and the generative network blends these parts in the most accurate fashion to the input image. Regarding the difficulty of training two individual networks, we also present a single network which consists of deconvolutional and convolutional layers to upsample faces by a large upscaling factor of $8\times$. These two methods only require frontal and ordinary aligned images in training. Therefore, our methods can super-resolve a wide range of LR images directly regardless of pose and facial expression variations.

State-of-the-art face hallucination methods rely heavily on accurate alignment of LR faces before upsampling them. Misalignments often lead to deficient results and unnatural artifacts for large upscaling factors. To overcome this challenge, we present an end-to-end transformative discriminative neural network (TDN) devised for super-resolving unaligned tiny face images. TDN embeds spatial transformation layers to enforce local receptive fields to line-up with similar spatial supports. In this manner, unaligned faces are automatically aligned in the upsampling procedure. Moreover, previous works often assume LR face images are noise-free. When input images are contaminated by noise, their super-resolution performance will degrade dramatically. To upsample noisy unaligned LR face images, we propose decoder-encoder-decoder networks. A transformative discriminative decoder network is employed to upsample and denoise LR inputs simultaneously. Then we project the intermediate HR faces to aligned and noise-free LR faces by a transformative encoder network. Finally, high-quality hallucinated HR images are generated by a second decoder.

When the resolutions of LR input images vary, previous deep neural network based face hallucination methods require input images at a fixed resolution. Down-

sampling LR input faces to a required resolution will lose high-frequency information of the original input images. This may lead to suboptimal super-resolution performance for the state-of-the-art face hallucination networks. We present an end-to-end multiscale transformative discriminative neural network (MTDN) to super-resolve unaligned LR face images in different resolutions ranging from 16×16 to 32×32 pixels in a unified framework.

Previous face hallucination methods do not account for facial structure and thus suffer from degradation due to large pose variations and misalignments. We propose a method that explicitly incorporates structural information of faces into the face super-resolution process by using a multi-task convolutional neural network (CNN). Our CNN has two branches: one branch for super-resolving face images and the other one for predicting salient regions of a face coined *facial component heatmaps*. These heatmaps encourage the upsampling stream to generate super-resolved faces in large poses with higher-quality details. Our method not only uses low-level information (*i.e.*, intensity similarity), but also employs middle-level information (*i.e.*, face structure) to super-resolve facial components in LR face images. Therefore, our method is able to super-resolve very small unaligned face images while preserving face structure.

Since an LR input patch may correspond to many HR candidate patches, this ambiguity may lead to distorted HR facial details and inaccurate facial attributes, such as gender reversal and age rejuvenation. We observe that an LR input face mainly contains low-frequency facial components of its HR version while its residual face image, defined as the difference between the HR ground-truth and interpolated LR images, contains the missing high-frequency facial details. We demonstrate that supplementing residual images or feature maps with additional facial attribute information can significantly reduce the ambiguity in face super-resolution. To explore this idea, we develop an attribute-embedded upsampling network. The upsampling network is composed of an autoencoder with skip-connections, which incorporates facial attribute vectors into the residual features of LR inputs at the bottleneck of the autoencoder, and deconvolutional layers used for upsampling. In this manner, our method is able to super-resolve LR faces by a large upscaling factor while reducing the uncertainty of one-to-many mappings remarkably.

We further push the boundaries of hallucinating a tiny, non-frontal face image to understand how much of this is possible by leveraging the availability of large datasets and deep networks. To this end, we introduce a novel transformative adversarial neural network (TANN) to jointly frontalize very LR out-of-plane rotated face images (including profile views) and aggressively super-resolve them by 8×, regardless of their original poses and without using any 3D information. TANN is composed of two components: a transformative upsampling network, which first projects/encodes side-view LR faces close to the latent representation of their corresponding frontal ones and then upsamples the latent representation, and a discriminative network that enforces the generated high-resolution frontal faces to lie on the same manifold as real frontal face images.

Besides super-resolving an HR face image from its LR version, this thesis also

addresses the task of restoring realistic faces from stylized portrait images, which can also be regarded as a type of face hallucination. We develop a style removal network composed of convolutional, fully-connected and deconvolutional layers. The convolutional layers are designed to extract facial components from stylized face images. Consecutively, the fully-connected layer transfers the extracted feature maps of stylized images into the corresponding feature maps of real faces and then the deconvolutional layers generate real faces from the transferred feature maps. To enforce the destylized faces to be similar to authentic face images, we employ a discriminative network, which consists of convolutional and fully connected layers. Furthermore, by constraining feature-wise similarity between the recovered faces and the ground-truth ones, we can achieve realistic faces much closer to their ground-truth in terms of appearance and identity similarity. Since the facial details are distorted in the stylized portraits, such as skin and hair colors, we embed facial attributes into the destylizing procedure to recover face images faithful to the ground-truth ones.

In summary, this thesis exploits deep neural networks to super-resolve HR face images from their LR counterparts in different challenging scenarios as well as to restore realistic face images from stylized portrait images. Our extensive experimental results demonstrate our proposed methods outperform the state-of-the-art.

Keywords: Face super-resolution, face hallucination, face destylization

Contents

List of Figures

List of Tables

Chapter 1

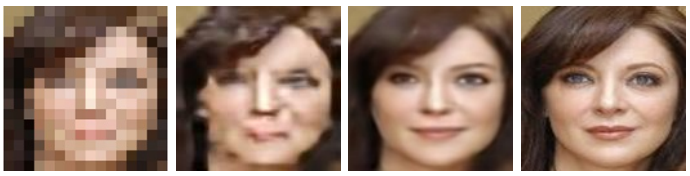
Introduction

1.1 Background

Face images arguably carry the most interesting and valuable visual information and can be obtained in a non-intrusive manner. Additionally, for many applications from content enhancement to forensics, face images require significant magnification. Obtaining high-resolution (HR) face images will facilitate the other face analysis tasks. However, due to camera settings and the large distances between objects and cameras, the resolutions of face images might be very small, (*e.g.*, in typical surveillance videos). There is little information that can be inferred from the captured face images, as visible in Fig. 1.1. Very low-resolution (LR) face images not only degrade the performance of the face recognition systems but also impede human interpretation.

When face images are imperceptibly small, their resolutions have to be increased by a large upscaling factor. However, conventional super-resolution (SR) methods are mostly limited up to $2 \sim 4\times$ upscaling factors. As reported by Yang et al. [2014], when the upscaling factor increases to $8\times$, the performance of state-of-the-art super-resolution techniques decreases rapidly, rendering them unsuitable for this task. This challenge motivates the reconstruction of HR face images from given LR counterparts, known as face hallucination, and has attracted increasing interest in recent years.

Existing face hallucination methods achieve exciting super-resolution results when accurate facial features and landmarks can be found in LR images, suitably similar HR images of the same person are included in the support dataset, or the exemplar HR face images are densely aligned [Tappen and Liu, 2012; Yang et al., 2013; Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Yang et al., 2010]. For instance, landmark based methods [Tappen and Liu, 2012; Yang et al., 2013] first localize facial components in the LR face images, and then transfer the similar HR facial components extracted from the exemplar dataset to the input LR faces. However, when the input image resolution becomes smaller, landmark based methods fail gravely because of erroneous landmark localization. In other words, their performance highly depends on the input image sizes. Another stream of face hallucination methods employs the similarity of the subspace projection of LR and HR images to reconstruct HR face images [Baker and Kanade, 2000; Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007]. Due to the variations of poses, lighting



(a) LR (b) Bicubic (c) Ours (d) HR

Figure 1.1: Illustration of face hallucination. (a) The input 16×16 LR image. (b) Bicubic interpolation of (a). (c) Our hallucinated result. (b) The original 128×128 HR image.

(a) Stylized (b) Recovered (c) Original

Figure 1.2: Illustration of face destylization. (a) Stylized input portrait. (b) Recovered realistic face image from (a). (c) Original ground-truth face image.

and expressions existing in LR face images, the appearances of the input LR face images may be different from the HR images in the dataset. Subspace based methods will degrade dramatically and produce ghosting artifacts in the outputs. To mitigate the ghosting artifacts caused by the pose variations in the input LR faces and the HR exemplar dataset, position-patch based methods are proposed [Ma et al., 2010; Yang et al., 2010]. However, as the magnification factor increases, the blocky artifacts between neighboring patches appear and lead to unnatural super-resolved HR faces.

As mentioned above, state-of-the-art face hallucination methods would suffer from obvious artifacts when they are applied to super-resolve very low-resolution faces undergoing misalignments or even contaminated by noise. Therefore, this thesis intends to better explore the structure of faces and appearance similarities between individuals and then recovers HR realistic face images by leveraging the emergence of large-scale face datasets [Huang et al., 2007; Liu et al., 2015] as well as deep neural networks.

In this thesis, we also investigate the problem of reverting an artistic portrait back to its photo-realistic version. Although applying artistic styles to existing photographs has attracted much attention in both academia and industry with several interesting applications, hallucinating a photo-realistic face image from an artistic portrait, dubbed face destylization, has not been studied thoroughly. As seen in Fig. 1.2, revealing the latent real faces can provide essential information for human perception, computer analysis and photo-realistic multimedia content editing. Since facial details and expressions in stylized portraits often undergo severe distortions

and become contaminated with artifacts, such as the changes of profile edges and colors, recovering a photo-realistic face image from its stylized version is very challenging.

Overall, this thesis presents face hallucination methods to upsample LR face images in different cases, including unaligned LR faces, inputs contaminated by noise, LR faces undergoing large pose variations. Furthermore, we also try to push the boundary of face hallucination methods, *i.e.*super-resolving and frontalizing LR faces simultaneously. Inspired by the ideas of face super-resolution, we also hallucinate realistic face images from abstract portrait images, thus providing a solution to reveal the real identity information in the portraits.

In this chapter, we first review the related works on face super-resolution as well as style transfer. Then, we outline our proposed solutions to the challenging problems in the field of face super-resolution and stylized portrait recovery. We also introduce the organization of this thesis and the relationship between each chapter. Since the format of this thesis is *"thesis by compilation"*, each chapter is composed of a published paper or manuscript written during my PhD period.

1.2 Related Works

Over the past decades, image super-resolution methods have been proposed to magnify an LR image to its HR version that comprises authentic high-frequency details. Super-resolution can be basically classified into two categories: generic super-resolution methods and class-specific super-resolution methods. When upsampling LR images, generic methods employ priors that ubiquitously exist in natural images without considering any image class information. Class-specific methods, also called face hallucination [Baker and Kanade, 2000] if the class is face, aim to exploit statistical information of objects in a certain class. Thus, they usually attain better results than generic methods when super-resolving images of a known class. In this thesis, we not only super-resolve LR faces but also frontalize input LR faces. Therefore, we also review the most related face frontalization methods. Due to the inherently under-determined nature of super-resolution, an LR face may correspond to many HR candidate faces, especially when the upscaling factor becomes larger. We embed facial attribute information into the super-resolution procedure to achieve more accurate hallucinated faces by a conditional generative adversarial network. Thus, we review the generative adversarial network [Goodfellow et al., 2014] and its variants related to our work. Furthermore, since we also address the problem of recovering realistic face images from stylized ones, state-of-the-art image style transfer methods are also introduced in this section.

1.2.1 Generic Super-resolution Methods

In general, there are three categories of generic super-resolution approaches: interpolation based techniques, image statistics based schemes [Peleg and Elad, 2014; Yang and Yang, 2013] and example/patch based methods [Freeman et al., 2002;

Hong Chang et al., 2004; Glasner et al., 2009; Yang et al., 2010; Schulter and Leistner, 2015; Huang et al., 2015]. Interpolation based techniques such as bilinear and bicubic upsampling are computationally efficient. However, they fail to establish high-frequency details since they generate overly smooth edges as the upscaling factor increases. Image statistics based schemes employ image priors to reconstruct HR images with sharper edges, but they are still limited to smaller scaling factors [Lin and Shum, 2006].

Example based methods have the potential to break this limitation. They can be further classified into two groups: internal and external example methods depending on how the reference samples are derived. The first group of methods [Glasner et al., 2009; Freedman and Fattal, 2010; Singh et al., 2014; Huang et al., 2015] exploit self-similarity of patches in the input image. Alternatively, several methods [Freeman et al., 2002; Hong Chang et al., 2004; Yang et al., 2010] aim to learn mappings between LR and HR patches from external reference datasets, and then utilize the learned correspondences to upsample LR images. Nevertheless, when the input image size is very small, it is difficult for internal example based methods to find similar patches across different scales. When the scaling factor is large, it is hard for external example based methods to determine the correct correspondences between LR and HR patches because many different HR patches can correspond to a single LR patch, which induces artifacts at intensity edges.

Recently, many generic super-resolution methods based on deep neural networks have been proposed [Dong et al., 2016a,b; Bruna et al., 2016; Kim et al., 2016a,b; Mao et al., 2016; Shi et al., 2016; Ledig et al., 2017]. For instance, SRCNN [Dong et al., 2016a] apply cascaded convolutional layers to obtain a mapping function between LR and HR patches from a large-scale dataset, while Kim et al. [2016a] learn to upsample the residuals between the HR and interpolated LR patches. To improve the performance of super-resolution without introducing extra parameters of the networks, Kim et al. [2016b] employ recursive convolutional layers to increase the depth of the convolutional layers. Mao et al. [2016] apply symmetric-skip connections between convolutional layers and deconvolutional layers to pass information to the latter layers, thus mitigating the difficulty of training their very deep network. Shi et al. [2016] employ convolutional layers to extract LR features and then rearrange the LR feature maps into HR images by a sub-pixel convolutional layer, which can be considered as a variant of deconvolutional layers. Dong et al. [2016b] use convolutional and deconvolutional layers with smaller filter sizes to speed up SRCNN [Dong et al., 2016a]. Ledig et al. [2017] exploit an adversarial loss and a perceptual loss [Johnson et al., 2016] to obtain more realistic upsampled results. Bruna et al. [2016] extract statistical priors using a convolutional neural network (CNN) to regularize the super-resolution process. Since these generic SR methods based on neural networks do not consider class-specific priors, they cannot achieve high performance when they are employed for super-resolving faces.

1.2.2 Face Hallucination Methods

Unlike generic methods, class-specific super-resolution methods [Baker and Kanade, 2000; Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Ma et al., 2010; Tappen and Liu, 2012; Yang et al., 2013; Zhou and Fan, 2015; Wang et al., 2014; Kolouri and Rohde, 2015; Jin and Bouganis, 2015; Zhu et al., 2016b] further exploit the statistical information in the image categories, thus leading to better performances.

In one of the earlier works, Baker and Kanade [2000] transfer high-frequency details from a face dataset by building the relationships between LR and HR patches. Their method can generate face images with richer details. However, due to the possible inconsistency of the transferred HR patches, their method tends to produce artifacts. Wang and Tang [2005] employ constraints on both LR and HR images, and then hallucinate HR face images by an eigen-transformation. Although it is able to magnify LR images by a large scaling factor, the output HR images suffer from ghosting artifacts as a result of using a subspace based on a holistic model. Similarly, Liu et al. [2007] enforce linear constraints for HR face images using a subspace learned from the training set via Principle Component Analysis (PCA), and a patch-based Markov Random Field (MRF) is proposed to reconstruct the high-frequency details in the HR face images. This method works when the images are precisely aligned at fixed poses and expressions. In other cases, the results usually contain ghosting artifacts due to PCA based holistic appearance model. To mitigate artifacts, a blind bilateral filtering is used as a post-processing step in [Liu et al., 2007]. Kolouri and Rohde [2015] use optimal transport in combination with subspace learning to morph an HR image from the LR input. Since the subspace based face hallucination methods require the HR images in the reference dataset to be precisely aligned and the LR test image to have the same pose and facial expression as the reference ones, they are overly sensitive to the misalignments of LR images. In particular, methods that depend on PCA based holistic appearance models suffer from ghosting artifacts.

Considering pose and expression variations in both LR and HR face images, it is difficult to hallucinate HR faces by employing only one global appearance model. Thus, local part based methods are proposed to super-resolve individual facial regions separately. They reconstruct the HR counterparts of LR inputs based on either reference patches or facial components in the training dataset. Ma et al. [2010] construct a super-resolved HR patch by multiple reference HR patches at the corresponding spatial position. Yang et al. [2010] and Li et al. [2014] model the local structures of faces as a sparse representation problem. Jin and Bouganis [2015] process multiple LR face images to recover an HR image by exploiting a patch-wise mixture of the probabilistic PCA prior instead of the holistic PCA prior in [Liu et al., 2007]. Hence, face hallucination methods that constrain the spatial positions of patches may avoid ghosting artifacts caused by PCA, but their performance degrades dramatically when LR image is not aligned precisely to the reference HR images.

To handle various poses and expressions, Tappen and Liu [2012] integrate the SIFT flow to align images. By exploiting local patterns, Yang et al. [2013] present a

structured face hallucination method. It first detects facial components in the given LR image and then transfers the corresponding HR facial components in the reference dataset to the LR input.

Very recently, deep convolutional neural networks based face hallucination methods are proposed. Zhou and Fan [2015] propose a bi-channel CNN to hallucinate face images in wild scenes. Since they require extraction of local features from the input images, the smallest input image size is limited to 48×48 pixels. Zhu et al. [2016b] employ a cascade bi-network, dubbed CBN, to upsample very low-resolution and unaligned faces, where the low-frequency parts are upsampled by a convolutional network and the high-frequency parts, *i.e.*, facial components, are firstly localized by a pre-defined model and then upsampled by the another network. Since CBN needs to localize facial components in LR images, CBN may produce ghosting faces when there are localization errors. Chen et al. [2018] present a two-stage network, where low-frequency components of LR face are first super-resolved and then face priors (*i.e.*, facial component locations) are also employed to enrich facial details. Nevertheless, these facial component based methods may fail to produce authentic HR face images due to potentially inaccurate landmark localization.

Xu et al. [2017] employ the framework of generative adversarial networks [Goodfellow et al., 2014; Radford et al., 2015] to recover blurry LR face images while enhancing the facial details by a multi-class discriminative loss. Dahl et al. [2017] leverage the framework of PixelCNN [Van Den Oord et al., 2016] to super-resolve very low-resolution faces. To relax the requirement of face alignment, Bulat and Tzimiropoulos [2017b] present a constraint that the landmarks of the upsampled faces should be close to the landmarks detected in their ground-truth images. Since ground-truth landmarks are not provided in the training stage and erroneous localization of landmarks may lead to distorted upsampled face images, their results are only restricted to 64×64 pixels and facial details are not sharp enough.

1.2.3 Face Frontalization

Generating a frontal face from a single non-frontal face image is very challenging due to self-occlusions and various pose variations, and has received significant attention in computer vision. Seminal works date back to the 3D Morphable Model (3DMM) [Blanz and Vetter, 1999], where a face is represented by the shape and texture bases in PCA subspace. After obtaining the the shape and texture coefficients of an input face image, Blanz and Vetter [1999] render novel views of an input face. Driven by 3DMM, Yang et al. [2011] estimate 3D surface from face appearance and then synthesize new expressions of the given face. However, these methods require the input face images to be nearly frontal in order to estimate the shape and appearance coefficients of input faces in PCA subspace. Dovgard and Basri [2004] exploit the facial symmetry to estimate 3D geometry of the given faces and render frontal faces. Similarly, Hassner et al. [2015] use facial symmetry to render out-of-view facial regions. Several methods [Asthana et al., 2011; Hassner, 2013; Taigman et al., 2014; Masi et al., 2016; Zhu et al., 2015] attempt to reconstruct frontal views by mapping

a 2D face image onto a 3D reference surface mesh after registering and normalizing the face image. Since they need to detect facial landmarks in the input images and establish correspondences of landmark points to 3D or 2D reference models, they require images in sufficiently high resolutions. Based on the fact that frontal faces have the minimum rank of all different poses, Sagonas et al. [2015] propose a statistical face frontalization method, but the appearance of their frontalized faces may not be consistent with the input faces.

Deep learning based face frontalization methods have been proposed recently as well [Zhu et al., 2014; Yim et al., 2015; Zhu et al., 2015; Tran et al., 2017b; Cole et al., 2017; Huang et al., 2017b; Chang et al., 2017; Yin et al., 2017]. Zhu et al. [2014] present a deep neural network to frontalize HR faces by exploiting the symmetry and similarity of facial components. Their method does not require estimation of a 3D model, but it cannot maintain appearance similarity between the frontalized and input faces either. Yim et al. [2015] develop a multi-task deep neural network to rotate faces, but their method outputs blurry frontal faces due to the aggressive downsampling operations in the encoder. Similarly, Cole et al. [2017] learn to generate facial landmarks and textures from features extracted by a face recognition network. Since Cole *et al.*warp input faces to the mean face geometry by using facial landmarks, the resolutions of their inputs need to be sufficiently large.

Very recently, Huang et al. [2017b] employ two deep neural networks, *i.e.*, global and local networks, to frontalize faces. However, their local network needs to extract HR facial components for identity preservation and to align HR facial components to pre-defined positions, and thus their method is not suitable for very LR unaligned non-frontal face images. Yin et al. [2017] combine 3DMM and a generative adversarial network to frontalize faces with arbitrary poses. They also need to localize facial landmarks when mapping the input faces to the 3DMM. Thus their method requires sufficient resolutions for input images. Tran et al. [2017a] present a convolutional neural network to regress 3DMM shape and texture parameters to speed up the optimization of 3DMM, but their method does not render frontalized faces which are similar to the input faces in terms of image intensity. Instead of localizing facial landmarks explicitly in the face images, Chang et al. [2017] employ a simple CNN to regress 6 degrees of freedom (6DoF) 3D head poses from image intensities. Then the estimated 6DoF parameters can be used to align face images without localizing facial landmarks explicitly. By transforming input image intensities with the estimated parameters, Chang et al. [2017] can preserve the appearance similarity between the input faces and their counterparts in the generated views. However, since their method needs to project facial landmarks from a 3D model to the input face images when rendering faces in new views, landmark misalignments between the 3D model and real face images may lead to artifacts in the generated images.

1.2.4 Generative Adversarial Networks

Image generation also has a close relationship to face hallucination when generated images are faces. Goodfellow et al. [2014] propose a generative adversarial network

(GAN), which is able to generate random face images from nothing but random noise, but the resolution of constructed images is limited (*i.e.*48×48 pixels) due to difficulty in training. Later, variants of GANs have been proposed to increase the resolutions and quality of generated images [Denton et al., 2015; Radford et al., 2015; Zhao et al., 2016; Arjovsky et al., 2017; Berthelot et al., 2017].

Similarly, variational auto-encoders (VAE) [Kingma and Welling, 2013] exploit neural networks to generate an entirely new image that is endowed similar properties to the training data distribution from a random noise input, but their results suffer from blurriness due to the lack of the discrimination in the model. Thus, the adversarial loss is fused into VAE, known as VAE-GAN, to enhance the quality of generated images [Larsen et al., 2016].

Instead of generating face images from noise, Reed et al. [2016] and Zhang et al. [2017b] generate images based on textual inputs. Yan et al. [2016] use a conditional CNN to generate faces based on attribute vectors. Perarnau et al. [2016] develop an invertible conditional GAN to generate new faces by manipulating facial attributes of the input images, while Shen and Liu [2016] change attributes of an input image on its residual image by training two generative networks in a complementary fashion. Since the above methods aim at generating new face images rather than super-resolving faces, they cannot guarantee the identity information as well as the appearance of the generated HR faces to be consistent with the input LR counterparts.

Conditional GANs have been used for the task of generating photographs from sketches [Sangkloy et al., 2017; Nejati and Sim, 2011; Yuen and Man, 2007; Tang and Wang, 2003; Sharma and Jacobs, 2011], or from semantic layout and scene attributes [Karacan et al., 2016]. Isola et al. [2016] develop "pix2pix" framework which uses Unet architecture and the patch-GAN to transfer low-level features from the input to the output domain. In addition, pix2pix framework needs paired images from both of the domains to train the networks. Considering paired images from two different domains may not be available, Zhu et al. [2017] present a CycleGAN to bridge two domains by unpaired images. Since these methods are patch based generative networks and may fail to capture the global structure of faces, these approaches produce visual artifacts when they are applied to transfer the style of face images.

1.2.5 Style Transfer

Style transfer is a technique which can render a given content image (input) by incorporating a specific painting style while preserving the contents of input. Style transfer methods can be roughly grouped into two categories: image optimization-based methods and feed-forward style transfer methods.

The seminal optimization-based work [Gatys et al., 2016b] transfers the style of an artistic image to a given photograph. It uses an iterative optimization to generate a target image which is randomly initialized (Gaussian distribution). During the optimization step, the statistics of the neural activations of the target, the content and style images are matched. The idea of Gatys et al. [2016b] inspires many follow-up

studies. Yin [2016] presents a content-aware style transfer method which initializes the optimization algorithm with a content image instead of a random noise. Li and Wand [2016a] propose a patch-based style transfer method by combining MRF and CNN. The work [Gatys et al., 2016a] proposes to transfer the style by using linear models, and it preserves colors of content images by matching color histograms. Gatys et al. [2017] decompose styles into perceptual factors and then manipulate them for the style transfer over spatial locations, color information and across spatial scales. Selim et al. [2016] modify the content loss through a gain map for the head portrait painting transfer. Risser et al. [2017] use histogram-based losses in their objective and build on the algorithm of Gatys et al. [2016b]. Although the above optimization-based methods further improve the quality of style transfer, they are computationally expensive due to the iterative optimization procedure, thus limiting their practical use.

To address the inefficient computational speed, feed-forward methods are proposed to replace the original on-line iterative optimization step with training a feed-forward neural network off-line and generating stylized images on-line [Ulyanov et al., 2016a; Johnson et al., 2016; Li and Wand, 2016b].

Johnson et al. [2016] train a generative network for a fast style transfer by using perceptual loss functions. The perceptual loss function is employed to control the content similarity between the stylized images and their original ones. The architecture of the generator network follows the work [Radford et al., 2015] and also uses residual blocks to increase the capacity of the network. Another concurrent work [Ulyanov et al., 2016a], named Texture Network, employs a multi-resolution architecture in the generator network. Furthermore, Ulyanov et al. [2016b] and Ulyanov et al. [2017] replace the spatial batch normalization with the instance normalization to achieve a faster convergence. Wang et al. [2017] enhance the granularity of the feed-forward style transfer with a multimodal CNN which performs stylization hierarchically via multiple losses deployed across multiple scales. Those feed-forward methods perform stylization $\sim$1000 times faster than the optimization-based methods. However, they cannot adapt to arbitrary styles that are not used for training. For instance, in order to synthesize an image from a new style, the entire networks need retraining.

To deal with such a restriction, a number of recent approaches encode multiple styles within a single feed-forward network [Dumoulin et al., 2016; Chen and Schmidt, 2016; Chen et al., 2017; Li et al., 2017a]. Dumoulin et al. [2016] use conditional instance normalization that learns normalization parameters for each style. Given feature activations of the content and style images, Chen and Schmidt [2016] replace content features with the closest-matching style features in a patch-by-patch manner. Chen et al. [2017] present a network that learns a set of new filters for every new style. Similarly, Li et al. [2017a] also adapt a single feed-forward network via a texture controller module which forces the network to synthesize the desired style only. Because style transfer aims to generate multiple stylized images from one image, the existing feed-forward approaches need to compromise between the generalization [Li et al., 2017a; Huang and Belongie, 2017; Zhang and Dana, 2017]

and quality [Ulyanov et al., 2017, 2016b; Gupta et al., 2017]. On the contrary, we target at generating high-quality realistic face images from multiple stylized faces, and style information is unknown beforehand. Therefore, state-of-the-art style transfer methods cannot be directly applied to our task.

1.3 Thesis Outline

This thesis is formatted as a compilation of my publications during my PhD period at The Australian National University.

In chapter 2, we develop a discriminative generative network to super-resolve aligned very LR face images. Our network is able to upsample LR faces with different pose and expression variations by a large upscaling factor 8×.

In chapter 3, we present a single network to ease the training difficulty of GAN based upsampling networks as well as reduce artifacts caused by deconvolutional layers and discriminative networks. We notice that our network may generate blurry upsampled HR faces to achieve better quantitative results in terms of peak signal-to-noise ratio (PSNR). Thus, we employ a post-processing step to enhance visual quality of final results.

In chapter 4, we present a transformative discriminative network which embeds spatial transformer network into the upsampling network to super-resolve unaligned LR input images while aligning them automatically.

In chapter 5, we tackle the noisy unaligned very LR face images by exploiting a transformative discriminative autoencoder, where a decoder-encoder-decoder structure is proposed to reduce noise while hallucinating input face images.

In chapter 6, we propose a two branch upsampling network to receive LR inputs in different resolutions ranging from 16×16 pixels to 32×32 pixels. In this manner, our network does not lose information of input images when feeding them into the upsampling network.

Our previous works can super-resolve faces in nearly frontal poses. When the pose variations are large, such as side-view faces, the proposed methods may fail to localize the low-resolution facial patterns accurately, thus generating severe artifacts in the upsampled faces. In chapter 7, we incorporate the facial structure into the super-resolution process to handle input faces with large pose variations. Thus, we propose a multitask convolutional neural network which not only super-resolves faces but also predicts facial components. The estimated structure information from upsampled feature maps is also exploited to facilitate face hallucination in return.

Even though the previous chapters propose different schemes to super-resolve faces in different situations, the inherently ill-posed nature of super-resolution may still lead to inaccurate hallucinated faces, such as gender reversal and age rejuvenation. In chapter 8, we introduce an attribute embedded face super-resolution network to mitigate the uncertainty caused by one to many mappings especially when the magnification factor is very large, such as 8×.

Unlike the previous works which only focus on super-resolving input faces, in

chapter 9, we design a transformative adversarial neural network to jointly frontalize and super-resolve LR face images. The LR profile faces are firstly projected to latent representations which are enforced to be similar to the representations of their frontal counterparts and then we upsample the encoded representations by deconvolutional layers. In this fashion, we achieve frontalized HR face images from the corresponding LR non-frontal ones.

Inspired by our previous works on hallucinating LR face images, in chapter 10, we develop a style removal network to recover photo-realistic face images from stylized portraits by employing an ℓ_2 loss to constrain the appearance similarity between the destylized faces and the ground-truth ones.

In chapter 11, we further exploit a feature-wise similarity constraint, known as perceptual loss, to enforce the identity similarity between the destylized faces and their ground-truths, as well as spatial transformer networks to align the destylized faces simultaneously, which facilitate the proposed network to learn common facial patterns.

During the procedure of face destylization, the original facial attributes, such as skin and hair colors, are hard to be determined from the input stylized portraits. In chapter 12, we present a face destylization network to restore photo-realistic faces by using supplementary facial attributes. Benefiting from the high-level semantic information (*i.e.*, facial attributes), we can restore the facial details much closer the ground-truth ones, even including skin and hair colors. Furthermore, we are able to generate different destylized faces by editing the attributes rather than a deterministic one, thus increasing the flexibility of face destylization.

In chapter 13, we draw the conclusions of our thesis and provide some future research directions.

Chapter 2

Ultra-Resolving Face Images by Discriminative Generative Networks

2.1 Foreword

As mentioned in the previous chapter, state-of-the-art methods require LR faces to be precisely aligned and to have similar poses and expressions to the HR exemplary training dataset. In order to relax those requirements as well as achieve better super-resolution performance, we exploit a deep convolutional neural network to super-resolve LR faces, regarding that convolutional neural networks are robust to translational misalignments and able to explore similar facial patterns from different individuals. Therefore, we propose a discriminative generative network to super-resolve face images by a large upscaling factor of $8\times$ in this chapter.

This chapter has been published as a conference paper: Xin Yu, Fatih Porikli: Ultra-Resolving Face Images by Discriminative Generative Networks. In *European Conference on Computer Vision (ECCV)*, 2016.

2.2 Abstract

Conventional face super-resolution methods, also known as face hallucination, are limited up to $2\sim4\times$ scaling factors where $4 \sim 16$ additional pixels are estimated for each given pixel. Besides, they become very fragile when the input low-resolution image size is too small that only little information is available in the input image. To address these shortcomings, we present a discriminative generative network that can ultra-resolve a very low resolution face image of size 16×16 pixels to its $8\times$ larger version by reconstructing 64 pixels from a single pixel. We introduce a pixel-wise ℓ_2 regularization term to the generative model and exploit the feedback of the discriminative network to make the upsampled face images more similar to real ones. In our framework, the discriminative network learns the essential constituent parts of the faces and the generative network blends these parts in the most accurate fashion to the input image. Since only frontal and ordinary aligned images are used in train-

ing, our method can ultra-resolve a wide range of very low-resolution images directly regardless of pose and facial expression variations. Our extensive experimental evaluations demonstrate that the presented ultra-resolution by discriminative generative networks (UR-DGN) achieves more appealing results than the state-of-the-art.

2.3 Motivation

Face images arguably carry the most interesting and valuable visual information and can be obtained in a non-intrusive manner. Still, for many applications from content enhancement to forensics, face images require significant magnification.

In order to generate high-resolution (HR) face images from low-resolution (L-R) inputs, face hallucination [Baker and Kanade, 2000; Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Yang et al., 2010; Ma et al., 2010; Tappen and Liu, 2012; Yang et al., 2013; Zhou and Fan, 2015; Wang et al., 2014] attracted great interest in the past. These state-of-the-art face hallucination methods can achieve exciting results up to $4\times$ upscaling factors when accurate facial features and landmarks can be found in LR images [Tappen and Liu, 2012; Yang et al., 2013], manual supervision is provided, suitably similar HR images of the same person are included in the support dataset, and the exemplar HR face images are densely aligned [Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Yang et al., 2010]. When the input image resolution becomes smaller, landmark based methods fail gravely because of erroneous landmark localization. In other words, their performances highly depend on the input image size. Furthermore, when the appearances of the input LR images are different from the HR images in the dataset due to pose, lighting and expression changes, subspace based methods degrade by producing ghosting artifacts in the outputs.

When ultra-resolving ($8\times$ scaling factor) a low-resolution image, almost 98.5% of the information is missing. This is a severely ill-posed problem. As indicated in [Yang et al., 2014], when the scaling factor increases to $8\times$, the performances of existing approaches degrade acutely.

Our intuition is that by better exploring the information available in the natural structure of face images, appearance similarities between individuals, and emerging large-scale face datasets [Huang et al., 2007; Liu et al., 2015], it may be possible to derive competent models to reconstruct authentic $8\times$ magnified HR face images. Deep neural networks, in particular convolutional neural networks (CNN), are inherently suitable for learning from large-scale datasets. Very recently, CNN based generic patch super-resolution methods have been proposed [Dong et al., 2016a; Kim et al., 2016a] without focusing on any image class. A straightforward retraining (fine-tuning) of these networks with face image patches cannot capture the global structure of faces. As shown in Fig. 2.1(e), these networks fail to produce realistic and visually pleasant results. In order to retain the global structure of faces while being able to reconstruct instance specific details, we use whole face images to train our networks.

(a) LR (b) HR (c) NN (d) Bicubic (e) Dong et al. (f) Ours (g) Ours

Figure 2.1: Comparison of our UR-DGN over CNN based super-resolution. (a) 16×16 pixels LR face images [given]. (b) 128×128 original HR images [not given]. (c) The nearest neighbors of (a) in the training set. (d) Upsampling by bicubic interpolation. (e) The results generated by the CNN based super-resolution [Dong et al., 2016a]. This network is *retrained* with face images. (f) Our UR-DGN without the feedback of the discriminative model. (g) Our UR-DGN.

We are inspired from the generative adversarial network (GAN) [Goodfellow et al., 2014] that consists of two topologies: a generative network G that is designed to learn the distribution of the training data samples and generate a new sample similar to the training data, and a discriminative network D that estimates the probability that a sample comes from the training dataset rather than G. This work is empowered with a Laplacian pyramid by Denton et al. [2015] to progressively generate images due to the higher dimensional nature of the training images. One advantage of GAN is that it generates face images yet sharp images from nothing but noise. However, it has two serious shortcomings: (i) The output faces are totally random. (ii) GAN has fixed output size limitation (32×32 [Goodfellow et al., 2014] and 64×64 [Denton et al., 2015] pixels). Therefore, GAN cannot be used for ultra-resolution directly.

Instead of noise, we apply the LR face image l as the input for our discriminative-generative network (DGN) and then generate an HR face image $\hat{h}$. In order to enforce the similarity between the generated HR face image $\hat{h}$ and the exemplar HR image h, we impose a pixel-wise ℓ_2 regularization on the differences between $\hat{h}$ and h in the generative network. This enables us to constrain the affinity between the exemplar HR images and the generated HR images. Hereby, a loss function layer is added to G. Finally, the generative network G produces an HR image consistent with the exemplar HR image. In training DGN, the discriminative network D provides feedback to G to distinguish whether the upsampled face image is considered (classified by the D) as real (sharp) or as generated (smooth). As shown in Fig. 2.1(f), by directly upsampling images by the generative network G, we are not able to obtain face images with sharp details. In contrast, with the help of the network D, we can generate much sharper HR face images, as shown in Fig. 2.1(g). Since the discriminative network is designed to distinguish between the real face images and generated ones, the generative network can produce HR face images more similar to real images.

Our method does not make any explicit assumption or require the location of the facial landmarks. Because the convolutional neural network topologies we use provide robustness to translations and deformations, our method does not need densely aligned HR face images or constrain the face images to controlled settings, such as the same pose, lighting and facial expression. Our approach only requires frontal and approximately nearby eye locations in the training images, which can be easily satisfied in most of face datasets. Hence, our UR-DGN method can ultra-resolve $8\times$ a wide range of LR images without taking other information into account.

Overall, the contributions of this paper are mainly in four aspects:

- We present a novel method to ultra-resolve, $8\times$ scaling factor, low-resolution face images. The size of our input low-resolution images is tiny, 16×16 pixels, which makes the magnification task even more challenging as almost all facial details are missing. We reconstruct 64 pixels from only 1 pixel.
- To the best of our knowledge, our method is the first attempt to develop discriminative generative networks for generating authentic face images. We demonstrate that our UR-DGN achieves better visual results than the state-of-the-art.
- We show that by introducing a pixel-wise ℓ_2 regularization term into the network and backpropagating its residual, it is possible to ultra-resolve in any size while GANs can only generate images in fixed and small sizes.
- When training our network, we only require frontal and approximately aligned images, which makes the training datasets more attainable. Our UR-DGN can ultra-resolve regardless of pose, lighting and facial expressions variations.
- Due to its feed-forward topology, our ultra-resolution method is very fast.

2.4 Related Work

Super-resolution can be basically classified into two categories: generic super-resolution methods and class-specific super-resolution methods. When upsampling LR images, generic methods employ priors that ubiquitously exist in natural images without considering any image class information. Class-specific methods, also called face hallucination [Baker and Kanade, 2000] if the class is face, aim to exploit statistical information of objects in a certain class. Thus, they usually attain better results than generic methods when super-resolving images of a known class.

Generic super-resolution: In general, generic single image super-resolution methods have three types: interpolation based methods, image statistics based methods [Peleg and Elad, 2014; Yang and Yang, 2013] and example (patch)-based methods [Freeman et al., 2002; Hong Chang et al., 2004; Glasner et al., 2009; Yang et al., 2010; Schulter and Leistner, 2015; Huang et al., 2015]. Interpolation based methods such as bilinear and bicubic upsampling are simple and computationally efficient, but as the

scaling factor increases, they generate overly smooth edges and fail create high resolution details. Image statistics based methods employ natural image priors to predict HR images, but they are limited to smaller scaling factors [Lin and Shum, 2006]. Example-based methods have potential to break this limitation of the maximum scaling factor. Several works [Glasner et al., 2009; Freedman and Fattal, 2010; Singh et al., 2014; Huang et al., 2015] exploit self-similarity of patches in an input image to generate high resolution patches. Freeman et al. [2002] and Hong Chang et al. [2004] construct LR and HR patch pairs from a training dataset, and then the nearest neighbor of the input patch is searched in the LR space. The HR output is reconstructed from the corresponding HR patch. Yang et al. [2010] propose a sparse representation formulation by reconstructing corresponding LR and HR dictionaries, while Gu et al. [2015] apply convolutional sparse coding instead of patch-based sparse coding. Recently, several deep learning based methods [Dong et al., 2016a; Kim et al., 2016a] have been proposed. Dong et al. [2016a] incorporate convolutional neural networks to learn a mapping function between LR and HR patches from a large-scale dataset. Since many different HR patches may correspond to one LR patch, the output images would suffer from artifacts at the intensity edges. In order to reduce the ambiguity between the LR and HR patches, Bruna et al. [2016] exploit the statistical information learned from deep convolutional network to reduce ambiguity between LR and HR patches.

Face hallucination: Unlike generic methods, class-specific super-resolution methods [Baker and Kanade, 2000; Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Ma et al., 2010; Tappen and Liu, 2012; Yang et al., 2013; Zhou and Fan, 2015; Wang et al., 2014] further exploit the statistical information in the image categories, thus leading to better performances. In one of the earlier works, Baker and Kanade [2000] build the relationship between HR and LR patches using Bayesian formulation such that high-frequency details can be transferred from the dataset for face hallucination. It can generate face images with richer details. However, artifacts also appear due to the possible inconsistency of the transferred HR patches.

The work [Wang and Tang, 2005] employs constraints on both LR and HR images, and then hallucinate HR face images by an eigen-transformation. Although it is able to magnify LR images by a large scaling factor, the output HR images suffer from ghosting artifacts as a result of using a subspace. Similarly, Liu et al. [2007] enforce linear constraints for HR face images using a subspace learned from the training set via Principle Component Analysis (PCA), and a patch-based Markov Random Field is proposed to reconstruct the high-frequency details in the HR face images. This method works only when the images are precisely aligned at fixed poses and expressions. In other cases, the results usually contain ghosting artifacts due to PCA based holistic appearance model. To mitigate artifacts a blind bilateral filtering is used as a post-processing step. Instead of imposing global constraints, Ma et al. [2010] use multiple local constraints learned from exemplar patches, and Li et al. [2014] reserve to sparse representation on the local structures of faces. Kolouri and Rohde [2015] use optimal transport in combination with subspace learning to morph

an HR image from the LR input. These subspace based methods require that face images in the dataset are precisely aligned and the test LR image has the same pose and facial expression as the HR face images.

In order to handle various poses and expressions, Tappen and Liu [2012] integrate SIFT flow to align images. This method performs adequately when the training face images are highly similar to the test face image in terms of identity, pose, and expression. Since it uses local features to match image segments, the global structure is not preserved either.

By exploiting local structures of face images, Yang et al. [2013] present a structured face hallucination method. It divides a face image into facial components, and then maintains the structure by matching gradients in the reconstructed output. However, this method relies on accurate facial landmark points that are usually unavailable when the image size is very small. The recent work in [Zhou and Fan, 2015] proposes a bichannel CNN to hallucinate face images in the wild. Since it needs to extract features from the input images, the smallest input image size is 48×48.

Some generative networks [Kingma and Welling, 2013; Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015] can generate random face images from nothing but random noise. Among those generative models, generative adversarial networks (GANs) [Goodfellow et al., 2014; Denton et al., 2015] can generate face images with much sharper details due to the discriminative network. However, the generated images are only similar in the class domain but different in the appearance domain. In other words, GAN is capable of generating only random faces. Moreover, GAN only uses the cross entropy loss function of discriminative models to optimize the entire network. Hence, the generative models in GAN are difficult to generate images in high resolutions. For instance, Goodfellow et al. [2014] only produce images of size 32×32 pixels.

2.5 Proposed Ultra-Resolution Method

A processing pipeline of UR-DGN is shown in Fig. 2.2. Below, we present the pipeline of UR-DGN and describe the details of training the network. We also discuss the differences between UR-DGN and GAN.

2.5.1 Model Architecture

Let us first recap the generative model G that takes a noise vector z from a distribution $P_{noise}(z)$ as an input and then outputs an image $\hat{x}$ in [Goodfellow et al., 2014]. The discriminative model D takes an image stochastically chosen from either the generated image $\hat{x}$ or the real image x drawn from the training dataset with a distribution $P_{data}(x)$ as an input. D is trained to output a scalar probability, which is large for real images and small for generated images from G. The generative model G is learned to maximize the probability of D making a mistake. Thus a minmax

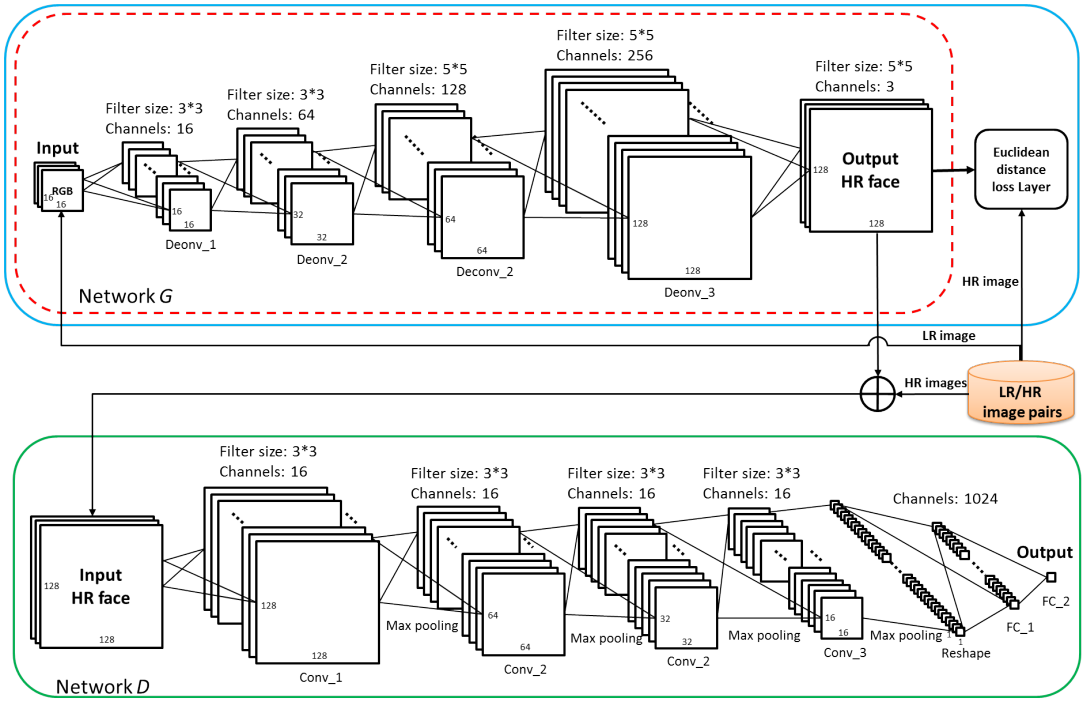


Figure 2.2: The pipeline of UR-DGN. In the testing phase, only the generative network in the red dashed block is employed.

objective is used to train these two models simultaneously

$$\min_G \max_D \mathbb{E}_{x\sim P_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim P_{noise}(z)}[\log(1 - D(G(z)))]. \tag{2.1}$$

This equation encourages G to fit $P_{data}(x)$ so as to fool D with its generated samples $\hat{x}$.

We cannot directly employ Eqn. 2.1 for the ultra-resolution task since GAN takes noise as input to learn the distribution on the training dataset. In UR-DGN, we design a deconvolutional network [Zeiler et al., 2011] as the generative model G to ultra-resolve LR inputs, and a convolutional network as the discriminative model D. We construct LR and HR face image pairs $\{l_i, h_i\}$ as the training dataset. Because the generated HR face image $\hat{h}_i$ should be similar to its corresponding HR image h_i, a pixel-wise ℓ_2 regularization term induces the similarity. Thus, the objective function $F(G, D)$ is modeled as follows:

$$\begin{aligned}\min_G \max_D F(G,D) &= \mathbb{E}_{h_i\sim P_H(h)}[\log D(h_i)] + \mathbb{E}_{l_i\sim P_L(l)}[\log(1 - D(G(l_i)))] \\ &\quad + \lambda \mathbb{E}_{(h_i,l_i)\sim P_{HL}(h,l)}[\|\hat{h}_i - h_i\|_F^2] \\ &= \mathbb{E}_{h_i\sim P_H(h)}[\log D(h_i)] + \mathbb{E}_{l_i\sim P_L(l)}[\log(1 - D(G(l_i)))] \\ &\quad + \lambda \mathbb{E}_{(h_i,l_i)\sim P_{HL}(h,l)}[\|G(l_i) - h_i\|_F^2],\end{aligned} \tag{2.2}$$

where $P_L(l)$ and $P_H(h)$ represent the distributions of LR and HR face images respectively, $P_{HL}(h,l)$ represents the joint distribution of HR and LR face images, and λ is a trade-off weight to balance the cross entropy loss of D and the Euclidean distance loss of G.

2.5.2 Training of the Network

The parameters of the generative network G and the discriminative network D are updated by backpropagating the loss in Eqn. 2.2 through their respective networks. Specifically, when training G, the loss of the last two terms in Eqn. 2.2 is backpropagated through G to update its parameters. When training D, the loss of the first two terms in Eqn. 2.2 is backpropagated through D to update its parameters.

Training D: Since D is a CNN with a negative cross-entropy loss function, backpropation is used to train the parameters of D. Thus, the derivative of the loss function $F(G,D)$ with respect to D is required when updating the parameters in D. It is formulated as follows:

$$\frac{\partial F(G,D)}{\partial D} = \nabla_{\theta_D}\left(\mathbb{E}_{h_i\sim P_H(h)}[\log D(h_i)] + \mathbb{E}_{l_i\sim P_L(l)}[\log(1-D(G(l_i)))]\right), \tag{2.3}$$

where θ_D is the parameters of D, and ∇ is the derivative operator. Specifically, given a batch of LR and HR image pairs $\{l_i,h_i\}, i=1,\ldots,N$, the stochastic gradient of the discriminator D is written as

$$\frac{\partial F(G,D)}{\partial D} = \nabla_{\theta_D}\left(\frac{1}{N}\sum_{i=1}^{N}\log D(h_i) + \log(1-D(G(l_i)))\right), \tag{2.4}$$

where N is the number of LR and HR face image pairs in the batch. Since we need to maximize D, the parameters θ_D are updated by ascending their stochastic gradients. RMSprop [Hinton, 2012] is employed to update the parameters θ_D as follows:

$$\begin{aligned} \delta^{j+1} &= \alpha\delta^j + (1-\alpha)(\frac{\partial F(G,D)}{\partial D})^2, \\ \theta_D^{j+1} &= \theta_D^j + \eta\frac{\partial F(G,D)}{\partial D}/\sqrt{\delta^{j+1}+\epsilon}. \end{aligned} \tag{2.5}$$

where η and α represent the learning rate and the decay rate respectively, j indicates the iteration index, ϵ is set to 10^{-8} as a regularizer to avoid division by zero, and δ is an auxiliary variable.

Training G: G is a deconvolutional neural network [Zeiler et al., 2011]. It is trained by backpropagation as well. Similar to training D, the derivative of the loss function $F(G,D)$ with respect to G is written as

$$\begin{aligned} \frac{\partial F(G,D)}{\partial G} = \nabla_{\theta_G}\Big(&\mathbb{E}_{l_i\sim P_L(l)}[\log(1-D(G(l_i)))] \\ &+\lambda\mathbb{E}_{(h_i,l_i)\sim P_{HL}(h,l)}[\|G(l_i)-h_i\|_F^2]\Big), \end{aligned} \tag{2.6}$$

Algorithm 1 Minibatch stochastic gradient descent training of UR-DGN

Input: minibatch size N, LR and HR face image pairs $\{l_i, h_i\}$, maximum number of iterations K.

1: **while** iter $<$ K **do**
2: Choose one minibatch of LR and HR image pairs $\{l_i, h_i\}, i = 1, \ldots, N$.
3: Generate one minibatch of HR face images $\hat{h}_i$ from $l_i, i = 1, \ldots, N$, where $\hat{h}_i = G(l_i)$.
4: Update the parameters of the discriminative network D by using Eqn. 2.4 and Eqn. 2.5.
5: Update the parameters of the generative network G by using Eqn. 2.7 and Eqn. 2.8.
6: **end while**

Output: UR-DGN.

where θ_G denotes the parameters of G. Given a batch of LR and HR face image pairs $\{l_i, h_i\}, i = 1, \ldots, N$, the stochastic gradient of the generator G is

$$\frac{\partial F(G,D)}{\partial G} = \nabla_{\theta_G} \left(\frac{1}{N} \sum_{i=1}^{N} \log(1 - D(G(l_i))) + \lambda \|G(l_i) - h_i\|_F^2 \right). \tag{2.7}$$

Since we will minimize the cost function for G, the parameters θ_G are updated by descending their stochastic gradients as follows:

$$\begin{aligned} \delta^{j+1} &= \alpha \delta^j + (1-\alpha)(\frac{\partial F(G,D)}{\partial G})^2, \\ \theta_G^{j+1} &= \theta_G^j - \eta \frac{\partial F(G,D)}{\partial G} / \sqrt{\delta^{j+1} + \epsilon}. \end{aligned} \tag{2.8}$$

In our algorithm, we set the learning rate η to 0.001 and the decay rate to 0.01, and the learning rate is multiplied by 0.99 after each epoch. Since we super-resolve an image rather than generate a face image, we set λ to 100 to constrain the similarity between the generated face image $G(l_i)$ and the exemplar HR face image h_i. The training procedure of our UR-DGN is presented in Algorithm 1.

2.5.3 Ultra-Resolution of a Given LR Image

The discriminative network D and the pixel-wise ℓ_2 regularization are only required in the training phase. In the ultra-resolution (testing) phase, we take LR face images as the inputs of the generative network G, and the outputs of G are the ultra-resolved face images. This end-to-end mapping is able to keep the global structure of HR face images while reconstructing local details.

(a) LR (b) HR (c) GAN* (d) GAN (e) Ours

Figure 2.3: Illustration of the differences between GAN and our UR-DGN. (a) Given LR image. (b) Original HR image (not used in training). (c) GAN*: GAN with no fully connected layer. Without a fully connected layer, GAN* cannot rearrange the convolutional layer features (activations) of the input noise to a face image. (d) GAN with fully connected layer. Given the test LR image (not noise!), GAN still outputs a random face image. (d) Result of our UR-DGN.

2.5.4 Differences between GAN and UR-DGN

GAN of [Goodfellow et al., 2014] consists of fully connected layers, while Denton et al. [2015] use a fully connected layer and deconvolutional layers. In [Denton et al., 2015], the noise input is required to be fed into a fully connected layer first before fed into deconvolutional layers. The fully connected layer can be considered as a nonlinear mapping from the noise to the activations of a feature map. If we remove the fully connected layer while leaving other layers unchanged, GAN will fail to produce face images, as shown in Fig. 2.3(c). Therefore, fully connected layers are necessary for GAN.

Since deconvolutional layers are able to project low-resolution feature maps back to high-resolution image space, we take an LR face image as a 3-channel feature map, and then project this LR feature map into the HR face image space. However, the fully connected layers are not necessary in our UR-DGN. Because LR face images are highly structured, they can be regarded as feature maps after normalization, which scales the range of intensities between -1.0 and 1.0. Feeding an LR face image into a fully connected layer may destroy the global structure of the feature map, *i.e.* the input LR face image. In other words, UR-DGN does not need a nonlinear mapping from an input LR image to a feature map via a fully connected layer.

Furthermore, since there is no pixel-wise regularization in GAN, it cannot produce HR results faithful to the input LR face images and generate high-quality face images as the output size increases as shown in Fig. 2.3(d). In conclusion, the original architecture of GAN cannot be employed in the ultra-resolution problem.

2.6 Experiments

In order to dissect the performance of UR-DGN, we evaluate it qualitatively and quantitatively, and compare with the state-of-the-art methods [Liu et al., 2007; Yang et al., 2010, 2013; Dong et al., 2016a; Ma et al., 2010]. The method of Liu et al. [2007] is a subspace based face hallucination method. The work in [Yang et al., 2010]

uses sparse representations to super-resolve HR images by constructing LR and HR dictionaries. The method of Yang et al. [2013] hallucinates face images by using facial components from exemplar images in the dataset. Dong et al. [2016a] employ CNN to upsample images. Ma et al. [2010] use position-patches in the dataset to reconstruct HR images.

2.6.1 Datasets

We trained UR-DGN with the celebrity face attributes (CelebA) dataset [Liu et al., 2015]. There are more than 200K images in this dataset, where Liu et al. [2015] use similarity transformation to align the locations of eye centers. We use the cropped face images for training. Notice that the images in this dataset cover remarkably large pose variations and facial expressions. We do not classify the face images into different subcategories according to their poses and facial expressions when training UR-DGN.

We randomly draw $16,000$ aligned and cropped face images from the CelebA dataset, and then resize them to 128×128. We use $15,000$ images for training, 500 images for validation, and 500 images for testing. Thus, our UR-DGN model never sees the test LR images in the training phase.

We downsample the HR face images to 16×16 pixels (without aliasing), and then construct the LR and HR image pairs $\{l_i, h_i\}$. The input of UR-DGN is an image of size 16×16 with 3 RGB channels, and the output is an image of size 128×128 with 3 RGB channels.

2.6.2 Comparisons with SoA

We do side-by-side comparisons with five state-of-the-art face hallucination methods. In case an approach does not allow $8\times$ scaling factor directly, *i.e.* [Yang et al., 2010] and [Dong et al., 2016a], we repeatedly (three times) apply a scaling factor $2\times$ when ultra-resolving an LR image. For a fair comparison, we use the same dataset CelebA for training of all other algorithms. Furthermore, we apply bicubic interpolation to all input LR images as another baseline.

Comparison with Liu *et al.*'s method: Since this method requires the face images in the dataset to be precisely aligned, it is difficult for it to learn a representative subspace from the CelebA dataset where face images have large variations. Therefore, the global model of the input LR image cannot be represented by the learned subspace, and its local model impels patchy artifacts on the output. As shown in Fig. 2.4(d), Fig. 2.5(d) and Fig. 2.6(d), this method cannot recover face details accurately, and suffers from distorted edges and blob-like artifacts.

Comparison with Yang *et al.*'s method: As illustrated in Fig. 2.4(e), Fig. 2.5(e) and Fig. 2.6(e), Yang *et al.*'s method does not recover high-frequency facial details. Besides, non-smooth over-emphasized edge artifacts appear in their results. As the scaling factor becomes larger, the correspondence between LR and HR patches becomes ambiguous. Therefore, their results suffer exaggerated pixellation pattern of

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 2.4: Comparison with the state-of-the-art methods on frontal faces. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Liu et al. [2007]. (e) The method of Yang et al. [2010]. (f) The method of Yang et al. [2013]. (g) The method of Dong et al. [2016a]. (h) The method of Ma et al. [2010]. (i) UR-DGN. (please zoom-in to see the differences between (f) and (g). In (f), there are artificial facial edges while (g) has jitter artifacts.)

the LR, similar to a contrast enhanced bicubic upsampled results.

Comparison with Yang *et al.*'s method: This method requires landmarks of facial components and building on them, and reconstructs transferred high-resolution facial components over the low-resolution image. In 16×16 input images, it is extremely difficult to localize landmarks. Hence, this method cannot correctly transfer facial components as shown in Fig. 2.4(f), Fig. 2.5(f) and Fig. 2.6(f). In contrast, UR-DGN does not need landmark localization and still preserve the global structure.

Comparison with Dong *et al.*'s method: It applies convolutional layers to learn a generic patch-based mapping function, and achieves state-of-the-art results on natural images. Even though we retrain their CNN on face images to suit better for face hallucination, this method cannot generate high-frequency facial details except some noisy spots in the HR images as shown in Fig. 2.4(g), Fig. 2.5(g) and Fig. 2.6(g).

Comparison with Ma *et al.*'s method: This method employs local constraints learned from positioned exemplar patches to avoid ghosting artifacts caused by a global model such as PCA. However, it requires the exemplar patches to be precisely

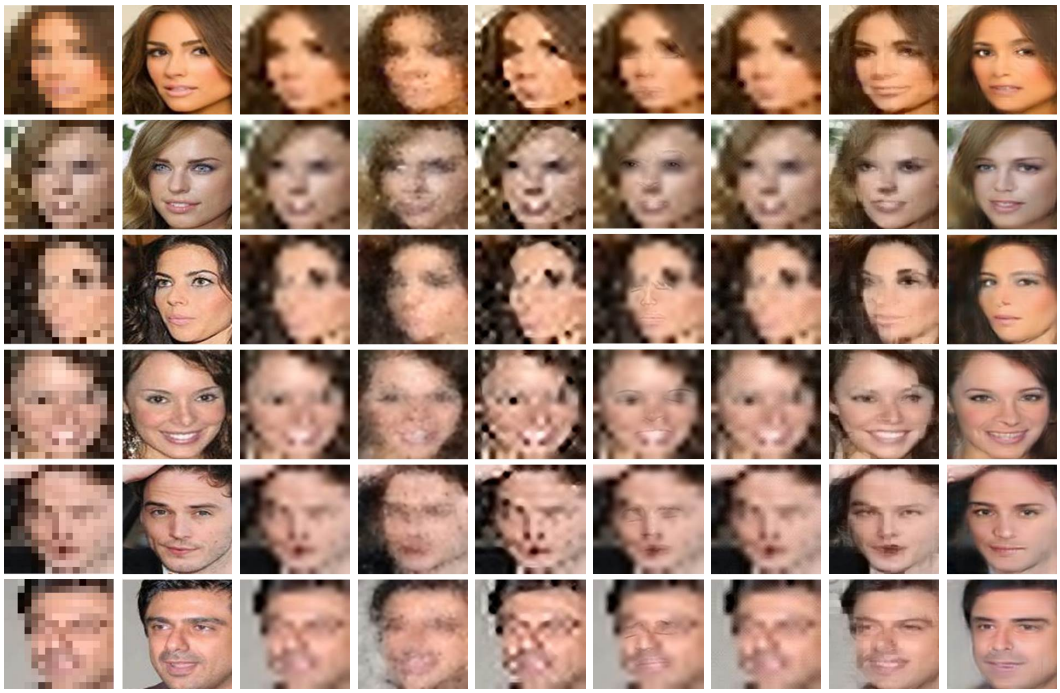
(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 2.5: Facial expression: Comparison with the state-of-the-art methods on images with facial expressions. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Liu et al. [2007]. (e) The method of Yang et al. [2010]. (f) The method of Yang et al. [2013]. (g) The method of Dong et al. [2016a]. (h) The method of Ma et al. [2010]. (i) UR-DGN. (please zoom-in to see the differences between (f) and (g))

Table 2.1: Quantitative comparisons on the entire test dataset

Methods	PSNR	SSIM
Bicubic	23.22	0.67
[Liu et al., 2007]	21.60	0.55
[Yang et al., 2010]	21.35	0.60
[Yang et al., 2013]	23.07	0.65
[Dong et al., 2016a]	23.11	0.65
[Ma et al., 2010]	23.12	0.64
Ours	**24.82**	**0.70**

aligned. As shown in Fig. 2.4(h), Fig. 2.5(h) and Fig. 2.6(h), this method suffers from obvious blocking artifacts and uneven oversmoothing as a result of the unaligned position patches in the dataset CelebA.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

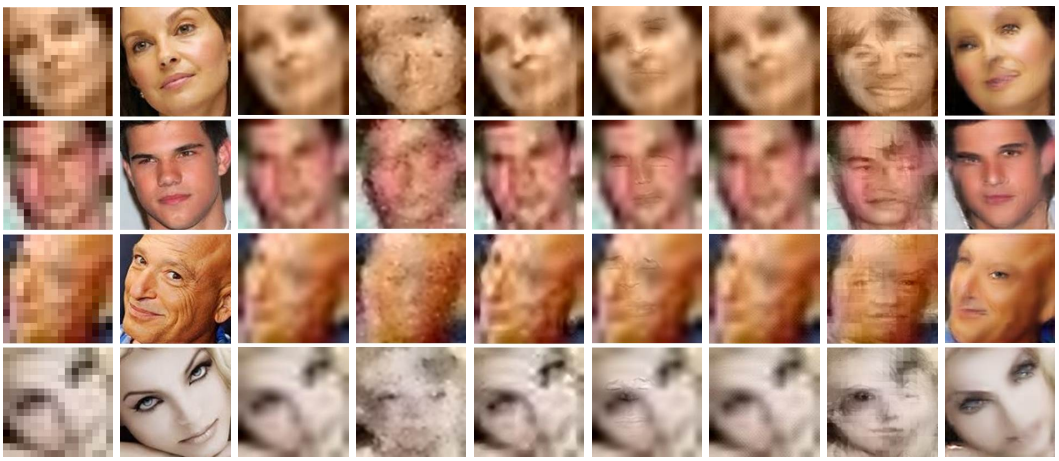
Figure 2.6: Pose: Comparison with the state-of-the-art methods on face images with different poses. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Liu et al. [2007]. (e) The method of Yang et al. [2010]. (f) The method of Yang et al. [2013]. (g) The method of Dong et al. [2016a]. (h) The method of Ma et al. [2010]. (i) UR-DGN. (please zoom-in to see the differences between (f) and (g))

In contrast to the above approaches, our method provides more visually pleasant HR face images that not only contain richer details but also are similar to the original (not given to our method). UR-DGN takes the input LR image as a whole and reduces the ambiguity of the correspondence between LR and HR patches. Our method attains much sharper results.

2.6.3 Quantitative Results

We also assess UR-DGN performance quantitatively by comparing the average PSNR and structural similarity (SSIM) on the entire test dataset. Table 2.1 shows that our method achieves the best performance. As expected, bicubic interpolation achieves better results than the other baselines since it explicitly builds on pixel-wise intensity values without any hallucination. Notice that bicubic interpolation achieves the second best results, which implies that the high-frequency details reconstructed by the state-of-the-art methods are not authentic. Our method on the other hand achieves facial details consistent with real faces as it attains the best PSNR and SSIM results

while improving the PSNR an impressive 1.6 dB over the previous best.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

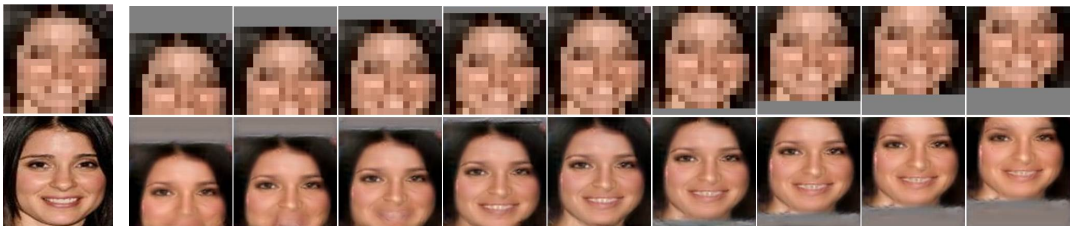
Figure 2.7: Comparison with the state-of-the-art methods on unaligned faces. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Liu et al. [2007]. (e) The method of Yang et al. [2010]. (f) The method of Yang et al. [2013]. (g) The method of Dong et al. [2016a]. (h) The method of Ma et al. [2010]. (g) UR-DGN.

(a) LR,HR (b) UR-DGN

Figure 2.8: Illustrations of influence of occlusions. Top row: the LR inputs, bottom row: the results of UR-DGN. (a) LR and HR images. (b) Results of UR-DGN with occlusions. As seen, occlusions of facial features and landmarks (eyes, mouth, etc.) does not cause any degradation of the unoccluded parts of the faces.

2.7 Limitations

Since we use a generative model to ultra-resolve LR face images, if there are occlusions in the images, our method cannot resolve the occlusions. Still, occlusions of facial features do not adversely affect ultra-resolution of the unoccluded parts as



(a) LR,HR (b) UR-DGN

Figure 2.9: Effects of misalignment. Top row: the LR images, bottom row: the results of UR-DGN. (a) LR and HR images. (b) Results with translations. From left to right, the y-axis translations are from -4 to +4 pixels. Notice that, the size of the LR image is 16×16 pixels. As visible, UR-DGN is robust against severe translational misalignments.

shown in Fig. 2.8.

Our algorithm alleviates the requirements of exact face alignment. As shown in Fig. 2.7 and Fig. 2.9, it is robust against translations, but sensitive to rotations. As a future work, we plan to investigate incorporating an affine transformation estimator and adapting the generative network according to estimated transformation parameters.

2.8 Conclusion

We present a new and very capable discriminative generative network to ultra-resolve very small LR face images. Our algorithm can both increase the input LR image size significantly, *i.e.* $8\times$, and reconstruct much richer facial details. The larger scaling factors beyond $8\times$ only require larger training datasets (e.g., larger than 128×128 training face images for 16×16 inputs), and it is straightforward to achieve even much extreme ultra resolution results.

By introducing a pixel-wise ℓ_2 regularization on the generated face images into the framework of DGN, our method is able to generate authentic HR faces. Since our method learns an end-to-end mapping between LR and HR face images, it preserves well the global structure of faces. Furthermore, in training, we only assume the locations of eyes to be approximately aligned, which significantly makes the other face datasets more attainable.

Chapter 3

Imagining the Unimaginable Faces by Deconvolutional Networks

3.1 Foreword

In chapter 2, we present an ultra-resolution discriminative generative network (URDGN) to hallucinate very low-resolution face images. Since our proposed URDGN also suffers from the training difficulty similar to generative adversarial networks (GANs), the convergence of URDGN might be unstable. In particular, the discriminative network may also introduce artifacts into the generative network. In this chapter, we present a single deconvolutional-convolutional network to ease the training difficulty of URDGN as well as reduce artifacts caused by the deconvolutional layers and the discriminative network in URDGN. Furthermore, we demonstrate that with data augmentation the proposed network is able to upsample rotationally unaligned faces.

This chapter has been published as a journal paper: Xin Yu, Fatih Porikli: Imagining the Unimaginable Faces by Deconvolutional Networks. *IEEE Transactions on Image Processing*, 27(6): 2747-2761, 2018.

3.2 Abstract

We tackle the challenge of constructing 64 pixels for each individual pixel of a thumbnail face image. We show that such an aggressive super-resolution objective can be attained by taking advantage of the global context and making the best use of the prior information portrayed by the image class. Our input image is so small (*e.g.*, 16×16 pixels) that it can be considered as a patch of itself. Thus, conventional patch-matching based super-resolution solutions are unsuitable. In order to enhance the resolution while enforcing the global context, we incorporate a pixel-wise appearance similarity objective into a deconvolutional neural network, which allows efficient learning of mappings between low-resolution input images and their high-resolution counterparts in the training dataset. Furthermore, the deconvolutional network blends the learned high-resolution constituent parts in an authentic manner where the face structure is naturally imposed and the global context is pre-

served. To account for the possible artifacts in upsampled feature maps, we employ a sub-network composed of additional convolutional layers. During training, we use roughly aligned images (only eye locations), yet demonstrate that our network has the capacity to super-resolve face images regardless of pose and facial expression variations. This significantly reduces the requirement of precisely face alignments in the dataset. Owing to the network topology we apply, our method is robust to translational misalignments. In addition, our method is able to upsample rotational unaligned faces with data augmentation. Our extensive experimental analysis manifests that our method achieves more appealing and superior results than the state-of-the-art.

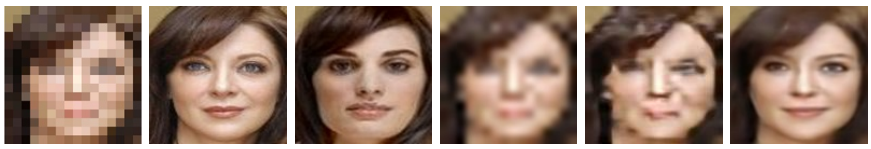
3.3 Introduction

The human face is perhaps the most powerful channel of nonverbal communication. It provides valuable clues to our own feelings and those of the people around us. Even in the most simple interaction, our attention naturally gravitates to the face, seeking to read some of the vital information is "written" there. Faces also play an important role in physical attractiveness.

Naturally, face perception is possible *if* the face is visible in *sufficient* detail and resolution. When the face image is imperceptibly small, its resolution has to be super-resolved with a large upscaling factor. However, conventional super-resolution (SR) methods are mostly limited up to $2 \sim 4\times$ upscaling factors. As reported in [Yang et al., 2014], when the upscaling factor increases to $8\times$, the performance of most SR techniques decreases rapidly, rendering them unsuitable for this challenge.

Existing state-of-the-art SR methods highly rely on a variety of assumptions about the quality of the given low-resolution (LR) image and the availability of an associated set of high-resolution (HR) images. They are applicable only when (i) accurate facial features and landmarks can be found in LR images [Yang et al., 2013; Zhu et al., 2016b], (ii) similar appearances of the "same" person are included in the reference HR dataset [Tappen and Liu, 2012], and (iii) the exemplar HR face images are "densely" aligned in order to derive a representative subspace [Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Yang et al., 2010; Kolouri and Rohde, 2015]. When the input image resolution is inadequately small, the performance of the face SR methods that require detection of precise landmarks for a dense alignment degrades dramatically due to the problematic localization of such refined features and landmark points. This is a consequence of the fact that there is little margin for error or flexibility when the LR image is tiny. Typical pose, facial expression and illumination differences between the input LR image and exemplary HR images hinder the ability of subspace-based face SR methods in capturing local variations and lead to unavoidable ghosting artifacts in the reconstructed HR images.

Several super-resolution methods based on deep neural networks have been proposed [Dong et al., 2016a,b; Kim et al., 2016a; Bruna et al., 2016; Kim et al., 2016a,b; Mao et al., 2016] recently. However, these methods are all patch based and ignore



(a) LR (b) HR (c) NN (d) Bicubic (e) RED (f) Ours

Figure 3.1: Comparison of our method with the CNN based super-resolution. (a) The input 16×16 LR image. (b) The original 128×128 HR image. (c) The corresponding HR version of the nearest neighbor of (a) in the training set. (d) Bicubic interpolation of (a). (e) The image generated by the CNN based super-resolution [Mao et al., 2016]. Notice that, the CNN based approach is further *fine-tuned* with a large corpus of face images. (f) Our result.

image class information. As shown in Fig. 3.1(e), the Convolutional Neural Network (CNN) based network [Mao et al., 2016], even when it has been retrained with face images, fails to produce authentic facial details.

When super-resolving an LR image with an $8\times$ upscaling factor, 98.5% of the original information is missing. Hallucinating such a significant chunk of missing information is an ill-posed problem without a competent regularization term and efficient exploitation of strong priors.

As a solution, we exploit a variant of deconvolutional neural networks [Zeiler and Fergus, 2014] to learn the mappings between the LR facial patterns and HR facial details across individual samples while maintaining the underlying global structure of face images by taking advantage of the collective representation power of large-scale face datasets [Huang et al., 2007; Liu et al., 2015]. Deconvolutional layers, also known as backwards-convolutional layers, are convolutional layers where the forward and backward passes are reversed. In other words, for a stride larger than 1, the output of such a deconvolutional layer has larger resolution than its inputs. They are first utilized in the works [Zeiler et al., 2010; Zeiler and Fergus, 2014] to visualize the features a CNN has learned by back-projecting activations in the low-dimensional feature maps to the high-dimensional image domain. Rather than projecting feature activations to the image domain, Long et al. [2015] use a deconvolutional network to upsample heat maps while Fischer et al. [2015] upsample optical flow fields. However, the upsampling results of these methods tend to be over-smoothed without pronounced high-frequency details. To enhance image details, Shi et al. [2016] present a variant of deconvolutional networks that rearranges multiple LR feature maps into an HR image as its output. These deconvolutional networks do not formulate the super-resolution task on class-specific settings; hence, they fail to model and generate valuable class-specific cues. Furthermore, since our deconvolutional layers are not used for back-projecting activations of feature maps, our method does not require unpooling layers for super-resolution.

Our intuition is that, deconvolutional networks can be trained to generate certain HR image patterns given specific LR activations by presenting the network with a

set of well-structured LR-HR image pairs. Such well-structured data conveniently exists for the face class. Our analysis in Sec. 3.5.3 demonstrates that deconvolutional networks can be trained to recognize particular facial patterns.

In the training stage of our deconvolutional neural network, we feed the *entire* images, *i.e.*, not patches but whole faces, into our network. This allows maintaining the global structure of faces while reconstructing instance specific details. As a result, our deconvolutional network produces realistic HR facial components that seamlessly blend into an HR face image. Since the filters in each layer of our deconvolutional neural network architecture are applied to the entire image, our method achieves robustness to spatial translations and deformations of input faces. For training, we use approximately frontal HR face images that are only aligned at eye locations, which is readily available for most face datasets. We do not make any assumption on facial landmarks and facial expressions.

Overall, our contributions are fourfold:

- We present a novel method to super-resolve with an $8\times$ upscaling factor a very small (16×16 pixels) face image.
- Our method consolidates a deconvolutional network for hallucinating face images. We demonstrate that without using an adversarial loss, our network is still able to super-resolve realistic HR face images and achieves an impressive 1.16 dB PSNR improvement over the state-of-the-art.
- Since only convolution operations are used in our network, our method is not sensitive to translational misalignments, which significantly reduces the accuracy requirement of the face localization in the LR image. This means, even when the face detector response may not be accurate since the face region is very small, our network can still super-resolve it.
- When training our network, we only require approximately frontal and roughly aligned images regardless of pose and facial expression variations, which makes the training datasets more attainable.

3.4 Related Work

Image super-resolution methods aim to magnify an LR image to its HR version that comprises authentic high-frequency details. In general, there are three categories of *generic* super-resolution approaches: interpolation based techniques, image statistics based schemes [Peleg and Elad, 2014; Yang and Yang, 2013] and example/patch based methods [Freeman et al., 2002; Hong Chang et al., 2004; Glasner et al., 2009; Yang et al., 2010; Schulter and Leistner, 2015; Huang et al., 2015]. Interpolation based techniques such as bilinear and bicubic upsampling are computationally efficient. However, they fail to establish high-frequency details since they generate overly smooth edges as the upscaling factor increases. Image statistics based schemes employ image priors to reconstruct HR images with sharper edges, but they are still limited to smaller scaling factors [Lin and Shum, 2006].

Example based methods have the potential to break this limitation. They can be further classified into two groups: internal and external example methods depending on how the reference samples are derived. The first group of methods [Glasner et al., 2009; Freedman and Fattal, 2010; Singh et al., 2014; Huang et al., 2015] exploit self-similarity of patches in the input image. Alternatively, several methods [Freeman et al., 2002; Hong Chang et al., 2004; Yang et al., 2010] aim to learn mappings between LR and HR patches from external reference datasets, and then utilize the learned correspondences to upsample LR images. Nevertheless, when the input image size is very small, it is difficult for internal example based methods to find similar patches across different scales. When the scaling factor is large, it is hard for external example based methods to determine the correct correspondences between LR and HR patches because many different HR patches can correspond to a single LR patch, which induces artifacts at intensity edges.

Recently, many generic super-resolution methods based on deep neural networks have been proposed [Dong et al., 2016a,b; Bruna et al., 2016; Kim et al., 2016a,b; Mao et al., 2016; Shi et al., 2016; Ledig et al., 2017]. For instance, SRCNN [Dong et al., 2016a] applies cascaded convolutional layers to obtain a mapping function between LR and HR patches from a large-scale dataset, while Kim et al. [2016a] learn to upsample the residuals between the HR and interpolated LR patches. To improve the performance of super-resolution without introducing extra parameters of the networks, Kim et al. [2016b] employ recursive convolutional layers to increase the depth of the convolutional layers. Mao et al. [2016] apply symmetric-skip connections between convolutional layers and deconvolutional layers to pass information to the latter layers, thus mitigating the difficulty of training their very deep network. Shi et al. [2016] employ convolutional layers to extract LR features and then rearrange the LR feature maps into HR images by a sub-pixel convolutional layer, which can be considered as a variant of deconvolutional layers. Dong et al. [2016b] use convolutional and deconvolutional layers with smaller filter sizes to speed up SRCNN [Dong et al., 2016a]. Ledig et al. [2017] exploit an adversarial loss and a perceptual loss [Johnson et al., 2016] to obtain more realistic upsampled results. Bruna et al. [2016] extract statistical priors using CNN to regularize the super-resolution process. Since these generic SR methods based on neural networks do not consider class-specific priors, they cannot achieve high performance when they are employed for super-resolving faces. Retraining (fine-tuning) of these networks with face image patches cannot capture the global structure of faces either.

Related to face hallucination, the works in generative adversarial networks (GANs) [Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015] and variational auto-encoders [Kingma and Welling, 2013] exploit neural networks to generate an entirely new image that endows similar properties to the training data distribution, from a random noise input.

Unlike generic SR methods, *class-specific* super-resolution approaches, such as face hallucination [Baker and Kanade, 2000; Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007; Jia and Gong, 2008; Ma et al., 2010; Tappen and Liu, 2012; Yang et al., 2013; Zhou and Fan, 2015; Wang et al., 2014; Kolouri and

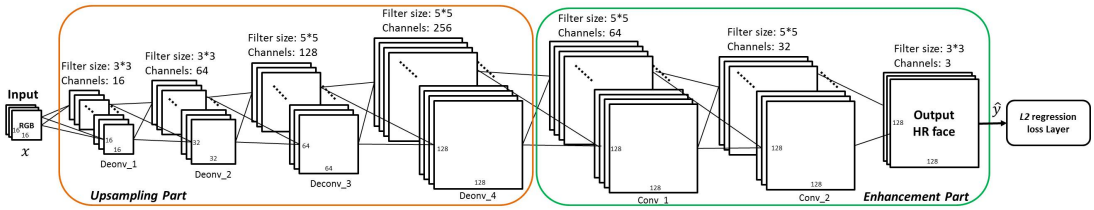


Figure 3.2: Our deconvolutional network consists of two parts: an upsampling part (the orange block) and an enhancement part (the green block).

Rohde, 2015; Jin and Bouganis, 2015; Zhu et al., 2016b; Yu and Porikli, 2016], explore the underlying patterns of a certain class, thus leading to better performance. Baker and Kanade [2000] transfer high-frequency details from a face dataset by building the relationships between LR and HR patches. Due to the possible inconsistency of the transferred HR patches, their method tends to produce artifacts. Eigen-transformation is employed to hallucinate face images by establishing a mapping between the LR and HR face subspaces in [Wang and Tang, 2005]. Similarly, Liu et al. [2007] employ a subspace that is learned from the training set via Principle Component Analysis (PCA) as a linear constraint for HR face images and propose a patch-based Markov Random Field (MRF) to reconstruct the missing high-frequency details. Kolouri and Rohde [2015] use optimal transport in combination with subspace learning to morph an HR image from the LR input. Since the subspace based face hallucination methods require the HR images in the reference dataset to be precisely aligned and the LR test image to have the same pose and facial expression as the reference ones, they are overly sensitive to the misalignments of LR images. In particular, methods that depend on PCA based holistic appearance models suffer from ghosting artifacts.

Rather than imposing global constraints, Ma et al. [2010] construct a super-resolved HR patch by multiple reference HR patches at the corresponding spatial position. Li et al. [2014] model the local structures of faces as a sparse representation problem. Jin and Bouganis [2015] process multiple LR face images to recover an HR image by exploiting a patch-wise mixture of probabilistic PCA prior instead of the holistic PCA prior in [Liu et al., 2007]. Hence, face hallucination methods that constrain the spatial positions of patches may avoid ghosting artifacts caused by PCA, but their performance degrades dramatically when LR image is not aligned precisely to the reference HR images. To handle various poses and expressions, Tappen and Liu [2012] integrate the SIFT flow to align images. By exploiting local patterns, Yang et al. [2013] present a structured face hallucination method. It first detects facial components in the given LR image and then transfers the corresponding HR facial components in the reference dataset to the LR input. Zhu et al. [2016b] present a deep bi-network to super-resolve LR faces. It uses a CNN to localize facial components and then recovers the high-frequency of the localized facial components by another CNN. Nevertheless, these facial component based methods may fail to produce authentic

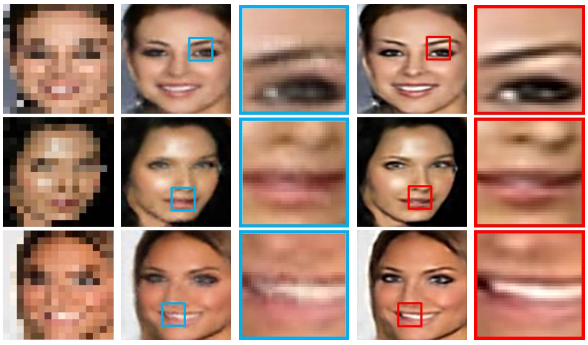


Figure 3.3: Blocking artifacts caused by the deconvolutional layers are effectively removed by the enhancement part. (a) LR input images. (b) Results upsampled only by the deconvolutional layers (the upsampling part). (c) The close-ups of (b). (d) Results upsampled by the entire network. (e) The close-ups of (d).

HR face images due to potentially inaccurate landmark localization. Zhou and Fan [2015] propose a bi-channel CNN to hallucinate face images in wild scenes. Since they require extraction of local features from the input images, the smallest input image size is limited to 48×48 pixels. Yu and Porikli [2016] extend the framework of GAN for very low-resolution face super-resolution. Their follow-up work [Yu and Porikli, 2017b] employs an adversarial loss to distinguish whether super-resolved HR faces are realistic, and uses spatial transformer networks (STN) [Jaderberg et al., 2015] in their deconvolutional networks to compensate for misalignments. When LR face images are aligned and in low noise levels, Yu and Porikli [2017b] super-resolve face images similar to the results of the work [Yu and Porikli, 2016] because they employ similar architectures for upsampling. Due to the sensitive training procedure of GAN, artifacts may appear in the HR outputs; as a result, their high-frequency details may be inconsistent with the ground-truth data.

3.5 Our Face Super-Resolution Network

As shown in Fig. 3.2, our complete network consists of two parts: an upsampling part (deconvolutional), and an image enhancement part (convolutional).

In the upsampling part, we employ deconvolutional layers, as our upsampling part, to super-resolve the LR face images as well as we exploit convolutional layers, as our enhancement part, to remove the blocking artifacts caused by the deconvolutional layers [Odena et al., 2016]. We utilize the ℓ_2 regression loss, also known as the Euclidean distance loss, as the objective of the entire network to attain appearance similarity between the reconstructed images and the original HR images in the training stage.

We first feed the input LR images into a convolutional layer to extract low-level

patterns (features). Since the resolution of input images is very small, *i.e.*, 16×16, the filter size is set to 3×3. The reason for applying a convolutional layer to LR inputs is to mitigate the artifacts introduced by the following deconvolutional layers. As reported in [Odena et al., 2016], a direct application of deconvolutional layers to input images may lead to severe blocking artifacts due to the overlapping regions between the receptive fields. We exploit deconvolutional layers to upsample feature maps, in which most of the activations are close to zero, and thus the artifacts can be mitigated. After the feature extraction, three deconvolutional layers are employed to upsample the feature maps. Each layer upsamples the previous feature maps by an upscaling factor of 2. Since upsampling images is an under-determined problem, we intend to increase the capacity of the network as the neural network goes deeper, *i.e.*, the resolutions of feature maps become larger. Hereby, we double the channel numbers of feature maps of previous layers. The filter sizes of these three deconvolutional layers are 3×3×64, 5×5×128, and 5×5×256, respectively. We apply batch normalization [Ioffe and Szegedy, 2015] after each deconvolutional layer to accelerate the convergence behavior of the network.

Since deconvolutional layers introduce aliasing artifacts in the output images, we incorporate convolutional layers as a subsequent enhancement subnetwork to remove such artifacts. We use three convolutional layers with the filter sizes of 5×5×64, 5×5×32 and 3×3×3 in the enhancement part. We note that Dong et al. [2015] indicate adding more convolutional layers does not suppress artifacts (in their case compression artifacts) but makes the training convergence of the network more difficult. This phenomenon also appears in SRCNN [Dong et al., 2016a], where they show that using more than three layers does not provide a significant improvement in the super-resolution performance. Moreover, a larger network cannot be fed into the GPU memory, either. Hence, we employ a three convolutional layers network to remove aliasing artifacts rather than using a deeper enhancement network.

To illustrate the effectiveness of the two parts of our network, we present the outputs of each part separately in Fig. 3.3. For visualization of the images that are super-resolved only by the upsampling part, we switch the output channel of the last deconvolutional layer to 3 and remove the enhancement part from the entire network. To retrain the upsampling part, we employ the ℓ_2 regression loss between the upsampled images and the HR ground-truth as the object function. As shown in Fig. 3.3(b), the upsampling part generates HR facial details, but the results suffer from the blocking and aliasing artifacts. As shown in Fig. 3.3(d), the artifacts are significantly suppressed, and the facial details are sharpened by the image enhancement part when we train the entire network comprised of the upsampling and enhancement parts. Additionally, the output of the entire network obtains almost 1.3 dB PSNR improvement over the output of the upsampling part on the test dataset. Notice that, the upsampling part produces a total of 256 feature maps.

(a) LR,HR (b) Our results with occlusions (c) Altered

Figure 3.4: Illustrations of influence of occlusions. Top row: the LR images, bottom row: the results of our deconvolutional network. (a) Result without occlusions. (b) Results for partially occluded input images. (c) Result when the upper-lower parts are altered.

3.5.1 Training of the Entire Network

We use LR-HR face image pairs $\{x_i, y_i\}$ as our training data. Since the output of the entire network $\hat{y}_i$ is imposed to be similar to the corresponding HR image y_i, a pixel-wise ℓ_2 regularization term is integrated to induce similarity. The loss E of the complete network for a mini-batch of N face image pairs becomes

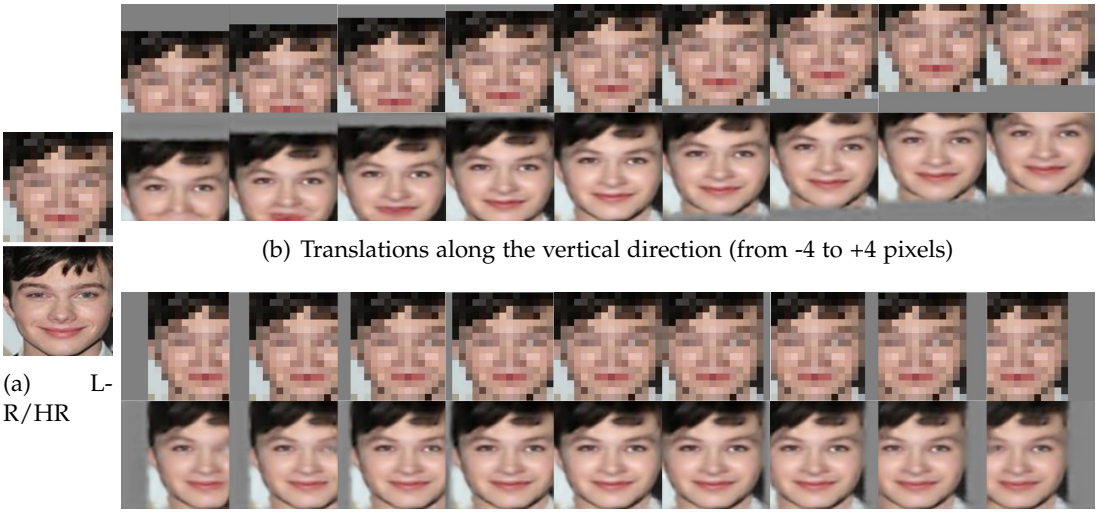
$$E = \frac{1}{ACN} \sum_{i=1}^{N} \|\Phi(x_i) - y_i\|_2^2, \tag{3.1}$$

where $\Phi(x_i) = \hat{y}_i$ denotes the output of the entire network. Here, A and C represent the area and the number of the channels of the training HR images.

The loss E in Eqn. 3.1 is back-propagated to update the parameters of the complete network. Since each layer of our network is differentiable, RMSprop [Hinton, 2012] is used for back-propagation. In RMSprop, we set the learning rate to 10^{-3} and the decay rate α to 0.9. In addition, the learning rate η is multiplied by 0.99 after each epoch.

3.5.2 Super-Resolution of an LR Face Image

We input the LR image x into our network to construct its upsampled HR image $\hat{y}$. In our previous work [Yu and Porikli, 2016], we used a discriminative network to enforce the final results to be similar to typical face images, yet that discriminative network has potential to inject ringing artifacts in the final results. To improve the overall visual quality, we also apply an unsharp filtering [Gonzalez and Wintz, 1977] to the upsampled HR results, which is an image enhancement technique and widely used in low-level image processing tasks, such as super-resolution [Gu et al., 2015] and deblurring [Yu et al., 2014]. Specifically, unsharp filtering is used to generate a sharp image by adding an difference image, which is obtained from subtracting an



(a) L-R/HR

(b) Translations along the vertical direction (from -4 to +4 pixels)

(c) Translations along the horizontal direction (from -4 to +4 pixels)

Figure 3.5: Our method is robust against the translational misalignments of the LR image.

image a blurred version of itself, to the original version. In this way, we preserve the visual fidelity while avoiding the artifacts introduced by the discriminative network.

Since only convolutional operations are used in the network, our end-to-end mapping can maintain the global structure of HR face images while infusing rich and localized details. It is also robust to translational misalignments of LR images. As illustrated in Fig. 3.5, our method can accurately reconstruct the corresponding HR face images even if the LR images are shifted in horizontal and vertical directions.

Thanks to its feed-forward architecture, our method runs in real-time on GPU when it super-resolves an LR image.

3.5.3 What does the Deconvolutional Network Learn?

In our deconvolutional network, the hallucination of the entire face and the formation of individual facial components are implemented seamlessly. To dissect what our deconvolutional network learns, we apply a set of masks to occlude different parts and facial components of the input image. Our assumption here is that a holistic face model based neural network can still generate a complete face without missing parts, even if the reconstructions of the originally occluded parts may be not realistic. Otherwise, it is more likely that the network learns face components.

Figure 3.4 suggests that our deconvolutional network learns facial components and their relative local arrangements. Figure 3.4(b) shows that the visible parts of the input images are super-resolved well while the masked parts are not recovered. Even when we switch the upper and lower parts of the face as shown in Fig. 3.4(c), which does not look like a face, the corresponding parts can be super-resolved by our network. As presented in Fig. 3.5, our network can reconstruct the translated

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 3.6: Comparison with *fine-tuned* SRCNNs [Dong et al., 2016a] and REDs [Mao et al., 2016]. (a) The LR image. (b) The original HR image. (c) Result of the original SRCNN applying an upscaling factor of 2× three times. (d) Result of the SRCNN fine-tuned and retrained with whole face images. (e) Result of the SRCNN retrained with patches with an upscaling factor of 8×. (f) Result of the original RED applying an upscaling factor of 2× three times. (g) Result of the RED fine-tuned and retrained with whole face images. (h) Result of the RED retrained with patches with an upscaling factor of 8×. (i) Our result.

versions of the HR face images consistent with the LR face images when the input face undergoes large translations. This also indicates that our network learns the facial components rather than a rigid holistic face model, and generates HR facial components given specific LR facial patterns.

3.5.4 Differences Between Our Network and CNN based Nets

One major difference between our network and CNN based super-resolution networks, such as SRCNN [Dong et al., 2016a] and RED [Mao et al., 2016], lies on the network architecture. Our method employs deconvolutional layers for upsampling LR face images, while CNN based super-resolution networks apply convolutional layers. For instance, SRCNN and RED firstly upsample the input LR patches by bicubic interpolation and then use convolutional layers to enhance the corresponding details of the interpolated LR patches. Since the corresponding receptive fields of the filters in the HR images are just the same as the filter sizes, only local information is incorporated in the generated high-frequency details. As shown in Fig. 3.6(c), when SRCNN is directly applied to the face hallucination task, the output HR face image is severely blurred due to the small size of the input image and the large upscaling factor. The same phenomenon for the RED can be seen in Fig. 3.6(f) as well.

Another difference is that generic super-resolution methods [Dong et al., 2016a; Mao et al., 2016] are patch based while our method uses the entire image. Since SRCNN released its training code, we can compare its variants more objectively. To achieve the most objective comparison, we not only assess the performance of the original SRCNN but also its possible adaptations for face hallucination. The original SRCNN does not provide a direct upscaling factor of 8× but requires 2× upsampling of the input image three times. When sequentially upsampling, facial components that appear in different scales cannot be learned by the original SRCNN. Hence, we first retrain SRCNN with face patches with an upscaling factor 8×. We use the same architecture and hyperparameters of SRCNN and retrain the network

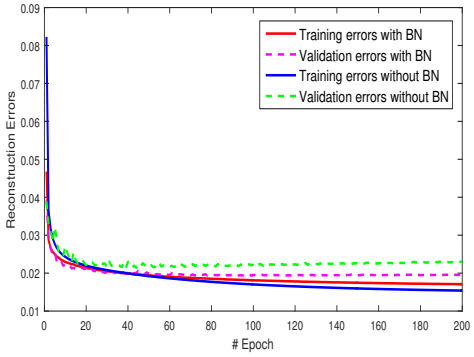


Figure 3.7: Comparisons of the training and validation errors with and without using batch normalization.

by using face patches with the scaling factor 8×. As shown in Fig. 3.6(e), SRCNN cannot produce an HR face image with authentic high-frequency details. Because the scaling factor is large, the interpolated LR images are too smooth for SRCNN to manage. In other words, local neighbors provide little information in enhancing the details. Moreover, when retraining SRCNN with entire face images, the large size of training patches, *i.e.*, 128×128 pixels, introduces more ambiguity in learning of the parameters, compared with the 33×33 pixels patch size the original SRCNN employs. During the training, the weights of SRCNN gets stuck into erroneous local minima and decrease to zero, thus produce a zero-valued image. As shown in Fig. 3.6(d), the SRCNN retrained with entire face images fails to provide high-quality HR face images.

One factor that affects the super-resolution performance is the depth of neural networks. Since SRCNN only has 3 convolutional layers, its performance may be limited. We also compare with another CNN based method, RED, which consists of 15 convolutional layers and 15 deconvolutional layers, much deeper than our network and trained on image patches of size 50×50 pixels. Note that, the deconvolutional layers employed in RED are different from our deconvolution layers; the deconvolutional layers in RED only implement backward convolutional operations without increasing the output resolutions. To tackle the vanishing gradient problem and obtain an efficient training scheme, RED passes information from the convolutional layers to their corresponding deconvolutional layers by exploiting skip connections. Similar to SRCNN, RED firstly upsamples inputs by bicubic interpolation and then enhances details. As shown in Fig. 3.6(f), directly applying RED to the LR face by an upscaling factor 2× three times cannot achieve realistic facial details, *e.g.*, the LR eye regions only consist of dark colors. It only enhances edges and textures rather than generating semantically new pixels, such as the white color in the eyeballs. As presented in Fig. 3.6(h), retraining RED with face patches by an upscaling factor 8× cannot obtain authentic facial details since the large upscaling factor introduces severe ambiguity between LR and HR patches. We also retrain RED with the whole face images as well as the same training protocol that we use. As seen in Fig. 3.6(g), RED fails to

generate realistic facial details; instead, it outputs ringing artifacts. Hence, simply increasing the depth of convolutional networks cannot super-resolve LR faces either.

In contrast to SRCNN and RED, our deconvolutional network upsamples the LR face images gradually without any bicubic interpolation. This strategy can be regarded as leveraging the image pyramid to address the under-determined task of $8\times$ super-resolution. In a hierarchical manner, we hallucinate facial details, thus mitigating the ambiguity between LR and HR face images. In contrast, bicubic interpolation employed in CNN based super-resolution methods cannot reduce the ambiguity between the interpolated LR and HR faces since it only relies on upsampling of pixels without any hallucination. Furthermore, the receptive field of the filters of our first deconvolutional layer is 24×24 pixels in the HR images, which is much larger than the largest receptive field of the filters in SRCNN, *i.e.*, 9×9 pixels. As a result, our network can better capture LR facial patterns, and it can access expanded spatial neighborhood to generate HR faces. Our deconvolutional layers are able to project the low-dimensional feature maps to the high-dimensional image domain and the learned feature patterns are embedded in the weights of the network. Hence, our deconvolutional network is more suitable to construct a mapping from LR face images to their HR versions.

Generic CNN based super-resolution methods, such as SRCNN and RED, do not incorporate batch normalization either. Batch normalization is originally invented to reduce internal covariate shift by whitening feature maps and widely used for classification tasks. Since batch normalization will change the intensity distributions of feature maps in each layer, it may distort the mapping relationships between LR and HR patches in the super-resolution problem. Specifically, generic CNN based SR methods construct a nonlinear mapping between different LR and HR patches on image intensities. Considering the intensity distributions of different patches may vary dramatically, the distributions of their corresponding feature maps in each layer would be significantly different because image patches are not normalized when they are fed into super-resolution networks. Thus, the mean and variance for each layer vary in a mini-batch. Using a statistical mean and variance to normalize the feature maps in each layer will shift activations of input patches. This effect will increase the ambiguity in super-resolution, thus increasing the training loss. As a result, the intensity of the reconstructed HR patches would be distorted. This phenomenon that embedding batch normalization into the CNN based super-resolution, *e.g.*, SRCNN and VDSR [Kim et al., 2016a], degrades the super-resolution performance is also observed in the very recent works [Ren et al., 2017; Yang et al., 2017b]. Therefore, it is not suitable to use batch normalization in generic patch based super-resolution convolutional networks.

Since our inputs are class-specific, the feature maps share similar distributions in each layer. Using batch normalization allows speeding up the training phase without shifting the reconstructed faces in our network. In Fig. 3.7, we compare the training errors with and without using batch normalization. As seen in the first 50 epochs, our network achieves lower training and validation errors by using batch normalization. It indicates that batch normalization speeds up the learning process of our network.

Even though after 50 epochs the training errors of the network without using batch normalization become lower than the one using batch normalization, their validation errors stop decreasing and the validation errors of the network without using batch normalization are higher than the one using batch normalization. It implies that batch normalization facilitates the generalization ability of our network.

3.6 Experimental Analysis

We compare our method with a large set of eleven state-of-the-art methods [Liu et al., 2007; Yang et al., 2010, 2013; Dong et al., 2016a; Ma et al., 2010; Kim et al., 2016a,b; Mao et al., 2016; Jin and Bouganis, 2015; Zhu et al., 2016b; Yu and Porikli, 2016] both qualitatively and quantitatively. Liu et al. [2007] employ a subspace based face hallucination method. Yang et al. [2010] use sparse representations to super-resolve HR images by constructing LR and HR dictionaries. The method in [Yang et al., 2013] hallucinates face images by using facial components from an exemplar image dataset while CBN [Zhu et al., 2016b] super-resolves facial components by deep cascaded bi-networks. SRCNN [Dong et al., 2016a], VDSR [Kim et al., 2016a], DRCN [Kim et al., 2016b], and RED [Mao et al., 2016] apply CNNs to upsample images. Ma et al. [2010] use the same position reference patches to reconstruct HR images. Jin and Bouganis [2015] exploit multiple LR faces to recover an HR version by a patch-wise mixture of probabilistic PCA prior (MPPCA).

3.6.1 Datasets

Our network is trained on the Celebrity Face Attributes (CelebA) dataset [Liu et al., 2015]. There are more than 200K face images in this dataset where only the similarity transformation is employed to align the locations of eye centers [Liu et al., 2015]. The images cover different pose variations and facial expressions. We simply use all available data regardless of these variations and do not require grouping the face images into different pose and facial expression subcategories.

We randomly select 30K cropped face images from the CelebA dataset, and then resize them to 128×128 pixels as HR images. We downsample the HR face images to 16×16 pixels to obtain the LR counterparts. We use 29K images for the training, 1K images for validation and 1K images for testing.

Our network never sees the test LR images in the training phase. The test and training images are substantially different. To illustrate this, we find the best matching LR image in the training data for a random input test LR image. As shown in Fig. 3.1, the corresponding HR version of the best match has significant differences from the original HR version of the LR test image.

3.6.2 Qualitative Comparisons

We perform side-by-side comparisons with eleven state-of-the-art face hallucination methods. In case an approach does not allow an 8× scaling factor directly, *e.g.*, [Yang

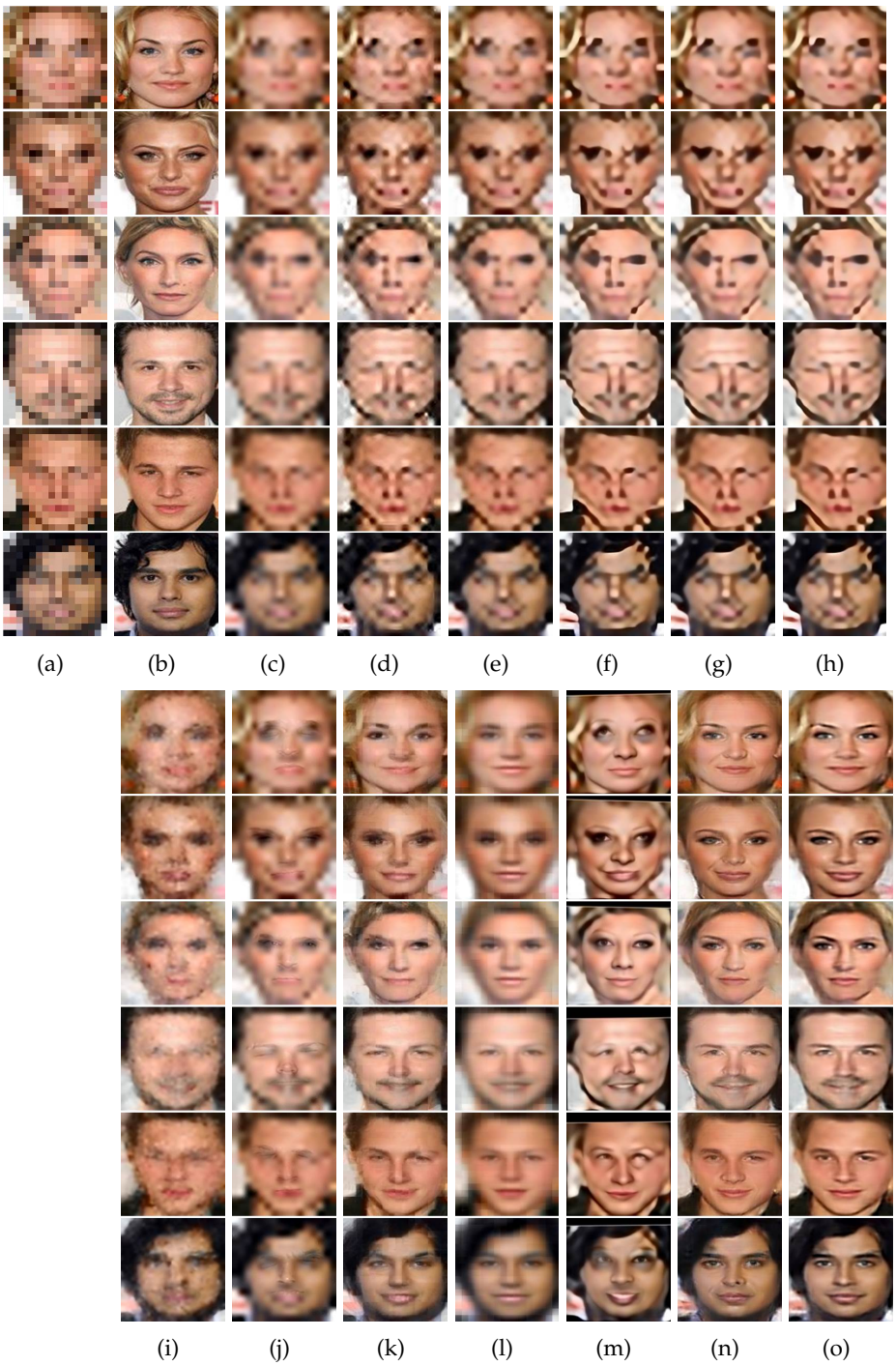


Figure 3.8: Comparison with the state-of-the-art on **frontal** face images. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Yang et al. [2010]. (e) The method of Dong et al. [2016a] (SRCNN). (f) The method of Kim et al. [2016a] (VDSR). (g) The method of Kim et al. [2016b] (DRCN). (h) The method of Mao et al. [2016] (RED). (i) The method of Liu et al. [2007]. (j) The method of Yang et al. [2013]. (k) The method of Ma et al. [2010]. (l) The method of Jin and Bouganis [2015] (MPPCA). (m) The method of Zhu et al. [2016b] (CBN). (n) The method of Yu and Porikli [2016] (URDGN). (o) Our method. (Please see the electronic version for fine-grained details)

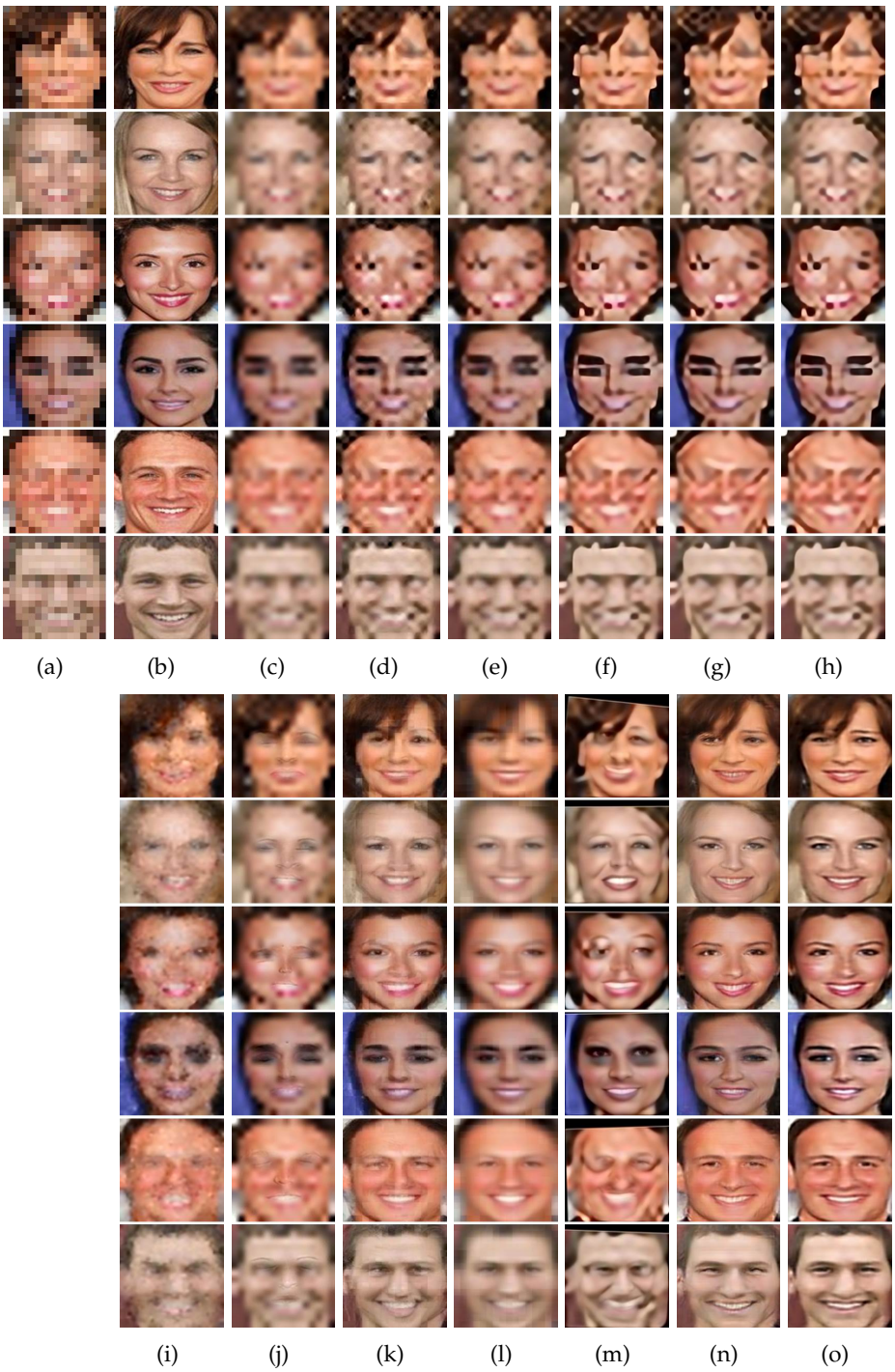


Figure 3.9: Comparison with the state-of-the-art on images with **facial expressions**. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Yang et al. [2010]. (e) The method of Dong et al. [2016a] (SRCNN). (f) The method of Kim et al. [2016a] (VDSR). (g) The method of Kim et al. [2016b] (DRCN). (h) The method of Mao et al. [2016] (RED). (i) The method of Liu et al. [2007]. (j) The method of Yang et al. [2013]. (k) The method of Ma et al. [2010]. (l) The method of Jin and Bouganis [2015] (MPPCA). (m) The method of Zhu et al. [2016b] (CBN). (n) The method of Yu and Porikli [2016] (URDGN). (o) Our method.

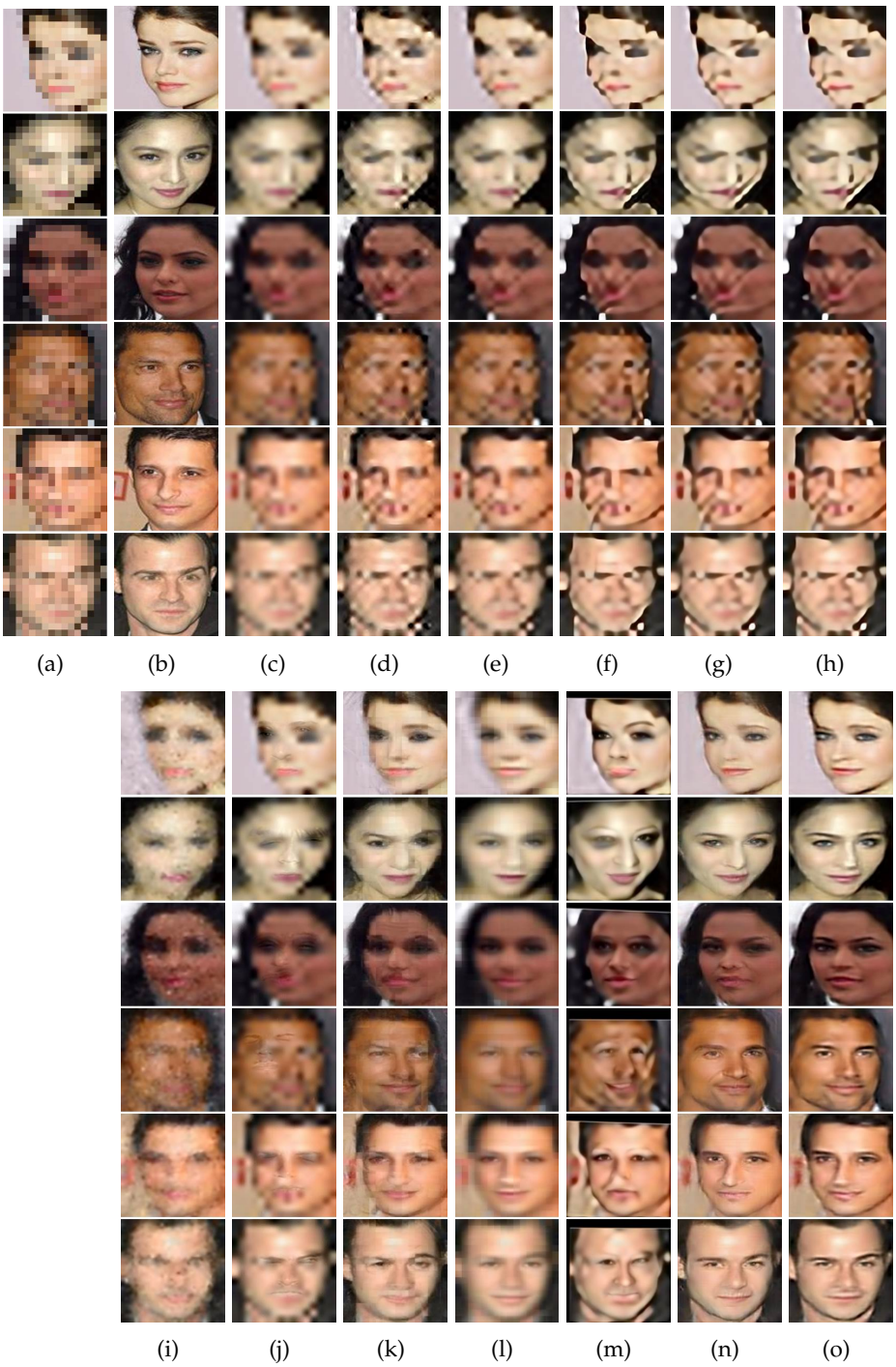


Figure 3.10: Comparison with the state-of-the-art on **different pose** face images. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Yang et al. [2010]. (e) The method of Dong et al. [2016a] (SRCNN). (f) The method of Kim et al. [2016a] (VDSR). (g) The method of Kim et al. [2016b] (DRCN). (h) The method of Mao et al. [2016] (RED). (i) The method of Liu et al. [2007]. (j) The method of Yang et al. [2013]. (k) The method of Ma et al. [2010]. (l) The method of Jin and Bouganis [2015] (MPPCA). (m) The method of Zhu et al. [2016b] (CBN). (n) The method of Yu and Porikli [2016] (URDGN). (o) Our method.

et al., 2010; Dong et al., 2016a; Kim et al., 2016a,b; Mao et al., 2016], we repeatedly apply a scaling factor of 2× three times. For fair comparisons, we use the same CelebA dataset for the training of all other algorithms. As another baseline, we present the bicubic interpolation results.

Comparison with Yang *et al.*'s method: As depicted in Fig. 3.8(d), Fig. 3.9(d), Fig. 3.10(d) and Fig. 3.11(d), Yang *et al.*'s method does not recover high-frequency facial details. Besides, irregular over-emphasized edge artifacts appear in the results. As the scaling factor increases, the correspondence between LR and HR patches becomes ambiguous. Therefore, the results suffer from exaggerated pixelation patterns.

Comparison with Dong *et al.*'s method: SRCNN applies convolutional layers to learn a generic patch-based mapping function. Even though we retrain their CNN on face images, SRCNN cannot generate high-frequency facial details in the HR images as shown in Fig. 3.8(e), Fig. 3.9(e), Fig. 3.10(e) and Fig. 3.11(e). This demonstrates that our deconvolutional network is more suitable to address the face hallucination task. In contrast to SRCNN, our deconvolutional network incorporates class-specific information to induce fine-grained patterns authentic to faces, thus leads to better performance.

Comparison with Kim *et al.*'s method: Kim *et al.* propose a very deep convolutional network for generic image super-resolution, known as VDSR, where they increase the number of the convolutional layers to 20 while SRCNN uses only 3. To accelerate the training of its network, VDSR learns the high-frequency residuals between the upsampled input patches and their HR ground truths instead of producing HR patches directly. Similar to SRCNN, VDSR also firstly upsamples LR input patches by bicubic interpolation and then reconstructs high-frequency details by a deep CNN. As shown in Fig. 3.8(f), Fig. 3.9(f) and Fig. 3.10(f), VDSR fails to output realistic facial details and over-enhances edges of the upsampled LR facial patterns. This also indicates that just increasing the depth of traditional convolutional networks may not necessarily generate authentic facial details.

Comparison with Kim *et al.*'s method: Kim *et al.* develop a deeply recursive convolutional network (DRCN) to super-resolve generic images. DRCN employs 16 recursive convolutional layers followed by ReLU layers to increase the super-resolution performance without introducing extra parameters. Similar to VDSR, the high-frequency residuals are learned from the neural network. As shown in Fig. 3.8(g), Fig. 3.9(g) and Fig. 3.10(g), DRCN over-emphasizes edges and cannot hallucinate authentic high-frequency facial textures, *i.e.*, eyes and mouths. In contrast, our network can reconstruct realistic facial details.

Comparison with Mao *et al.*'s method: Mao *et al.* employ a very deep residual encoder-decoder network to upsample images, named as RED, which has 15 convolutional and 15 deconvolutional layers to recover the missing high-frequency contents in LR patches. Different from our deconvolutional layers, the deconvolutional layers in RED do not increase the resolution of feature maps. RED is a patch-based generic super-resolution method, and it is trained with generic image patches. As shown in Fig. 3.8(h), Fig. 3.9(h), Fig. 3.10(h), Fig. 3.11(d) and Fig. 3.12(c), RED cannot produce authentic HR face images either. Hence, we conclude that directly upsam-

pling LR inputs by bicubic interpolation and then generating image details from the interpolated images by CNNs is not suitable for the face hallucination task.

Comparison with Liu *et al.*'s method: Since Liu *et al.*'s method requires the face images in the dataset to be precisely aligned, it is difficult for their method to learn a representative subspace from the CelebA dataset where large variations exist. Therefore, the global model of the input LR image cannot be represented by the learned subspace, and its local model leads to patchy artifacts in the results. As shown in Fig. 3.8(i), Fig. 3.9(i) and Fig. 3.10(i), this method cannot recover face details correctly, and noisy artifacts appear in the final results.

Comparison with Yang *et al.*'s method: This method requires landmarks of facial components. It reconstructs LR images by transferring high-resolution facial components. In a 16×16 input image, it is extremely difficult to localize landmarks. Hence, this method cannot correctly transfer facial components as shown in Fig. 3.8(j), Fig. 3.9(j), Fig. 3.10(j) and Fig. 3.12(d). Moreover, as seen in Fig. 3.11(e), facial details cannot be recovered either due to the very large upscaling factor. To our advantage, our method does not need landmark localization and still preserves the global structure of the faces.

Comparison with Ma *et al.*'s method: This method requires the reference images to be precisely aligned. As shown in Fig. 3.8(k), Fig. 3.9(k) and Fig. 3.10(k), it suffers from obvious blocking artifacts and uneven over-smoothing as a result of unaligned reference patches in the training dataset and the large scaling factor. As illustrated in Fig. 3.11(h) and Fig. 3.12(g), this method mixes the magnified input face with a reference positioned ghost face due to translational and rotational misalignments. Our method, on the other hand, can still upsample the misaligned LR face images with rich high-frequency details.

Comparison with the method of Jin and Bouganis: Instead of generating a holistic face model by PCA, this method, also known as MPPCA, super-resolves each patch of an LR face by exploiting a prior of the mixture probabilistic principal component analysis [Tipping and Bishop, 1999]. MPPCA uses multiple LR images to recover an HR face. As reported in their experimental part, MPPCA utilizes multiple LR images synthesized from a single HR image to evaluate its performance. Hence, following its experimental protocol, we also generate multiple LR faces from an HR ground-truth image and then apply MPPCA to reconstruct the HR face. Because MPPCA needs to estimate the motion transformations between LR images, any error in transformation parameter estimation causes reconstruction errors. To prevent from this, we use the ground-truth motion transformation parameters to align LR images in our experiments. Since each pixel of the LR inputs corresponds to an MPPCA model and the upscaling factor is large, *i.e.*, $8\times$, inconsistency may appear along the boundaries of generated HR patches. As seen in Fig. 3.8(l), Fig. 3.9(l), Fig. 3.10(l), Fig. 3.11(g) and Fig. 3.12(f), MPPCA suffers visible blocking artifacts and produces overly smooth HR faces due to the large upscaling factor.

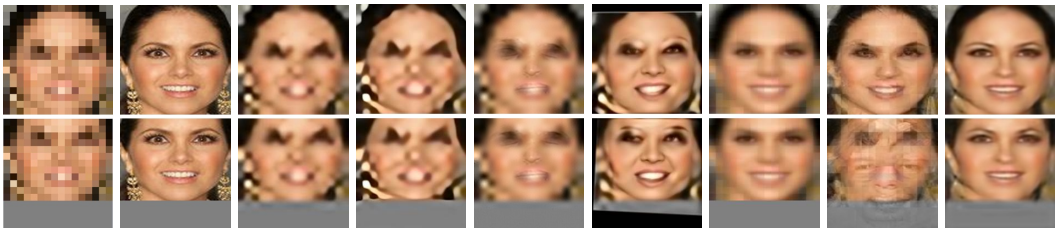
Comparison with Zhu *et al.*'s method: Zhu *et al.*'s method, called as CBN, first detects the facial components and then applies a deep neural network to super-resolve facial components. Since the resolution of the input faces is very small, it

Table 3.1: Quantitative evaluation on the entire test dataset

Methods	PSNR	SSIM
Bicubic	23.15	0.67
[Yang et al., 2010]	21.29	0.60
[Dong et al., 2016a]	22.25	0.65
[Kim et al., 2016a]	20.17	0.58
[Kim et al., 2016b]	20.75	0.60
[Mao et al., 2016]	20.11	0.58
[Liu et al., 2007]	21.54	0.55
[Yang et al., 2013]	23.05	0.66
[Ma et al., 2010]	23.09	0.64
[Jin and Bouganis, 2015]	22.96	0.64
[Zhu et al., 2016b]	20.27	0.58
[Yu and Porikli, 2016]	23.88	0.71
Ours$^-$	24.39	0.72
Ours	**25.04**	**0.74**

is difficult to detect and localize facial components accurately. Such errors directly lead to ghosting artifacts. As illustrated in Fig. 3.8(m), Fig. 3.9(m) and Fig. 3.10(m), CBN fails to output authentic HR faces when erroneous localization of the LR facial components occurs. As shown in Fig. 3.11(f) and Fig. 3.12(e), the upsampled facial details are inconsistent with the LR faces. CBN firstly aligns the LR inputs to its predefined coordinates and then generates high-frequency details. When we transform the hallucinated faces back onto the original coordinates, the black regions appear in the final results.

Comparison with the method of Yu and Porikli: Yu and Porikli's method, also known as URDGN, exploits the framework of the generative adversarial network [Goodfellow et al., 2014] to super-resolve HR faces. Its discriminator network enforces the generated HR face images to be similar to the real ones, but it may also introduce artifacts and thus distorts the hallucinated facial details. As shown in Fig. 3.8(n), Fig. 3.9(n) and Fig. 3.10(n), although the results of URDGN are sharp, the high-frequency details may not comply with the HR ground-truth as indicated in the quantitative evaluation. In contrast, our method can recover facial details more faithfully to the ground-truth faces. Note that the artifacts caused by deconvolutional layers as well as the adversarial loss are not suppressed by URDGN while they are significantly reduced by our convolutional layers. Furthermore, URDGN employs the procedure of generative adversarial networks (GAN) to train its entire network, and it is difficult to maintain the balance between the generative and discriminative networks. Thus, the convergence of URDGN is not as stable as our method.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 3.11: Comparison with the state-of-the-art on translational **misaligned** face images. (a) LR inputs. (b) Original HR images. (c) The method of Dong et al. [2016a] (SRCNN). (d) The method of Mao et al. [2016] (RED). (e) The method of Yang et al. [2013]. (f) The method of Zhu et al. [2016b] (CBN). (g) The method of Jin and Bouganis [2015] (MPPCA). (h) The method of Ma et al. [2010]. (i) Our method.

3.6.3 Quantitative Comparisons

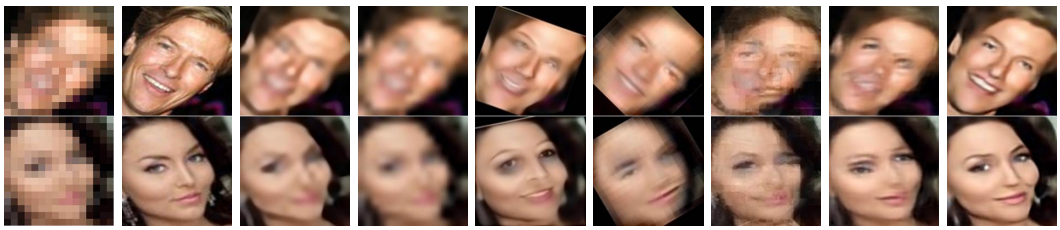
We also measure the performance by the average PSNR and the structural similarity (SSIM) scores on the entire test dataset. Table 3.1 shows that our method achieves the best performance with an impressive 1.16 dB PSNR improvement. In Tab. 3.1, we also compare the PSNR and SSIM scores without using batch normalization, as indicated by Ours^{-}. Benefiting from batch normalization, our method is able to achieve higher PSNR and SSIM scores.

Notice that, the bicubic interpolation explicitly builds on pixel-wise intensities without any hallucination, and attains better performance than several state-of-the-art methods. This implies that either the high-frequency details reconstructed by the state-of-the-art methods are *not* authentic or the artifacts caused by those methods severely degrade their quantitative results.

Unlike the existing approaches, our method consistently provides visually appealing super-resolved HR face images that contain rich details, and at the same time, exhibit close similarity to the original ones (not used in the training). Since our method takes the input LR image as a whole and learns facial components in a data-driven manner, it reduces the ambiguity of the correspondence between LR and HR patches, leading to superior results both qualitatively and quantitatively.

3.6.4 Sensitivity to Translational Misalignments

Since the low-resolution of the input face images is very small, state-of-the-art face detectors may not localize the face precisely. In particular, when the translational alignments occur, the previous face hallucination methods may fail as seen in Fig. 3.11. By contrast, our method is able to upsample the LR face images without any degradation. In our method the translational alignment requirement is significantly relaxed. Even when the face detector fails to localize LR faces accurately, our method can still upsample the face images that have the similar sizes as the faces in



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 3.12: Comparison with the state-of-the-art on rotational **misaligned** face images. (a) LR inputs. (b) Original HR images. (c) The method of Mao et al. [2016] (RED). (d) The method of Yang et al. [2013]. (e) The method of Zhu et al. [2016b] (CBN). (f) The method of Jin and Bouganis [2015] (MPPCA). (g) The method of Ma et al. [2010]. (h) Our method. (i) Our method with rotated face augmentation.

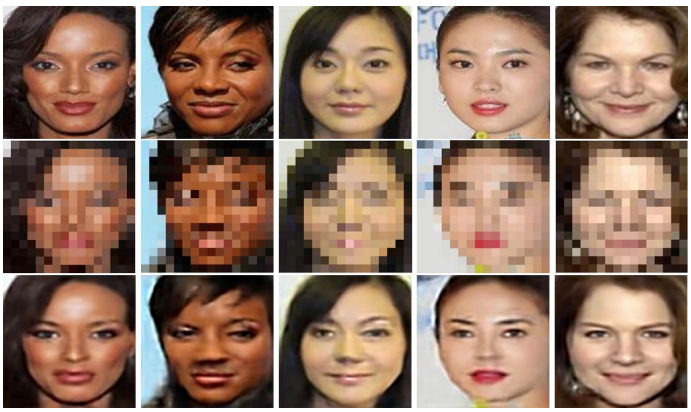


Figure 3.13: Our method can hallucinate face images regardless of the racial profiles of the input images. Top row: the original HR face images. Middle row: the input LR face images. Bottom row: our results.

the training dataset.

3.6.5 Sensitivity to Rotational Misalignments

As shown in Fig. 3.5 and Fig. 3.11, our method significantly reduces the requirement of face alignment, in particular, it can tolerate the translational misalignments of LR face images. Having said that, our network is trained with only upright face images; thus its performance would decrease when LR face images undergo large rotations as shown in Fig. 3.12(h). The rotated facial parts are not explicitly learned in the training stage. Therefore, our network may not recognize the corresponding low-dimensional features. As a result, we crop the HR faces from CelebA, randomly rotate HR faces and then downsample the HR faces to 16×16 pixels as LR faces. We augment our training and testing datasets with the rotated faces and then retrain our

Figure 3.14: Hallucinating face images with eyeglasses. Top row: the input LR face images. Bottom row: our results.

network on the augmented dataset. Notice that, the ground-truth images may not be upright due to the data augmentation. As shown in Fig. 3.12(i), our method can super-resolve LR faces with rotational misalignments as well.

3.6.6 Face Super-Resolution without a Face Detector

In Fig. 3.15, we present an example where the face region in the LR image is directly super-resolved without a face detector, *i.e.*, the face region is not detected and cropped before it is applied to our network. As visible, the face region is restored with sufficient and pleasant high-frequency facial details while the background regions are also upsampled without artifacts. Our method can efficiently remove the blocking artifacts along the edges in the background. In comparison, the CNN based super-resolution not only fails to generate authentic facial features such as mouth and eyes but also injects faulty checkerboard patterns (around fingers, hair, etc.) and overemphasized edges (around the black dress).

This example demonstrates that our deconvolutional network allows generating high-frequency details for faces without creating artifacts in the generic regions. Our method can recognize and super-resolve the LR facial features regardless of the locations of the features. We can upsample the LR faces without using a face detector when the LR faces approximately have the size of 16×16 pixels while the existing face hallucination methods rely on face detectors to crop faces in advance.

3.6.7 Different Racial Profiles

When training our deconvolutional network, we do not partition the training face images into different training sets based on their racial profiles. Instead, we use all available face images. We observe that our network can still conceive the shared characteristics of each race and upsample LR input images without requiring different models for different races. In other words, our method does not need a face attribute for the input image. As shown in Fig. 3.13, our method can super-resolve while maintaining the original racial profiles without mixing different racial characteristics.

(a) LR (b) SRCNN (c) Ours

Figure 3.15: Hallucinating face images without detecting and cropping faces. (a) The input LR image. (b) The result of SRCNN. (c) Our result. Note that the face region upsampled by our method contains much richer high-frequency details, such as the eyes and mouth. (please see the electronic version for details)

3.6.8 Glasses

There are three cases around the super-resolution of faces with eyeglasses. The first one is the people wearing sunglasses, as shown in the first column of Fig. 3.14. In this case, eyes are occluded by the sunglasses. Obviously, the eyes cannot be super-resolved while the other facial parts including the sunglasses can be well reconstructed. The second case is that the frames of eyeglasses are thin and invisible in the small LR images. Since the eyeglasses are not visible, they cannot be reconstructed in the HR outputs, *i.e.*, the eyeglasses will not affect the face super-resolution. Lastly, the frames of eyeglasses might be thick enough to be hinted in the small LR images. Since the resolution of the LR image is only 16×16 pixels, the pixels corresponding to the eyeglasses frames are blended with the pixels of the eyes (the last column of Fig. 3.14). This may introduce some degradation around the upsampled eyes yet the rest of the face is well hallucinated. Since the styles and colors of eyeglass frames vary remarkably, using a proportionally larger dataset of annotated training images with eyeglasses can provide a remedy. However, this may not be practical.

3.6.9 Training Dataset Bias

In the CelebA dataset, the most common facial expression is the smile, which constitutes 48.2 percent of all faces in the dataset. This is the reason that in Fig. 3.9 most of the samples have smiling expressions. Although there are other expressions in the dataset, they do not exist in sufficient numbers to train a deep neural network. Given enough training samples, our deconvolutional network can be devised to hallucinate any facial expressions.

3.6.10 Limitations

Since our deconvolutional network hallucinates facial parts from a very low-resolution face image and then assembles them in an authentic manner into an HR face image, our method does not generate a complete face image when some regions are occluded in the LR input image. Nevertheless, as shown in Fig. 3.4(b), such occlusions do not degrade the super-resolution performance of the visible parts.

3.7 Conclusion

We presented an effective method to super-resolve very small LR face images by exploiting deconvolutional neural networks. Our method increases the input LR image size significantly, *i.e.*, $8\times$, and reconstructs rich facial details. Since it learns an end-to-end mapping between the LR and HR face images and uses only convolutional operations, it preserves the global structure of faces while mitigating the alignment requirements of LR inputs. Due to the simple feed-forward network architecture, our method runs in real-time.

Chapter 4

Face Hallucination with Tiny Unaligned Images by Transformative Discriminative Neural Networks

4.1 Foreword

In chapter 2 and chapter 3, our ultra-resolution discriminative generative network (URDGN) as well as deconvolutional network are only designed to hallucinate very low-resolution aligned face images. Since the networks are composed of deconvolutional and convolutional layers, they are inherently robust to translational misalignments but not to rotational misalignments. In this chapter, we present a transformative discriminative network to super-resolve unaligned low-resolution input images while aligning them automatically. Since spatial transformer networks are able to align images or image regions, we embed them into our upsampling network to compensate for the misalignments in the low-resolution face images in the process of super-resolution.

This chapter has been published as a conference paper: Xin Yu, Fatih Porikli: Face Hallucination with Tiny Unaligned Images by Transformative Discriminative Neural Networks. In *The Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*, 4327-4333, 2017.

4.2 Abstract

Conventional face hallucination methods rely heavily on accurate alignment of low-resolution (LR) faces before upsampling them. Misalignment often leads to deficient results and unnatural artifacts for large upscaling factors. However, due to the diverse range of poses and different facial expressions, aligning an LR input image, in particular when it is tiny, is severely difficult. To overcome this challenge, here we present an end-to-end transformative discriminative neural network (TDN) devised for super-resolving unaligned and very small face images with an extreme upscaling

factor of 8. Our method employs an upsampling network where we embed spatial transformation layers to allow local receptive fields to line-up with similar spatial supports. Furthermore, we incorporate a class-specific loss in our objective through a successive discriminative network to improve the alignment and upsampling performance with semantic information. Extensive experiments on large face datasets show that the proposed method significantly outperforms the state-of-the-art.

4.3 Introduction

Face images provide vital information for visual perception and identity analysis. Nonetheless, when the resolution of the face image is very small (*e.g.* in typical surveillance videos), there is little information that can be inferred from it. Very low-resolution (LR) face images not only degrade the performance of the recognition systems but also impede human interpretation. This challenge motivates the reconstruction of high-resolution (HR) images from given LR counterparts, known as face hallucination, and attracts increasing interest in recent years.

Previously proposed face hallucination methods based on holistic appearance models [Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007; Hennings-Yeomans et al., 2008; Ma et al., 2010; Yang et al., 2010; Li et al., 2014; Arandjelović, 2014; Kolouri and Rohde, 2015] demand LR faces to be precisely aligned beforehand. However, aligning LR faces to appearance models is not a straightforward task itself, and more often, it requires expert feedback when the input image is small. Pose and expression variations that naturally exist in LR face images hinder the accuracy of automatic alignment techniques, which usually assume facial landmarks are visible and detectable. As a result, the performance of face hallucination degrades severely. Such a broad spectrum of pose and expression variations also makes learning a comprehensive appearance model even harder. For instance, Principal Component Analysis (PCA) based schemes become critically ineffective to learn a reliable face model while aiming to capture different in- and out-plane rotations, scale changes, translational shifts, and facial expressions. As a result, these methods lead to unavoidable artifacts when LR faces are misaligned or depict different poses and facial expressions from the base appearance model.

Rather than learning holistic appearance models, many methods upsample facial components by transferring references from an HR training dataset and then blending them into an HR version [Tappen and Liu, 2012; Yang et al., 2013; Zhou and Fan, 2015]. These methods expect the resolution of input faces to be sufficient enough for detecting the facial landmarks and parts. When the resolution is very low, they fail to localize the components accurately, thus producing non-realistic faces. In other words, the facial component based methods are unsuitable to upsample very LR faces.

In this paper, we present a new transformative discriminative neural network (TDN) to overcome the above issues and achieve super-resolving a tiny (*i.e.*16×16 pixels) and unaligned face image by a remarkable upscaling factor 8, where we re-

construct 64 pixels for each single pixel of the input LR image.

Our network consists of two components: an upsampling network that comprises deconvolutional and spatial transformation network [Jaderberg et al., 2015] layers, and a discriminative network. The upsampling network is designed to progressively improve the resolution of the latent feature maps at each deconvolutional layer. We do not assume the LR face is aligned in advance. Instead, we compensate for any misalignment and changes through the spatial transformation network layers that are embedded into the upsampling network. One can use the pixel-wise intensity similarity between the estimated and the ground-truth HR face images as the objective function in the training stage. However, when the upscaling factor becomes larger, employing only the pixel-wise intensity similarity causes over-smoothed outputs. Therefore, we incorporate class similarity information that is provided by a discriminative network to enforce the upsampled HR faces to be similar to real face images. We back-propagate the discriminative errors to the upsampling network. Our end-to-end solution allows fusing the pixel-wise and class-wise information in a manner robust to spatial transformations and obtaining a super-resolved output with much richer details.

Overall, our main contributions have four aspects:

- We present a novel end-to-end transformative discriminative network (TDN) to super-resolve very low-resolution (16×16 pixels) face images with an upscaling factor 8×.
- For tiny input images where landmark based methods inherently fail, our method is the first solution to hallucinate an unaligned LR face image without requiring precise alignment in advance, which makes our method practical.
- Fusion of pixel-wise appearance similarity and class-wise discriminative information allows the super-resolution process to take full advantage of class-specific cues for the alignment and detail enhancement tasks.
- Our method achieves almost 4 dB PSNR improvement over the state-of-the-art.

4.4 Related Work

Face hallucination aims to magnify an LR image to its HR version, which contains extra high-frequency details. State-of-the-art face hallucination methods can be grouped into two categories: appearance based methods and facial components based methods.

Appearance based methods employ PCA to build a holistic face model or apply reference HR patches to reconstruct the HR counterparts of the LR patches. Baker and Kanade [2002] construct high-frequency details of aligned frontal face images by searching the best mapping between LR and HR patches from the training dataset. Wang and Tang [2005] develop an eigen-tranformation to super-resolve face images by establishing a linear mapping between LR and HR face subspaces. Liu et al.

[2007] employ a PCA based global appearance model to upsample LR faces and a local non-parametric model to enhance the facial details. Kolouri and Rohde [2015] explore optimal transport and subspace learning to morph an HR output. Ma et al. [2010] hallucinate an LR face image with position patches sampled from multiple aligned HR images, while Li et al. [2014] model the local face patches as a sparse coding problem. Since appearance based face hallucination methods require that the LR images are precisely aligned and have the same pose and expression as the HR references, these methods are sensitive to the misalignment of LR images. When misalignment or different poses and expressions exist, their performance may degrade dramatically.

Facial components based methods super-resolve facial parts rather than entire faces, and thus they can address various poses and expressions. Tappen and Liu [2012] use SIFT flow [Liu et al., 2011] to align LR images, and then restore the details of LR images by deforming the reference HR images. Yang et al. [2013] first detect facial components in the LR images and then transfer the most similar HR facial components in the dataset to the LR input. Since the facial components based methods require to extract facial components from LR inputs, the resolution of the input LR images cannot be very low. Otherwise, these methods may fail to localize facial components, thus generating non-realistic HR results.

Recently, convolutional neural network (CNN) based methods have been proposed and claimed the state-of-the-art performance [Dong et al., 2016a; Kim et al., 2016a; Wang et al., 2015; Bruna et al., 2016]. Because these methods are designed to upsample generic patches and do not fully exploit class-specific information, they are not suitable to hallucinate tiny faces. Zhou and Fan [2015] present a bi-channel CNN to hallucinate blurry face images. They first use CNN to extract facial features and then feed the features to fully connected layers to generate high-frequency facial details. This method is restricted to the input image size as the other facial component based approaches.

4.5 Proposed Method: TDN

Our transformative discriminative neural network achieves the image alignment and super-resolution simultaneously. The entire processing pipeline is shown in Fig. 4.1.

4.5.1 Network Architecture

The transformative discriminative neural network consists of two parts: an upsampling network that combines spatial transformation network layers and deconvolutional layers, and a discriminative network.

4.5.1.1 Upsampling Network

The parameters of our upsampling network are shown in Fig. 4.1 (red frame).

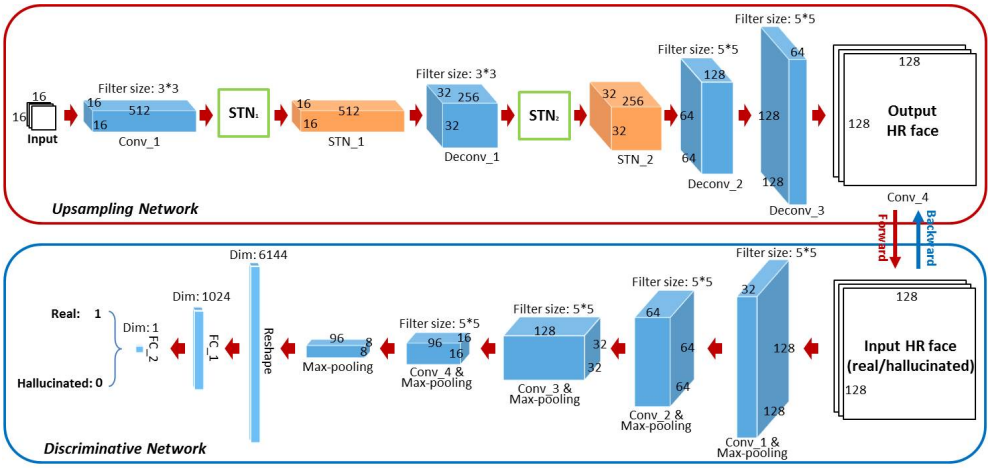


Figure 4.1: Our TDN consists of two parts: an upsampling network (in the red frame) and a discriminative network (in the blue frame).

Deconvolutional Layers: The deconvolutional layer, also known back-convolutional layer, can be made of a cascade of an upsampling layer and a convolutional layer, or a convolutional layer with a fractional stride. Therefore, the resolution of the output of the deconvolutional layers is larger than the resolution of its input. We employ the ℓ_2 regression loss, also known as Euclidean distance loss, to constrain the similarity between the hallucinated HR faces and their original HR ground-truth versions. We notice that previous works also employ similar deconvolutional layers to upsample natural scenes [Long et al., 2015; Fischer et al., 2015]. However, they only apply to generic images without exploiting any class-specific cues. Thus, their results tend to be smooth. In contrast, we train the network with face images and let it learn and memorize the facial parts for hallucination.

Spatial Transformation Layers: The spatial transformation network (STN) is recently proposed by Jaderberg et al. [2015]. It can estimate the motion parameters of images, and warp images to the canonical view. In our architecture, the spatial transformation network layers are represented as the green boxes in Fig. 4.1. These layers contain three modules: a localization module, a grid generator module, and a sampler. The localization module consists of a number of hidden layers and outputs the transformation parameters of an input relative to the canonical view. The grid generator module creates a sampling grid according to the estimated parameters. Finally, the sampler module maps the input onto the generated grid by bilinear interpolation.

Since we focus on in-plane rotations, translations, and scale changes without requiring a 3D face model, we employ the similarity transformation for face alignment. Although the STN can warp images, it is not straightforward to use them directly to align very LR face images. There are several factors needed to be considered: (i) After the alignment of LR images, facial patterns are blurred due to the resampling of the aligned faces by bilinear interpolation. (ii) Since the resolution is very low and

a wide range of poses exists, spatial transformations lead to alignment errors. (iii) Due to the blur and alignment errors, the upsampling network may fail to generate realistic HR faces. These factors can be observed in Fig. 4.2(f), where simply employing an STN to align an LR image causes artifacts in the upsampled faces due to interpolation blur and alignment errors.

Instead of using a single STN to align LR face images, we employ multiple STN layers to line up the feature maps. Using multiple layers significantly reduces the load on each spatial transformation network. In addition, resampling feature maps by multiple STN layers prevents from damaging or blurring input LR facial patterns. Since STN layers and the upsampling network are interwoven together (rather than being two individual networks), the upsampling network can learn to eliminate the undesired effects of misalignment in the training stage. As shown in Fig. 4.2(e), our upsampling network can reconstruct more high-frequency details than the CNN based super-resolution method (SRCNN) [Dong et al., 2016a], even when SRCNN is retrained with face patches.

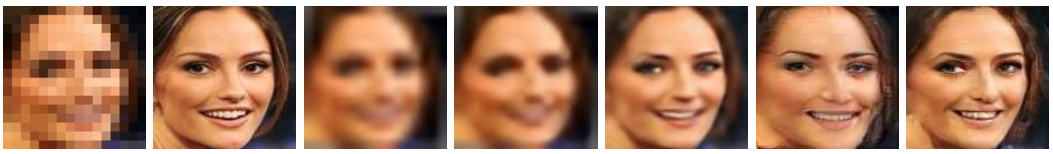
4.5.1.2 Discriminative Network

As seen in Fig. 4.2(e), the hallucinated faces are not sharp enough because the common parts learned by the upsampling network are averaged from similar components shared by different individuals. Thus, there is a quality gap between the real face images and the hallucinated faces. To bridge this gap, we inject class information. We integrate a discriminative network to distinguish whether the generated image is classified as an upright real face image or not. The parameters of the discriminative network are shown in the blue frame of Fig. 4.1. We employ a binary cross-entropy as the loss function. We backpropagate the discriminative error to revise the coefficients of the upsampling network, which enforces the facial parts learned by the deconvolutional layers to be as sharp and authentic as the real ones. A similar idea is employed in the generative adversarial networks [Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015], which are designed to generate a new face. Furthermore, the use of class information also improves the performance of the STN layers for face alignment since only upright faces are classified as valid faces. Therefore, the discriminative network also determines whether the faces are upright or not. As shown in Fig. 4.2(g), with the help of the discriminative information, the hallucinated face embodies more authentic, much sharper and better aligned details.

4.5.2 Training Details of TDN

In the training stage of our TDN, we assemble LR and HR face image pairs $\{L_i, H_i\}$ as our training dataset. Notice that the LR image L_i is not directly downsampled from the HR image H_i. There are different rotations, translations, and scale changes applied in the LR images while the training HR images are kept upright.

For the upsampling network, we use a pixel-wise ℓ_2 regression loss. Our intuition here is that the hallucinated HR face image $\hat{H}_i$ should be similar to its correspond-



(a) (b) (c) (d) (e) (f) (g)

Figure 4.2: Illustration of TDN with different configurations. (a) Unaligned 16×16 LR image. (b) Original 128×128 HR image. (c) Bicubic interpolation. (d) Result of SRCNN [Dong et al., 2016a] retrained with face patches. (e) Result of TDN without the discriminator network. (f) Result of TDN where an STN applied on the LR image directly. (g) Our full TDN.

ing reference HR image H_i. Since the STN layers are embedded in the upsampling network, the objective function $V(u,t)$ of the upsampling network is modeled as

$$\min_{u,t} V(u,t) = \mathbb{E}_{p(L_i,H_i)} \|\hat{H}_i - H_i\|_F^2, \tag{4.1}$$

where u and t represent the parameters of the upsampling network and the STN layers that are updated jointly. The STN layers align the feature maps while the upsampling network super-resolves the LR images with the deconvolutional layers. Above, $p(L_i, H_i)$ represents the joint probability distribution of the LR and HR faces in the training dataset.

As we mentioned, we exploit the discriminative information to achieve high-quality super-resolution of face images. To this end, we employ a set of convolutional layers in our discriminative network. These layers assess whether the hallucinated face is real and upright, or not. If the upsampling network can hallucinate an HR face that can convince the discriminative network that it is an authentic face, our super-resolved face will be very similar to real face images. In other words, the discriminative network cannot differentiate upsampled faces from real faces. This objective is achieved by maximizing the cross entropy. Therefore, we optimize the loss function of the discriminative network D as follows:

$$\begin{aligned} \max_{d} D(d) &= \mathbb{E}\left[\log D(H_i) + \log(1 - D(\hat{H}_i))\right] \\ &= \mathbb{E}_{p(H_i)}[\log D(H_i)] + \mathbb{E}_{p(\hat{H}_i)}[\log(1 - D(\hat{H}_i))], \end{aligned} \tag{4.2}$$

where d indicates the parameters of the discriminative network, and $p(H_i)$ and $p(\hat{H}_i)$ represent the distributions of real faces and the hallucinated faces from LR faces in the dataset. The above objective reaches the maximum when the network cannot distinguish H_i and $\hat{H}_i$. The loss D is backpropagated to the upsampling network to update the parameters u and t. By tuning u and t, the upsampling network not only can super-resolve the LR face images with appearance similarity, but also makes the hallucinated faces contain more class-specific details.

We use RMSprop [Hinton, 2012] to update the parameters u, t and d. In order to maximize D, the parameters d are updated by the stochastic gradient ascent,

$$\begin{aligned} \Delta^{j+1} &= \alpha\Delta^j + (1-\alpha)(\frac{\partial D}{\partial d})^2, \\ d^{j+1} &= d^j + \gamma\frac{\partial D}{\partial d}\frac{1}{\sqrt{\Delta^{j+1}+\epsilon}}, \end{aligned} \tag{4.3}$$

where γ and α are the learning rate and the decay rate, j represents the iteration index, Δ is an auxiliary variable, and ϵ is set to 10^{-8} to avoid division by zero. The parameters u and t are not only updated by the loss V but also D. For simplicity, let $T = (u, t)$, and the parameters are updated by the stochastic gradient descent,

$$\begin{aligned} \Delta^{j+1} &= \alpha\Delta^j + (1-\alpha)(\frac{\partial V}{\partial T} + \lambda\frac{\partial D}{\partial T})^2, \\ T^{j+1} &= T^j - \gamma(\frac{\partial V}{\partial T} + \lambda\frac{\partial D}{\partial T})\frac{1}{\sqrt{\Delta^{j+1}+\epsilon}}, \end{aligned} \tag{4.4}$$

where λ is used to trade off the appearance similarity constraint and the class-specific discriminative constraint. Since we aim to super-resolve an LR image, we put more constraint on appearance similarity. In our experiments, we set λ to 0.01. As the iterations progress, the upsampled faces become more similar to real faces, and thus we reduce the impact of the discriminative network gradually,

$$\lambda^i = \max\{\lambda \cdot 0.99^i, \lambda/2\}, \tag{4.5}$$

where i indicates the index of the epochs. Eqn. 4.5 guarantees that the influence of the discriminative information is preserved in the upsampling network. In our algorithm, the learning rate γ is set to 0.001 and multiplied by 0.99 after each epoch, and the decay rate is set to 0.01.

4.5.3 Hallucinating a Very LR Face Image

The discriminative network is only used for training of the upsampling network. In the testing stage (super-resolving a given test image), we feed the LR image into the upsampling network to obtain its upright super-resolved HR version. Because the ground-truth HR face images are upright in the training stage of the entire network, the output of the upsampling network will be an upright face image. As a result, our method does not require alignment of the very low-resolution images in advance. Our network provides an end-to-end mapping from an unaligned LR face image to an upright HR version, which mitigates potential artifacts caused by misalignment.

4.5.4 Implementation Details

In Fig. 4.1, the STN layers are constructed by convolutional and ReLU layers (Conv+ReLU), max-pooling layers with a stride 2 (MP2) and fully connected layers (FC).

Table 4.1: Quantitative evaluation on the entire test dataset.

Methods	PSNR	SSIM
Bicubic	18.41	0.54
[Yang et al., 2010]	18.21	0.52
[Dong et al., 2016a]	18.28	0.54
[Liu et al., 2007]	18.00	0.48
[Yang et al., 2013]	18.40	0.53
[Ma et al., 2010]	18.34	0.52
Ours	**22.66**	**0.66**

In particular, STN_1 layer is cascaded by: MP2, Conv+ReLU (with the filter size: 512×20×5×5), MP2, Conv+ReLU (with the filter size: 20×20×5×5), FC+ReLU (from 400 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_2 is cascaded by: MP2, Conv+ReLU (with the filter size: 256×128×5×5), MP2, Conv+ReLU (with the filter size: 128×20×5×5), MP2, Conv+ReLU (with the filter size: 20×20×3×3), FC+ReLU (from 180 to 20 dimensions) and FC (from 20 to 4 dimensions). In the convolution operations, we do not use padding.

In the following experimental part, some algorithms [Liu et al., 2007; Ma et al., 2010] require the alignments of LR inputs. Thus, we use STN_0 to align the LR inputs images for those methods. The only difference between STN_0 and STN_1 is that the first MP2 step in STN_1 is removed in STN_0.

4.6 Experiments

In this section, we compare our method with the state-of-the-art methods qualitatively and quantitatively.

4.6.1 Dataset

Our network is trained on the Celebrity Face Attributes (CelebA) dataset [Liu et al., 2015]. There are more than 200K face images in this dataset, and the images cover different pose variations and facial expressions. In training our network, we disregard these variations without grouping the face images into different pose and facial expression subcategories.

When generating the LR and HR face pairs, we randomly select 30K cropped face images from the CelebA dataset, and then resize them to 128×128 pixels as HR images. We manually transform the HR images while constraining the faces in the image region, and then downsample the HR images to generate their corresponding LR images. Note that, we do not explicitly change the scale of faces because in the CelebA the face sizes are different. (All protocol details, data, and code for this paper will be released.)

4.6.2 Comparison with the State-of-the-Art

Since we super-resolve an image with a substantial upscaling factor of 8×, for the methods that do not provide 8×, we apply the maximum upscaling factors recommended by the original papers multiple times (*e.g.*, twice 4× upscaling). For the face hallucination methods that assume very low-resolution faces are aligned beforehand, we use STN_0 to align LR faces. For fair comparisons and better illustration, we transform all the LR input images to the upright view as the inputs of the other methods.

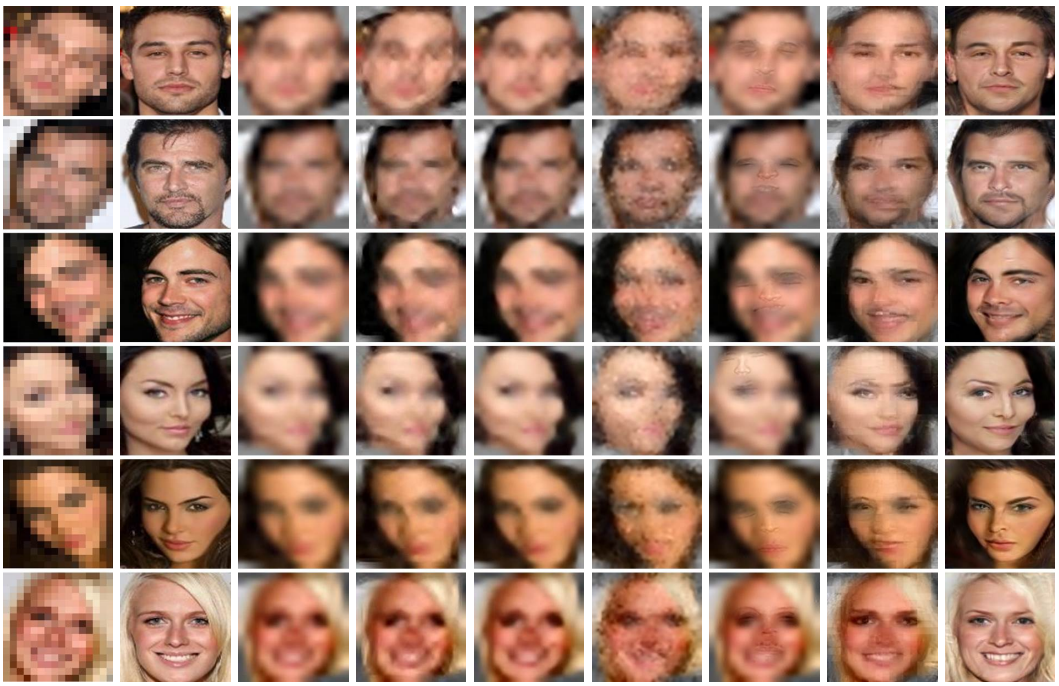
In Tab. 4.1, we report the quantitative comparison results using the average PSNR and structural similarity scores (SSIM) on the entire test dataset. As indicated in Tab. 4.1, our TDN attains the best PSNR and SSIM results. We found that if we only use the upsampling network to super-resolve LR faces, we can gain an extra 0.18 dB improvement but produces over-smoothed results. Therefore, there is a trade-off between the upsampling and discriminative networks. Since we aim to hallucinate high-resolution realistic facial details, we incorporate our discriminative network, and our TDN achieves an impressive 4.25 dB PSNR improvement over the state-of-the-art.

As shown in Fig. 4.3(c), traditional upsampling methods, *i.e.*, bicubic interpolation, cannot hallucinate authentic facial details. Since the resolution of inputs is very small, little information is contained in the input images. Simply interpolating input LR images cannot recover extra high-frequency details. As seen in Fig. 4.3(c), the upsampled images by bicubic interpolation still have some skew effects rather than laying in the upright view. This implies that simply using STN_0 to align input images still suffers from misalignment. Since we apply multiple STNs on the feature maps, which improves the alignment of the LR inputs, our method outputs well-aligned faces. As shown in the last row of Fig. 4.3, STN_0 uses bilinear interpolation to resample images, which changes the intensities of the LR input and introduces extra blurriness as well. In contrast, with the help of the discriminator network, our method can achieve much sharper results.

As shown in Fig. 4.3(d), the sparse coding based super-resolution (SCSR) method [Yang et al., 2010] cannot reconstruct high-frequency details either when the scaling factor is very large (*e.g.*8×), because the SCSR method cannot find a consistent correspondence between LR and HR patches as the upscaling factor becomes larger.

Dong et al. [2016a] propose a patch based convolutional network to super-resolve generic images, also known as SRCNN. This method is trained on generic patches and the maximum upscaling factor is 4. SRCNN, as a patch based method, cannot capture the whole face structure. However, training SRCNN with the whole face will introduce more ambiguity between LR and HR patches because the training patch size (*i.e.*128×128) is too large to learn a valid non-linear mapping. Hence, we retrain their model with face patches and an upscaling factor 8. As seen in Fig. 4.2(e), SRCNN cannot produce authentic high-frequency facial details. This also implies that our upsampling network is more suitable for the face hallucination task.

The face hallucination method based on appearance model [Liu et al., 2007] can



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 4.3: Comparison with the state-of-the-arts methods. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Yang et al. [2010]. (e) The method of Dong et al. [2016a] (SRCNN). (f) The method of Liu et al. [2007]. (g) The method of Yang et al. [2013]. (h) The method of Ma et al. [2010]. (i) Our method.

super-resolve very LR face images when the faces are precisely aligned (*i.e.*, face positions and head poses). Because the alignment errors of LR faces by STN_0 exist, the aligned LR faces have shifts with the appearance model. Besides, we use all the faces in the training dataset to train an appearance model, and there are different facial expressions and poses in the training dataset, which make the appearance model noisy. Hence, as shown in Fig. 4.3(f), their results suffer severe artifacts without hallucinating authentic facial details.

The structured face hallucination method [Yang et al., 2013] looks for the most similar facial components in the dataset and then transfer those HR components to the LR input ones. However, when the resolution of the input images is very small, localizing facial landmarks in LR inputs is difficult. Thus their method cannot accurately find the most similar facial components in the dataset and fails to output HR transferred components, as illustrated in Fig. 4.3(g). Therefore, this method is unsuitable to hallucinate very LR face images.

The method of Ma et al. [2010] exploits position patches to hallucinate HR faces. Thus this method requires the LR inputs to be precisely aligned with the reference images in the training dataset. As seen in Fig. 4.3(h), when there are obvious align-

ment errors in the aligned LR faces, their method will output mixed faces in their results. Furthermore, as the upscaling factor increases, the correspondences between LR and HR patches become more inconsistent. Hence, this method suffers from obvious block artifacts around the boundaries of different patches.

As shown in Fig. 4.3(i), our method reconstructs authentic facial details. Note that, the reconstructed faces have different poses and facial expressions. Since our method applies multiple STNs on feature maps to align face images, we can achieve better alignment results without damaging input LR images. Furthermore, our method does not need to warp input images directly, so there are no blank regions in our results. It implies that our method can exploit information better than the other methods.

4.7 Conclusions

We presented a transformative discriminative network to super-resolve unaligned very low-resolution face images in an end-to-end manner. Our network learns how to align faces and how to upsample them by making use of the class-specific information. It attains a significant upsampling factor of $8\times$ while hallucinating rich and authentic facial details. Since our method does not require any feedback of face poses and facial expressions, it is very practical.

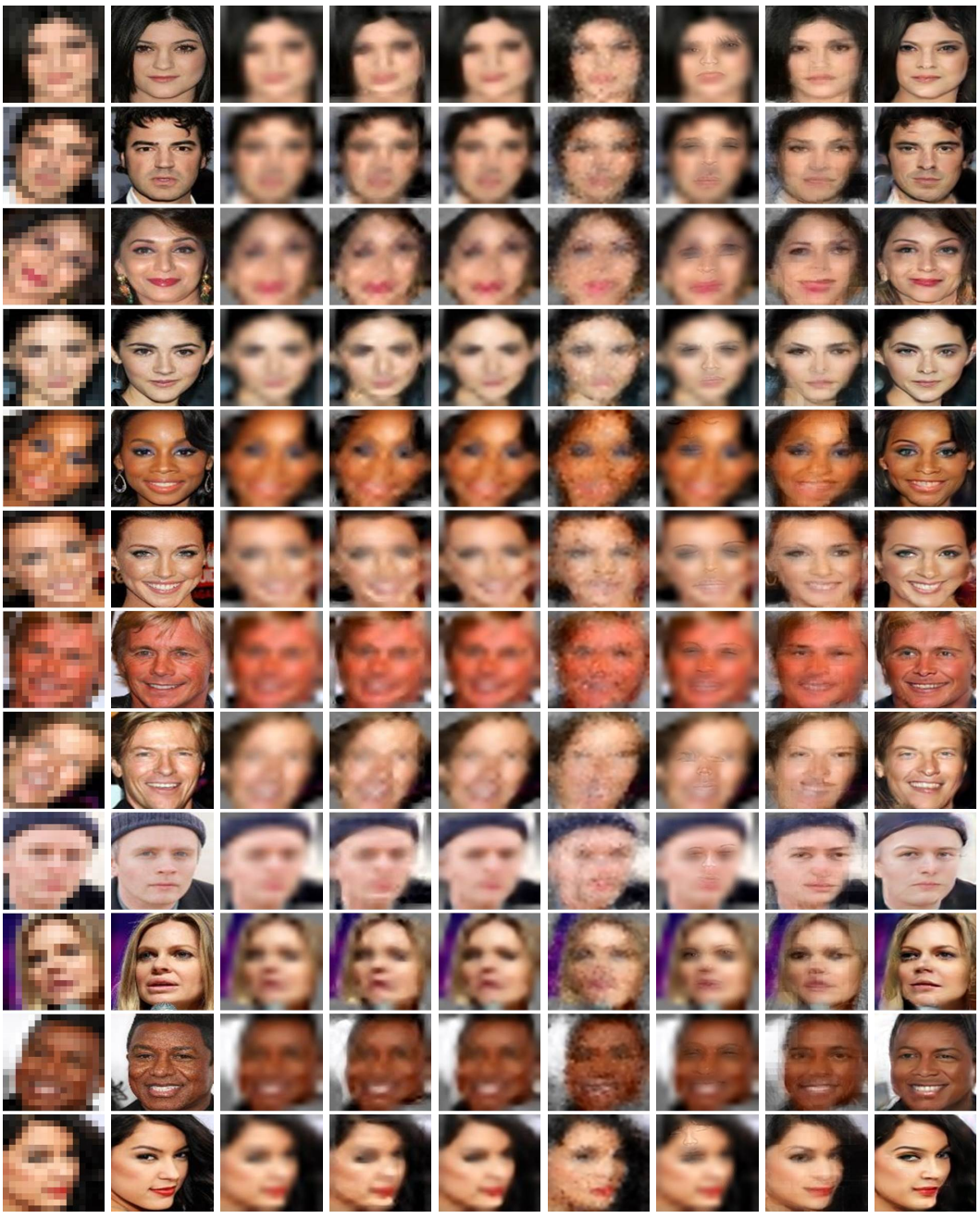
4.8 Appendix

4.8.1 Impact of Using Multiple STNs

In this part, we demonstrate that using multiple STNs can improve the accuracy of face alignment according to the work [Jaderberg et al., 2015] and the impact of each STN on the alignment. We employ STN_1 and STN_2 individually in the upsampling network, and then compare the upsampled results quantitatively by PSNR and SSIM. The results of solely applying STN_1 or STN_2 to align feature maps in the upsampling network are shown in the first and second columns of Tab. 4.2. As illustrated in the third column of Tab. 4.2, by using multiple STNs, our TDN achieves the highest PSNR and SSIM. It implies that better alignment accuracy can be obtained by employing multiple STNs.

Table 4.2: Evaluation on using different STNs

Modules	STN_1	STN_2	$\text{STN}_1 + \text{STN}_2$
PSNR	22.50	22.23	22.66
SSIM	0.65	0.63	0.66



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 4.4: Comparison with the state-of-the-arts. (a) LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Yang et al. [2010]. (e) The method of Dong et al. [2016a] (SRCNN). (f) The method of Liu et al. [2007]. (g) The method of Yang et al. [2013]. (h) The method of Ma et al. [2010]. (i) Our method.

4.8.2 Addtional Experimental Results

Here, we demonstrate more experimental comparisons with the state-of-the-art methods in Fig. 4.4.

We do not apply an STN on the third deconvolutional layer due to the limitation of hardware memory. Because the upsampling network learns common facial components from aligned feature maps, aligning feature maps in the early layers can facilitate the learning of the upsampling network. As expected, Tab. 4.2 shows that applying an STN in the first deconvolutional layer (STN_1) can achieve better results than in the second layer (STN_2).

Chapter 5

Hallucinating Very Low-Resolution Unaligned and Noisy Face Images by Transformative Discriminative Autoencoders

5.1 Foreword

In chapter 2 and chapter 3, our methods require all the low-resolution face images to be aligned. Then we reduce the requirements of alignments of low-resolution faces by incorporating spatial transformer networks into our upsampling network in chapter 4. However, our previous works assume the low-resolution face images are noise-free. Since the resolution of input face images is very small, every pixel matters significantly in super-resolution. Thus, image noise will deteriorate the face hallucination performance dramatically. In this chapter we present a transformative discriminative autoencoder to upsample noisy low-resolution face images while suppressing artifacts caused by the noise. We observe that directly denoising low-resolution faces may corrupt low-resolution facial patterns and leads to distortions and artifacts in upsampled high-resolution face images. Instead of denoising low-resolution images, we first super-resolve low-resolution face images while reducing noise. Since noise may cause artifacts in upsampled high-resolution faces, we then project denoised and upsampled high-resolution faces to noise-free low-resolution ones to reduce the artifacts. Finally, we can achieve high-quality high-resolution face images by upsampling the noise-free low-resolution ones.

This chapter has been published as a conference paper: Xin Yu, Fatih Porikli: Hallucinating Very Low-Resolution Unaligned and Noisy Face Images by Transformative Discriminative Autoencoders. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3760-3768, 2017.

5.2 Abstract

Most of the conventional face hallucination methods assume the input image is sufficiently large and aligned, and all require the input image to be noise-free. Their performance degrades drastically if the input image is tiny, unaligned, and contaminated by noise.

In this paper, we introduce a novel transformative discriminative autoencoder to 8× super-resolve unaligned noisy and tiny (16×16) low-resolution face images. In contrast to encoder-decoder based autoencoders, our method uses decoder-encoder-decoder networks. We first employ a transformative discriminative decoder network to upsample and denoise simultaneously. Then we use a transformative encoder network to project the intermediate HR faces to aligned and noise-free LR faces. Finally, we use the second decoder to generate hallucinated HR images. Our extensive evaluations on a very large face dataset show that our method achieves superior hallucination results and outperforms the state-of-the-art by a large margin of **1.82** dB PSNR.

5.3 Introduction

Face images provide critical information for visual perception and identity analysis. However, when they are noisy and their resolutions are inadequately small (*e.g.*as in some surveillance videos), there is little information available to be inferred reliably from them. Very low-resolution and noisy face images not only impede human perception but also impair computer analysis.

To tackle this challenge, face hallucination techniques aim at recovering high-resolution (HR) counterparts from low-resolution (LR) face images and have received significant attention in recent years. Previous state-of-the-art methods mainly focus on recovering HR faces from aligned and noise-free LR face images. More specifically, face hallucination methods based on holistic appearance models [Baker and Kanade, 2000, 2002; Liu et al., 2001; Wang and Tang, 2005; Liu et al., 2007; Hennings-Yeomans et al., 2008; Ma et al., 2010; Yang et al., 2010; Li et al., 2014; Kolouri and Rohde, 2015; Wang et al., 2014; Yu and Porikli, 2016] require LR faces to be precisely aligned beforehand. However, when the LR images are contaminated by noise, the accuracy of face alignment degrades dramatically. Besides, due to the wide range of pose and expression variations, it is difficult to learn a comprehensive, holistic appearance model for LR images not aligned appropriately. As a result, these methods often produce ghosting artifacts for noisy unaligned LR inputs.

Rather than learning holistic appearance models, facial components based face hallucination methods have been proposed [Tappen and Liu, 2012; Yang et al., 2013; Zhou and Fan, 2015; Zhu et al., 2016b]. They transfer HR facial components from the training dataset to the input LR images without requiring alignment of LR input images in advance. These methods heavily rely on the successful localization of facial landmarks. Because facial landmarks are difficult to detect in very low resolution (16×16 pixels) images, they fail to localize the facial components accurately and thus

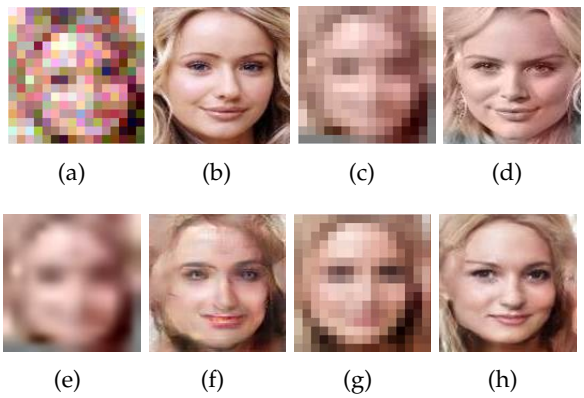


Figure 5.1: Comparison of our method with the CNN based face hallucination URDGN [Yu and Porikli, 2016]. (a) 16 × 16 LR input image. (b) 128 × 128 HR original image. (c) Denoised and aligned LR image. We firstly apply BM3D [Dabov et al., 2007] and then STN [Jaderberg et al., 2015]. (d) The corresponding most similar face in the training dataset. (e) Bicubic interpolation of (c). (f) Image generated by URDGN. Note that, URDGN super-resolves the denoised and aligned LR image, not the original LR input (in favor of URDGN). (g) The denoised and aligned LR image by our decoder-encoder as an intermediate output. (h) The final hallucinated face by our TDAE method.

produce artifacts in the upsampled face images. In other words, the facial component based methods are not suitable to upsample noisy unaligned LR faces either.

Considering the resolution of faces is too small and the presence of noise, face detectors may also fail to locate such tiny noisy faces. Thus, using pose specific face detectors as a preprocessing step to compensate for misalignments is also impractical.

In this paper, we propose a new transformative discriminative autoencoder (TDAE) to super-resolve a tiny (16×16 pixels) unaligned and noisy face image by a remarkable upscaling factor of 8×, where we estimate 64 pixels for each single pixel of the input LR image. Furthermore, each pixel has also been contaminated by noise, making the task even more challenging.

Our TDAE consists of three serial components: a decoder, an encoder, and a second decoder. Our decoder network comprises deconvolutional and spatial transformation layers [Jaderberg et al., 2015]. It can progressively upsample the resolutions of the feature maps by its deconvolutional layers while aligning the feature maps by its spatial transformation layers. Similar to [Yu and Porikli, 2016], we employ not only the pixel-wise intensity similarity between the hallucinated face images and the ground-truth HR face images but also the class similarity constraint that enforces the upsampled faces to lie on the manifold of real faces by a discriminative network. Hence, we achieve a transformative decoder that is also discriminative. Since the LR inputs are noisy, the hallucinated faces after the decoder may still contain artifacts. In order to obtain aligned and noise-free LR faces, we project the upsampled HR

faces back onto the LR face domain by a transformative encoder. Finally, we train our second decoder on the projected LR faces to attain hallucinated HR face images. In this manner, the artifacts are greatly reduced and our TDAE produces authentic HR face images.

Overall, the contributions of this paper are mainly in four aspects:

- We propose a new transformative-discriminative architecture to hallucinate tiny (16×16 pixels) unaligned and noisy face images by an upscaling factor of 8×.
- In contrast to conventional autoencoders, we first device a decoder-encoder structure to generate noise-free and aligned LR faces, and then a second decoder trained on the encoded LR faces to hallucinate high-quality HR face images.
- Our method does not require to model or estimate noise parameters. It is agnostic to the underlying spatial deformations and contaminated noise.
- To the best of our knowledge, our method is the first attempt to address the super-resolution of tiny and noisy face images without requiring alignment of LR faces beforehand, which makes our method practical.

5.4 Related Work

Face hallucination has received significant attention in recent years [Tappen and Liu, 2012; Yang et al., 2013; Wang et al., 2014; Kolouri and Rohde, 2015; Zhou and Fan, 2015; Zhu et al., 2016b; Yu and Porikli, 2016]. Previous face hallucination methods mainly focus on recovering HR faces from aligned and noise-free LR face images, and in general, they can be grouped into two categories: holistic methods and part-based methods.

Holistic methods use global face models learned by PCA to hallucinate entire HR faces. In the work [Wang and Tang, 2005], an eigen-transformation is proposed to generate HR face images by establishing a linear mapping between LR and HR face subspaces. Similarly, Liu et al. [2007] employ a global appearance model learned by PCA to upsample aligned LR faces and a local non-parametric model to enhance the facial details. The work [Kolouri and Rohde, 2015] explores optimal transport and subspace learning to morph an HR output according to the given aligned LR faces. Since holistic methods require LR face images to be precisely aligned and share the same pose and expression as the HR references, they are very sensitive to the misalignments of LR images. Besides, image noise makes the alignment of LR faces even more difficult.

Part-based methods upsample facial parts rather than entire faces, and thus they can handle various poses and expressions. They either employ a training dataset of reference patches to reconstruct the HR counterparts of the input LR patches or exploit facial components. In [Baker and Kanade, 2002], high-frequency details of aligned frontal face images are reconstructed by finding the best mapping between

LR and HR patches. The work in [Yang et al., 2010] uses coupled LR/HR dictionaries to enhance the details. In [Ma et al., 2010], an LR face image is super-resolved with position patches sampled from multiple aligned HR images. Li et al. [2014] model the local face patches as a sparse coding problem rather than averaging the reference HR patches directly. In [Tappen and Liu, 2012], SIFT flow [Liu et al., 2011] is exploited to align the facial parts of LR images, and then the details of LR images are reconstructed by warping the reference HR images. Yang et al. [2013] first localize facial components in the LR images and then transfer the most similar HR facial components in the dataset to the LR inputs. Since part-based methods often require extraction of facial components in LR inputs, their performance degrades dramatically when the LR faces are tiny or noisy.

As large-scale data becomes available, convolutional neural network (CNN) based SR methods [Kim et al., 2016a; Wang et al., 2015; Dong et al., 2016a; Bruna et al., 2016] have been proposed and achieved the state-of-the-art performance. However, because these SR methods are designed to upsample generic patches and do not fully exploit class-specific information, they are not suitable to hallucinate tiny faces. The work [Zhou and Fan, 2015] employs a CNN to extract facial features and then generates high-frequency facial details based on the extracted features. Due to the requirement of the facial feature extraction, the resolution of the input cannot be low. Very recently, Yu and Porikli [2016] present a discriminative generative network to super-resolve LR face images. Their method addresses different facial expressions and head poses without requiring facial landmarks, but it needs the eyes to be aligned in advance. Zhu et al. [2016b] propose a cascade bi-network to super-resolve very low-resolution and unaligned faces. However, when there is noise in the LR images, this method may fail to localize the face parts accurately, thus producing artifacts in the outputs.

5.5 Proposed Method: TDAE

Our transformative discriminative autoencoder has three complementary components: two transformative discriminative decoders (as shown in Fig. 5.2) and a transformative encoder (as shown in Fig. 5.3). In the training phase, our parameters of TDAE are learned in three steps (Sec. 5.5.3). In the testing phase, we cascade the transformative upsampling network of the first decoder DEC_1, the encoder ENC, and the second decoder DEC_2 together to hallucinate the final HR faces in an end-to-end manner. The whole pipeline is illustrated in Fig. 5.4

5.5.1 Architecture of Decoder

Our decoder architecture is composed of two sub-networks, a transformative upsampling network (TUN) and a discriminative network. In the transformative upsampling network, we first apply two convolutional layers with larger receptive fields to partially reduce noise artifacts rather than feeding noisy images into the deconvolutional layers directly. The deconvolutional layer can be made of a cascade of an upsampling layer and a convolutional layer, or a convolutional layer with a fractional

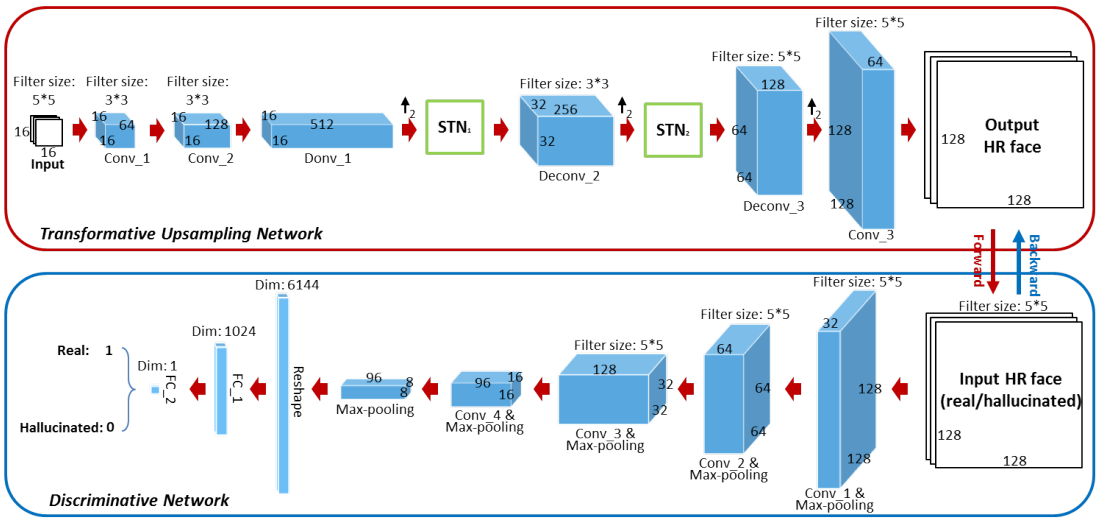


Figure 5.2: Our transformative discriminative decoder consists of two parts: a transformative upsampling network (in the red frame) and a discriminative network (in the blue frame).

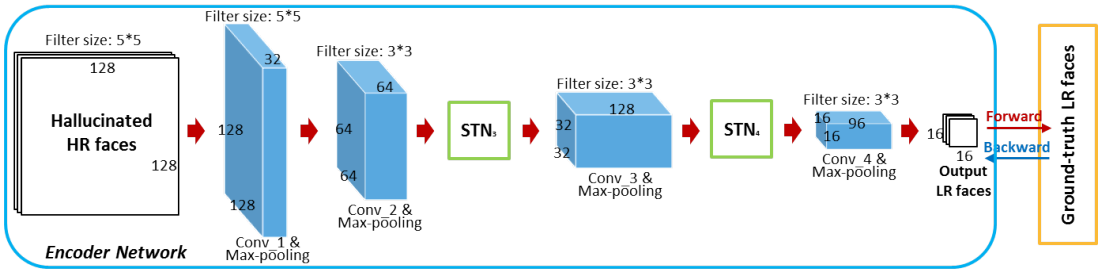


Figure 5.3: Architecture of our transformative encoder.

stride [Zeiler et al., 2010; Zeiler and Fergus, 2014]. Therefore, the resolution of the output image of the deconvolutional layer is larger than the resolution of its input image. We employ the ℓ_2 regression loss, also known as Euclidean distance loss, to constrain the similarity between the hallucinated HR faces and their HR ground-truth versions.

As reported in the work [Yu and Porikli, 2016], deconvolutional layers supervised by ℓ_2 loss tend to produce over-smoothed results. To tackle this, we embed the class-specific discriminative information into the deconvolutional layers by a discriminative network (as shown in the blue frame in Fig. 5.2). The discriminative network is able to distinguish whether an image (its input) is sampled from authentic face images or hallucinated ones. The corresponding discriminative information is backpropagated to the deconvolutional layers. Hence, the deconvolutional layers can generate HR face images more similar to the real faces.

We notice that rotational and scale misalignments of LR face images will lead to apparent artifacts in the upsampled face images in [Yu and Porikli, 2016]. By con-

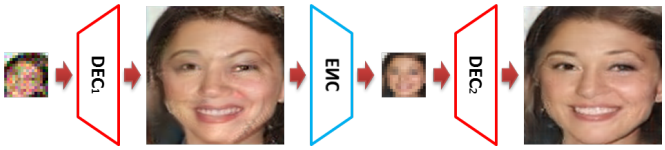


Figure 5.4: Workflow of our transformative discriminative autoencoder. Colors of the boxes refer to the networks in Fig.5.2 and Fig.5.3.

trast, our decoder can align the LR faces automatically and hallucinate face images simultaneously. In order to align LR faces, we incorporate the spatial transformation network (STN) [Jaderberg et al., 2015] into our network, as shown in the green box in Fig. 5.2. STN can estimate the transformation parameters of images, and then warp images to a canonical view.

There are three modules in STN: a localization module, a grid generator module, and a sampler. The localization module consists of a number of hidden layers and outputs the transformation parameters of an input relative to the canonical view. The grid generator module constructs a sampling grid according to the estimated parameters, and then the sampler module maps the input onto the generated grid by bilinear interpolation.

Here, we mainly focus on in-plane rotations, translations, and scale changes, and thus use the similarity transformation to align faces. Considering the resolution of our inputs is very small and input images are noisy, using state-of-the-art denoising algorithms to reduce noise and then employing an STN to align LR faces will introduce extra blurriness, as shown in Fig. 5.1(c) and Fig. 5.5(c). Therefore, aligning LR faces in the image domain may blur the original LR facial patterns and leads to artifacts as visible in the results of Yu and Porikli [2016] in Fig. 5.1(f). To prevent from this, we apply STNs to *align feature maps*. As reported in [Jaderberg et al., 2015], using multiple STNs can improve the accuracy of the alignment. As a trade-off between the accuracy and GPU memory usage, we employ two STNs following the first two deconvolutional layers.

Our decoder not only embeds discriminative information but also processes multiple tasks (denoising, alignment, and upsampling) simultaneously. As shown in Fig. 5.5(f), our transformative discriminative decoder can reconstruct more salient high-frequency details and aligned upsampled HR face images as well.

5.5.2 Architecture of Encoder

By feeding an unaligned and noisy LR input to our transformative discriminative decoder network DEC_1, we obtain an intermediate HR face image. As shown in Fig. 5.5(f), the intermediate HR face contains more high-frequency details and it is roughly aligned. The noise is comparatively reduced as well. However, the intermediate images may still contain artifacts, which are mainly caused by noise. We observe that noise not only distorts the LR facial patterns but also affects the face alignment. In order to achieve authentic HR face images, these artifacts should be

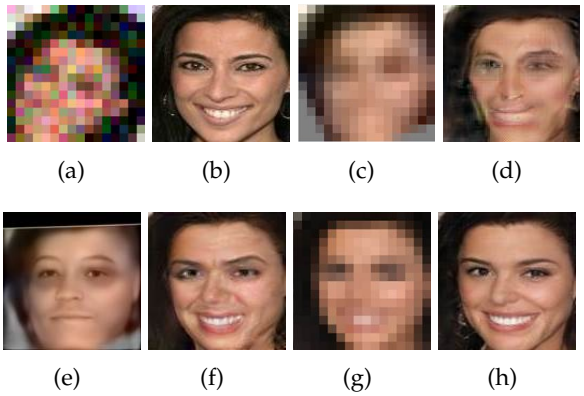


Figure 5.5: Comparison of our method with the CNN based face hallucination methods. (a) The input 16×16 LR image. (b) The original upright 128×128 HR image (for comparison purposes). (c) The denoised and aligned version of (a). (d) The result of URDGN [Yu and Porikli, 2016]. (e) The result of CBN [Zhu et al., 2016b]. (f) The result of our DEC_1. (g) The aligned and noise-free LR face projected by our ENC. (h) Our final result.

removed while preserving the high-frequency facial details.

Our intuition is that projecting intermediate HR images to LR images, artifacts and noise can be suppressed further, which would allow us to apply our decoder to super-resolve these almost noise-free and approximately aligned LR faces. However, a decimation with anti-aliasing or simple downsampling may introduce additional artifacts into the LR face images. Therefore, we design another CNN, regarded as the encoder ENC, to project intermediate HR images to noise-free LR versions as illustrated in Fig. 5.3. Considering the upsampled HR faces may still have misalignments, we also incorporate STNs into our encoder to provide further alignment improvement.

When training the encoder, we constrain the projected LR faces to be similar to the aligned ground-truth LR faces. This helps us to generate aligned and noise-free LR faces, as shown in Fig. 5.1(g) and Fig. 5.5(g).

To obtain HR face images, we employ a second decoder DEC_2 to super-resolve the LR faces projected by the ENC. The decoder DEC_2 shares the same architecture as the one in Fig. 5.2. By employing the decoder-encoder structure, we can jointly align the input LR faces and handle noise as shown in Fig. 5.1(g) and Fig. 5.5(g). By exploiting the encoder-decoder structure, we are able to remove artifacts in the upsampled HR faces, thus achieving high-quality, more authentic, hallucinated HR face images as shown in Fig. 5.5(h).

5.5.3 Training Details of TDAE

We divide the training phase of our TDAE into three stages: i) Training the transformative discriminative decoder network DEC_1, as illustrated in Fig. 5.2. ii) Training

the encoder ENC, as shown in Fig. 5.3. iii) Training the decoder DEC_2, which shares the same architecture as DEC_1.

5.5.3.1 Training Discriminative Decoder

We construct LR and HR face image pairs $\{l_i^n, h_i\}$ as our training dataset for the training of our transformative discriminative decoder DEC_1. Here, h_i represents aligned HR face images, and l_i^n is *not* directly downsampled from the HR face image h_i. We apply rotations, translations, and scale changes to h_i to obtain unaligned HR image h_i^u. Then, we downsample h_i^u and then add Gaussian noise to obtain the noisy unaligned LR faces l_i^n.

Since we impose the upsampled image $\hat{h}_i$ by our decoder should be similar to its corresponding reference HR image h_i, we use pixel-wise Euclidean distance, as known as ℓ_2 regression loss, to enforce the intensity similarity. The loss function $U(s)$ of the TUN is modeled as,

$$\min_s U(s) = \mathbb{E}_{(l_i^n, h_i)\sim p(l^n, h)} \|\hat{h}_i - h_i\|_F^2, \tag{5.1}$$

where s indicates the parameters of the TUN. The convolutional layers, the STN layers, and the deconvolutional layers are updated jointly in the TUN. The STN layers align the feature maps while the deconvolutional layers upsample the resolution of the feature maps gradually. Here, $p(l^n, h)$ indicates the joint distribution of the LR and HR face images in the training dataset.

As mentioned in [Yu and Porikli, 2016], only applying intensity similarity constraint will lead to over-smoothed results. Similar to [Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015; Yu and Porikli, 2016], we infuse class-specific discriminative information into the TUN by exploiting a discriminative network. The architecture of the discriminative network is illustrated in the blue frame in Fig. 5.2. It is designed to distinguish whether an image is realistic or hallucinated. If an HR face super-resolved by our decoder can convince the discriminative network that it is a real face image, our hallucinated faces will be similar to real face images. In other words, our goal is to make the discriminative network fail to distinguish hallucinated faces from real ones. Hence, we maximize the cross-entropy of the discriminative network L as follows:

$$\begin{aligned} \max_t L(t) &= \mathbb{E}\left[\log D(h_i) + \log(1 - D(\hat{h}_i))\right] \\ &= \mathbb{E}_{h_i\sim p(h)}[\log D(h_i)] + \mathbb{E}_{\hat{h}_i\sim p(\hat{h}))}[\log(1 - D(\hat{h}_i))], \end{aligned} \tag{5.2}$$

where t represents the parameters of the discriminative network, $p(h)$ and $p(\hat{h})$ indicate the distributions of the real faces and the hallucinated faces, and $D(h_i)$ and $D(\hat{h}_i)$ are the outputs of the discriminative network. The loss L is backpropagated to the TUN in order to update the parameters s. By injecting discriminative information to s, our decoder can hallucinate more authentic HR faces.

In our decoder network, every layer is differentiable, and thus we use backpropa-

gation to learn its parameters. RMSprop [Hinton, 2012] is employed to update s and t. To maximize the discriminative network objective L, we use the stochastic gradient ascent that updates the parameters t as follows:

$$\begin{aligned} \Delta^{i+1} &= \gamma\Delta^{i} + (1-\gamma)(\frac{\partial L}{\partial t})^2, \\ t^{i+1} &= t^{i} + r\frac{\partial L}{\partial t}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}, \end{aligned} \tag{5.3}$$

where r and γ are the learning rate and decay rate, respectively, i is the index of iteration, Δ is an auxiliary variable, and ϵ is set to 10^{-8} to avoid division by zero. For the TUN, both losses U and L are used to update the parameters s by the stochastic gradient descent,

$$\begin{aligned} \Delta^{i+1} &= \gamma\Delta^{i} + (1-\gamma)(\frac{\partial U}{\partial s} + \lambda\frac{\partial L}{\partial s})^2, \\ s^{i+1} &= s^{i} - r(\frac{\partial U}{\partial s} + \lambda\frac{\partial L}{\partial s})\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}, \end{aligned} \tag{5.4}$$

where λ is a trade-off weight between the intensity similarity term and the class similarity term. Since our goal is to hallucinate an HR face, we put a higher weight on the intensity similarity term and set λ to 0.01. As the iteration progresses, the super-resolved faces will be more similar to real faces. Therefore, we gradually reduce the impact of the discriminative network by decreasing λ as,

$$\lambda^{j} = \max\{\lambda \cdot 0.99^{j}, \lambda/2\}, \tag{5.5}$$

where j indicates the index of the epochs. Eqn. 5.5 also guarantees that the class-specific discriminative information is preserved in the decoder network during the training phase.

5.5.3.2 Training Encoder

In training our transformative encoder, we use the outputs of DEC_1 $\hat{h}_i$ and the ground-truth aligned LR images l_i as our training dataset. Since there may be mis-alignment in $\hat{h}_i$, we also embed STNs into our encoder ENC to align the LR faces. During the training of the transformative encoder, the downsampled LR faces $\hat{l}_i$ is constrained to be similar to the ground-truth aligned LR faces l_i. Therefore, the objective function of the transformative encoder $E(e)$ is modeled as,

$$\begin{aligned} \min_{e} E(e) &= \mathbb{E}_{(l_i,\hat{h}_i)\sim p(l,\hat{h})}\|\Psi(\hat{h}_i) - l_i\|_F^2 \\ &= \mathbb{E}_{(l_i,\hat{h}_i)\sim p(l,\hat{h})}\|\hat{l}_i - l_i\|_F^2, \end{aligned} \tag{5.6}$$

where e is the parameters of the transformative encoder, and $\Psi(\hat{h}_i)$ represents the mapping from the intermediate upsampled HR faces $\hat{h}_i$ to the projected LR faces $\hat{l}_i$. Similar to Eqn. 5.1 and Eqn. 5.2, we also use RMSprop to update e by the stochastic

gradient descent,

$$\begin{aligned} \Delta^{i+1} &= \gamma\Delta^{i} + (1-\gamma)(\frac{\partial E}{\partial e})^2, \\ e^{i+1} &= e^{i} - r\frac{\partial E}{\partial s}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}. \end{aligned} \tag{5.7}$$

To obtain the final HR faces, we integrate a second decoder DEC_2 to super-resolve the projected LR face images. DEC_2, as shown in Fig. 5.4, is trained on the encoded LR and aligned ground-truth HR image pairs $\{\hat{l}_i, h_i\}$.

After training the encoder network, we use the encoder ENC to generate the training dataset $\hat{l}_i$, and then train DEC_2 by using the image pairs $\{\hat{l}_i, h_i\}$. The training procedure of DEC_2 is as the same as Sec. 5.5.3.1.

5.5.4 Hallucinating HR from Unaligned & Noisy LR

The discriminative network is only employed in training our decoders. When hallucinating HR faces, the discriminative work is not used. In the testing phase, we first feed an unaligned and noisy LR face l_i^n into the decoder DEC_1 to obtain an upsampled intermediate HR image $\hat{h}_i$. Then, we use our encoder ENC to project the intermediate HR face $\hat{h}_i$ to an aligned LR face $\hat{l}_i$. Finally, we use the decoder DEC_2 to super-resolve the aligned LR face $\hat{l}_i$ and attain our final hallucinated face $\tilde{h}_i$.

Since in the training phase we use upright HR faces as targets, our TDAE not only super-resolves the LR faces but also aligns HR face images simultaneously. Although we need to train our network in three steps, it can hallucinate an unaligned and noisy LR face to an upright HR version in an end-to-end fashion.

5.5.5 Implementation Details

The STN layers, as shown in Fig. 5.2 and Fig. 5.3, are built by convolutional and ReLU layers (Conv+ReLU), max-pooling layers with a stride 2 (MP2) and fully connected layers (FC). Specifically, STN_1 layer is built by cascading the layers: MP2, Conv+ReLU (filter size: 512×20×5×5), MP2, Conv+ReLU (20×20×5×5), FC+ReLU (from 400 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_2 is constructed by cascading the layers: MP2, Conv+ReLU (256×128×5×5), MP2, Conv+ReLU (128×20×5×5), MP2, Conv+ReLU (20×20×3×3), FC+ReLU (from 180 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_3 is constructed by cascading the layers: MP2, Conv+ReLU (128×20×5×5), MP2, Conv+ReLU (filter size: 20×20×5×5), MP2, FC+ReLU (from 80 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_4 layer is built by cascading the layers: Conv+ReLU (96×20×5×5), MP2, Conv+ReLU (20×20×5×5), FC+ReLU (from 80 to 20 dimensions) and FC (from 20 to 4 dimensions). In the convolution operations, padding is not used.

In the following experimental part, some algorithms [Ma et al., 2010; Yu and Porikli, 2016] require the alignment of LR inputs. Thus, we employ STN_0 to align the

LR images for those methods. The only difference between STN_0 and STN_1 is that the first MP2 step in STN_1 is removed in STN_0.

In training our decoders and encoder, we use the same learning rate r and decay rate γ. We set the learning rate r to 0.001 and multiply 0.99 after each epoch, and the decay rate is set to 0.01.

5.6 Experiments

We compare our method with the state-of-the-art methods qualitatively and quantitatively. We employ BM3D [Dabov et al., 2007] to reduce the image noise, and then align the LR inputs by STN_0. In the experiments, we only show the upright HR ground-truth faces h_i for comparison purposes.

5.6.1 Dataset

We use the Celebrity Face Attributes (CelebA) dataset [Liu et al., 2015] to train our TDAE. There are more than 200K face images in this dataset, and the images cover different pose variations and facial expressions. We use these images without grouping them into different pose and facial expression subcategories.

When generating the LR and HR face pairs, we randomly select 30K cropped aligned face images from the CelebA dataset, and then resize them to 128×128 pixels as HR images. We use 28K images for training and 2K for our tests. We manually transform the HR images while constraining the faces to be visible in the in the image, downsample the HR images to generate LR images, and add Gaussian noise. In the training of the decoder DEC_1, we apply zero mean Gaussian noise with the standard deviation 10% of the maximum image intensity to the LR images.

5.6.2 Qualitative Comparison with the SoA

Since some super-resolution baselines [Ma et al., 2010; Yu and Porikli, 2016] require the input LR faces to be aligned, for a fair comparison we align the LR faces by STN_0 for the compared methods. We present only the aligned upright HR ground-truth faces for easy comparisons.

As shown in Fig. 5.6(c), conventional bicubic interpolation cannot generate facial details. Since the resolution of inputs is very small, little information is contained in the input images. Furthermore, the upsampled images also have some deformations. This indicates that aligning very LR images is more difficult when there is noise in the images.

Dong et al. [2016a] present a CNN based general purpose super-resolution method, also known as SRCNN. Since SRCNN is patch based, it cannot capture the global face structure. Training SRCNN with the full face images introduces more ambiguity because the patch size (*i.e.*128×128) is too large to learn a valid non-linear mapping. Hence, we employ an upscaling factor of 8× to retrain it. As seen in Fig. 5.6(d), SRCNN cannot produce authentic facial details.

(a) (b) (c) (d) (e) (f) (g) (h)

Figure 5.6: Comparison with the state-of-the-arts methods at the noise level 10%. (a) Unaligned and noisy LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Results of Dong et al. [2016a]. (e) Results of Ma et al. [2010]. (f) Results of Zhu et al. [2016b]. (g) Results of Yu and Porikli [2016]. (h) Our method.

Ma et al. [2010] exploit position patches to hallucinate HR faces. This method requires the LR inputs to be precisely aligned with the reference images in the training dataset. As visible in Fig. 5.6(e), when there are alignment errors, it produces deformed faces. Moreover, as the upscaling factor increases, the correspondences between LR and HR patches become inconsistent. Hence, it suffers from severe block artifacts around the boundaries of different patches.

Zhu et al. [2016b] propose a deep cascaded bi-network for face hallucination, known as CBN. This method has its own aligning process that localizes facial landmarks used to fit a global face model. When the noise level is low, it can align LR faces based on the landmarks. However, when the noise is not negligible, it fails to localize landmarks thus produces ghosting artifacts (see Fig. 5.6(f)). Since noise impedes the landmark detection, we apply BM3D as a remedy. However, LR faces becomes smooth, and detecting facial landmarks becomes even difficult. Our observation is that CBN is not designed for noisy images.

Yu and Porikli [2016] develop a discriminative generative network to super-resolve very low resolution face images, known as URDGN. Their method also employs deconvolutional layers to upsample LR faces and a discriminative network is used to

Table 5.1: Quantitative evaluations on the entire test dataset. Different configurations: (1) STN+SR+BM3D, (2) STN+BM3D+SR, (3) BM3D+STN+SR. Here, SR is the compared super-resolution method. Our method does not use BM3D or a separate STN.

		PSNR		SSIM	
	Noise	5%	10%	5%	10%
1	Bicubic	17.93	17.77	0.51	0.49
	SRCNN	17.77	17.53	0.51	0.48
	Ma	17.98	17.90	0.51	0.50
	CBN	17.16	16.93	0.47	0.44
	URDGN	16.58	16.45	0.38	0.36
2	Bicubic	18.59	18.30	0.52	0.51
	SRCNN	18.59	18.32	0.53	0.51
	Ma	18.63	18.37	0.50	0.49
	CBN	18.34	18.26	0.52	0.52
	URDGN	16.95	16.79	0.41	0.40
3	Bicubic	17.87	17.63	0.52	0.50
	SRCNN	17.74	17.53	0.51	0.50
	Ma	17.86	17.65	0.49	0.48
	CBN	17.39	17.28	0.49	0.48
	URDGN	18.95	18.65	0.49	0.47
	Ours	**21.02**	**20.47**	**0.58**	**0.56**

force the generate network to produce sharper results. However, this method requires aligned images and cannot super-resolve unaligned faces. In addition, noise may damage the LR facial patterns, which may degrade the performance as visible in Fig. 5.6(g).

In comparison, our method reconstructs authentic facial details as shown in Fig. 5.6(h). We note that the input faces have different poses and facial expressions. Since our method applies multiple STNs on feature maps to align face images and remove noise simultaneously, it achieves much better alignment. With the help of the encoder, it obtains aligned and noise-free LR images. With its second decoder, it produces visually pleasing results, which are similar to the ground-truth faces as well. Our method does not need any landmark localization or any information about the noise. When the noise is low, it also attains superior performance.

5.6.3 Quantitative Comparison with the SoA

We quantitatively measure the performance of all methods on the entire test dataset in different noise levels by the average PSNR and the structural similarity (SSIM) scores. Table 5.1 presents that our method achieves superior performance in compar-

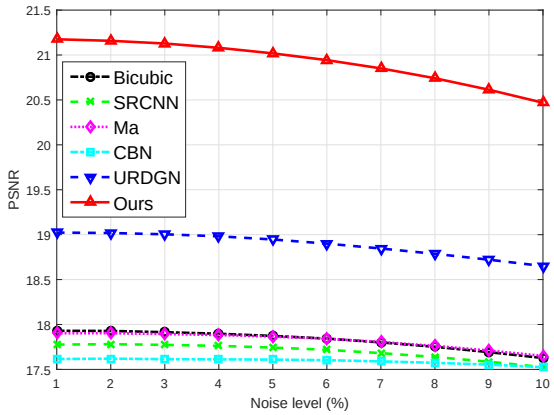
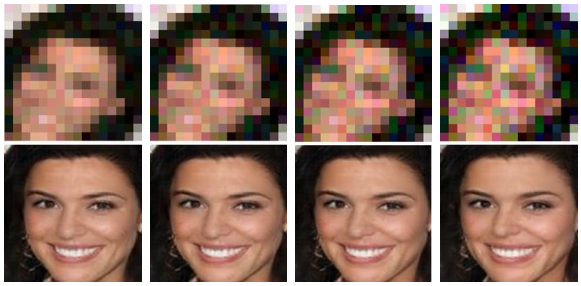


Figure 5.7: The PSNR curves of the state-of-the-art methods on synthetic test datasets with noise level from 1% to 10%.



(a) 3% (b) 5% (c) 7% (d) 10%

Figure 5.8: Visualization of our results for different noise levels. Please refer to Fig. 5.5(b) for the ground-truth HR image.

ison to other methods, outperforming the second best with a large margin of **1.82** dB in PSNR.

For an objective comparison with the SoA methods, we report results for three possible scenarios. In the first case, we first apply STN_0 to align noisy LR faces, then super-resolve the aligned LR images by the SoA, and finally use BM3D to remove the noise in the upsampled HR images. In the second case, we apply STN_0 followed by BM3D and then super-resolution. In the third case, we first denoise by BM3D, then align by STN_0, and finally super-resolve. When aligning noisy LR images, we train STN_0 with noisy LR faces. Otherwise, if we first use BM3D to reduce noise, we train STN_0 with noise-reduced LR faces.

Table 5.1 also indicates that simply denoising and then aligning, or aligning and then denoising LR faces cannot lead to good performance by the SoA methods.

Furthermore, we demonstrate that our method can successfully hallucinate faces in different noise levels in Fig. 5.8. When the noise level increases, our hallucinated faces remain consistent and retain their visual quality, which implies that our method is robust to noise variations.

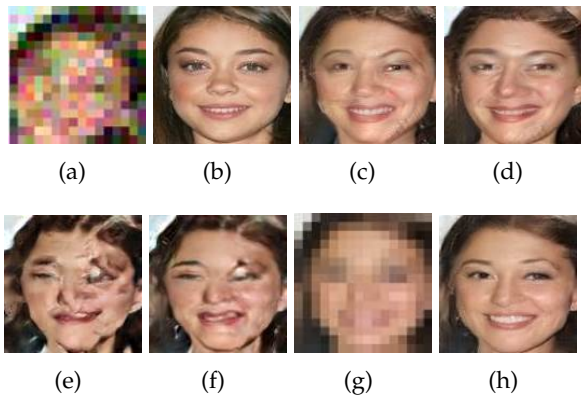


Figure 5.9: Illustration of necessity of the transformative encoder. (a) unaligned and noisy LR input with noise level 10%. (b) original HR image. (c) the output of DEC_1. (d) super-resolution of the downsampled result by DEC_1. (e) super-resolution of the downsampled result by DEC_2. (f) super-resolution of the downsampled result by the method of Yu and Porikli [2016]. (g) the output of our transformative encoder. (h) our final result.

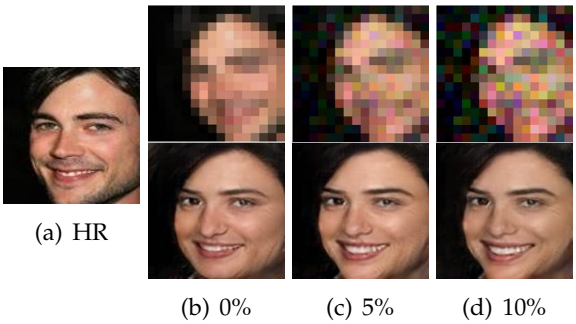


Figure 5.10: Visualization of our results for different noise levels. Notice that, in (b) our method is able to super-resolve a noise-free LR face.

Figure 5.7 shows the PSNR curves for different noise levels. We observe that our method achieves higher PSNRs over the other methods, and for lower noise levels it performs even better. Note that, we do not need to know the noise level in our algorithm.

5.7 Conclusion

We presented a transformative autoencoder network to super-resolve very low-resolution (16×16 pixels) unaligned and noisy face images with a challenging upsampling factor of 8×. We leverage on a new decoder-encoder-decoder architecture. Our networks jointly align, remove noise, and discriminatively hallucinate input images. Since our

(a) (b) (c) (d) (e) (f) (g) (h)

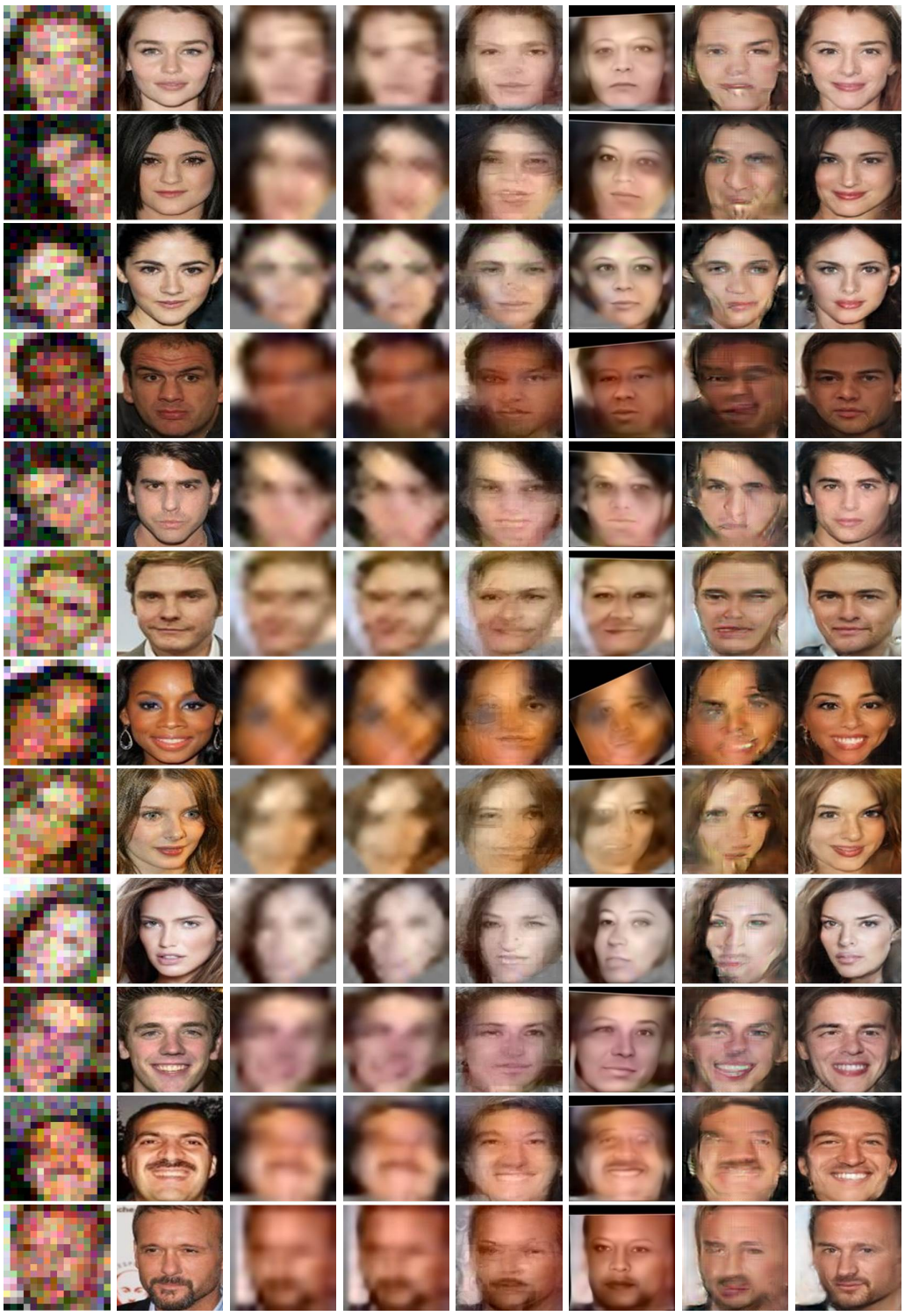
Figure 5.11: Comparison with the state-of-the-arts methods at the noise level 5%. (a) Unaligned and noisy LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Results of Dong et al. [2016a]. (e) Results of Ma et al. [2010]. (f) Results of Zhu et al. [2016b]. (g) Results of Yu and Porikli [2016]. (h) Our method.

method is agnostic to image noise, face pose, and spatial deformations, it is very practical. At the same time, it can generate rich and authentic facial details.

5.8 Appendix

5.8.1 Necessity of Transformative Encoder

In Fig. 5.9, we illustrate the necessity of our transformative encoder. We firstly downsample the output of the decoder DEC_1, and then apply three different ways to super-resolve the downsampled face image: (1) we employ our first decoder DEC_1 to upsample the dowmsampled face image, as shown in Fig. 5.9(d). (2) we employ our second decoder DEC_2 to upsample the downsampled version, as shown in Fig. 5.9(e). (3) we employ the method of Yu and Porikli [2016] to super-resolve the downsampled LR image, as shown in Fig. 5.9(f). As shown in Fig. 5.9(d), Fig. 5.9(e) and Fig. 5.9(f), the artifacts still remain in the upsampled HR face images. Hence, simply downsampling the upsampled HR faces and then super-resolving the downsampled images



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 5.12: Comparison with the state-of-the-arts methods at the noise level 10%. (a) Unaligned and noisy LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Results of Dong et al. [2016a]. (e) Results of Ma et al. [2010]. (f) Results of Zhu et al. [2016b]. (g) Results of Yu and Porikli [2016]. (h) Our method.

cannot attain high-quality HR face images. By contrast, our method can remove the artifacts and output a realistic HR face image, as illustrated in Fig. 5.9(h).

5.9 Additional Experimental Results

Figure 5.10 illustrates another example that our method can upsample an unaligned noisy LR image regardless of noise levels. Furthermore, as shown in Fig. 5.10(b), our method can super-resolve unaligned and noise-free LR face images as well.

In Fig. 5.6, we have compared with the state-of-the-art methods when the noise level is 10%. In order to demonstrate our method is able to super-resolve LR images at different noise levels, we also compare with the state-of-the-art methods at the noise level 5%, as shown in Fig. 5.11. Notice that, the visual quality of our results does not degrade as noise levels vary. Furthermore, we also show some extra experimental results in Fig. 5.12. In these experiments, we also demonstrate that our method is able to hallucinate face images regardless of facial expressions and head poses.

Chapter 6

Hallucinating Unaligned Face Images by Multiscale Transformative Discriminative Networks

6.1 Foreword

Previous chapters only address low-resolution face images at a fixed resolution. When the resolutions of input face images are larger than the desired resolution of the upsampling networks, our previous methods need to downsize the input images. In this way, some high-frequency details of input images will be discarded, thus leading to inferior super-resolution performance. In this chapter, we develop a multiscale upsampling network which is able to explore all the information in input images for face hallucination. Furthermore, we employ a feature-wise constraint, known as perceptual loss, to enforce the upsampled HR faces to share similar facial features to the ground-truths. By doing so, the upsampled facial features are much closer to their ground-truth features. Therefore, our multiscale network outperforms our previous networks qualitatively and quantitatively.

This chapter has been submitted to *International Journal of Computer Vision* as a journal paper: Xin Yu, Basura Fernando, Fatih Porikli, Richard Hartley: Hallucinating Unaligned Face Images by Multiscale Transformative Discriminative Networks.

6.2 Abstract

Conventional face hallucination methods heavily rely on accurate alignment of low-resolution (LR) faces before upsampling them. Misalignment often leads to deficient results and unnatural artifacts for large upscaling factors. However, due to the diverse range of poses and different facial expressions, aligning an LR input image, in particular when it is tiny, is severely difficult. In addition, when the resolutions of LR input images vary, previous deep neural network based face hallucination methods require input images at a fixed resolution. Downsampling LR input faces to a re-

quired resolution will lose high-frequency information of the original input images. This may lead to suboptimal super-resolution performance for the state-of-the-art face hallucination networks. To overcome these challenges, we present an end-to-end multiscale transformative discriminative neural network (MTDN) devised for super-resolving unaligned and very small face images of different resolutions ranging from 16×16 to 32×32 pixels in a unified framework. Our proposed network embeds spatial transformation layers to allow local receptive fields to line-up with similar spatial supports, thus obtaining a better mapping between LR and HR facial patterns. Furthermore, we incorporate a class-specific loss designed to classify upright realistic faces in our objective through a successive discriminative network to improve the alignment and upsampling performance with semantic information. Extensive experiments on a large face dataset show that the proposed method significantly outperforms the state-of-the-art.

6.3 Introduction

Face images provide vital information for visual perception and identity analysis. Nonetheless, when the resolution of the face image is very small (*e.g.* in typical surveillance videos), there is little information that can be inferred from it. Very low-resolution (LR) face images not only degrade the performance of the recognition systems but also impede human interpretation. This challenge motivates the reconstruction of high-resolution (HR) images from given LR counterparts, known as face hallucination, and has attracted increasing interest in recent years.

Previous face hallucination methods based on holistic appearance models [Liu et al., 2001; Baker and Kanade, 2002; Wang and Tang, 2005; Liu et al., 2007; Hennings-Yeomans et al., 2008; Ma et al., 2010; Yang et al., 2010; Li et al., 2014; Arandjelović, 2014; Kolouri and Rohde, 2015] demand LR faces to be precisely aligned beforehand. However, aligning LR faces to appearance models is not a straightforward task itself, and more often, it requires expert feedback when the input image is small. Regarding pose and expression variations naturally exist in LR face images, aligning LR faces by state-of-the-art automatic alignment techniques [Zhu and Ramanan, 2012; Bulat and Tzimiropoulos, 2017a] which usually assume facial landmarks are visible and detectable would be even more difficult. As a result, the performance of face hallucination degrades severely. Such a broad spectrum of pose and expression variations also makes learning a comprehensive appearance model even harder. For instance, Principal Component Analysis (PCA) based schemes become critically ineffective to learn a reliable face model while aiming to capture different in- and out-of-plane rotations, scale changes, translational shifts, and facial expressions. As a result, these methods lead to unavoidable artifacts when LR faces are misaligned or depict different poses and facial expressions from the base appearance model. Moreover, once appearance models are learned, input LR faces at different resolutions need to be downscaled to fit the input size of the learned models. By doing so, some high-frequency information of LR faces will be lost and different LR faces tend to be

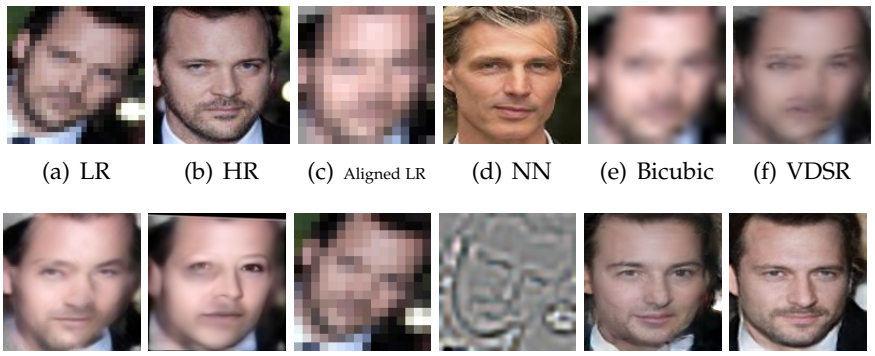


Figure 6.1: Comparison of our method with the CNN based super-resolution. (a) The input 24×24 LR image. (b) The original 128×128 HR image. (c) Aligned LR image of (a). The resolution of the aligned LR image is 16×16 pixels since STN_0 only outputs a fixed resolution for all images. (d) The corresponding HR version of the nearest neighbor (NN) of (c) in the training set. (e) Bicubic interpolation of (c). (f) The image generated by a CNN based generic super-resolution, *i.e.*, VDSR [Kim et al., 2016a]. We retrain VDSR with face images to better capture LR facial patterns in super-resolution. (g) The image upsampled by a GAN based generic super-resolution method, *i.e.*, SRGAN [Ledig et al., 2017]. Here, SRGAN is also fine-tuned on face images. (h) The image super-resolved by a state-of-the-art face hallucination method, *i.e.*, CBN [Zhu et al., 2016b]. (i) The low-frequency component of (a). (j) The high-frequency component of (a). (k) The upsampled face by our previous method [Yu and Porikli, 2017b], which only uses the image (i) as input. (l) The result of our MTDN.

indistinguishable at a lower resolution. Thus, the downscaling operation may result in suboptimal super-resolution performance.

Rather than learning holistic appearance models, many methods upsample *facial components* by transferring references from an HR training dataset and then blending them into an HR version [Tappen and Liu, 2012; Yang et al., 2013, 2017a]. Although these methods do not need LR face images to be aligned in advance or to resize input images to a fixed resolution, they expect the resolution of input faces to be sufficient enough for detecting the facial landmarks and parts. When the resolution is very low, they fail to localize the components accurately, thus producing non-realistic faces. In other words, the facial component based methods are unsuitable to upsample very low-resolution faces.

By better exploring the information available in the natural structure of face images, appearance similarities between individuals and emerging large-scale face datasets [Huang et al., 2007; Liu et al., 2015], it becomes possible to derive competent models to reconstruct authentic $4\times$ $\sim 8\times$ magnified HR face images. Deep neural networks, in particular convolutional neural networks (CNN), are inherently suitable for learning from large-scale datasets. Very recently, CNN based generic patch

super-resolution methods [Dong et al., 2016a; Kim et al., 2016a; Ledig et al., 2017] have been proposed without focusing on any image class. A straightforward retraining (fine-tuning) of the networks, *i.e.*, VDSR [Kim et al., 2016a] and SRGAN [Ledig et al., 2017] with face images cannot produce realistic and visually pleasant results, as shown in Fig. 6.1(f) and Fig. 6.1(g), because these networks cannot address misalignments of LR inputs inherently. Misalignments of LR faces lead to the degradation of the super-resolution performance.

Recently, deep neural network based face hallucination methods have been proposed, and achieve state-of-the-art performance [Yu and Porikli, 2016, 2017a,b, 2018; Zhu et al., 2016b; Huang et al., 2017a]. However, those networks are only designed to super-resolve fixed-sized LR face images. When the input images are larger than the desired input size of the networks, images are required to be downsampled to fit the input size of the networks. After downsampling, some high-frequency components are lost. Thus, those deep learning based methods cannot fully exploit all the information of input images and output suboptimal results.

In this paper, we present a new multiscale transformative discriminative neural network (MTDN) to overcome the above issues. Our proposed network is able to super-resolve a range of small and unaligned face images (*i.e.*, from 16×16 to 32×32 pixels) to HR images of 128×128 pixels. In particular, when the resolution of input images is 16×16 pixels, we upsample LR faces by a remarkable upscaling factor 8×, where we reconstruct 64 pixels for each single pixel of an input LR image. Unlike previous works [Yu and Porikli, 2016, 2017a,b], when the resolutions of input images are larger than the input size of the networks, *i.e.*, 16×16 pixels, our network can preserve all the information of input face images. Specifically, our MTDN develops two branches to receive an downscaled LR input image as well as its residuals. In this fashion, our MTDN is able to exploit the residuals from the downscaled images for super-resolution. In order to retain the global structure of faces while being able to reconstruct instance specific details, we use whole face images to train our networks.

Our network consists of two components: an upsampling network that comprises deconvolutional and spatial transformation network [Jaderberg et al., 2015] layers, and a discriminative network. The upsampling network is designed to progressively improve the resolution of the latent feature maps at each deconvolutional layer. We do not assume the LR face is aligned in advance. Instead, we compensate for any misalignment and changes through the spatial transformation network layers that are embedded into the upsampling network. In order to avoid the loss of information caused by downsampling LR face images, we separate LR images into two branches, *i.e.*, a low-frequency branch and a high-frequency branch. For instance, we downsample an LR image of 24×24 pixels to 16×16 pixels to obtain the low-frequency image as well as upsample its residual image (*i.e.*, an image is subtracted from the original LR image by the resized low-frequency image) to 32×32 pixels to achieve the high-frequency image. Then, we extract features from these two branches and then combine the feature maps for further super-resolution without losing information of inputs. One can use the pixel-wise intensity similarity between the estimated and the ground-truth HR face images as the objective function in the training stage. However,

when the upscaling factor becomes larger, employing only the pixel-wise intensity similarity causes over-smoothed outputs. In order to force the upsampled faces to share facial features similar to their ground-truth counterparts, we employ the perceptual loss [Johnson et al., 2016]. Since face hallucination is an under-determined problem, there would be one-to-many mappings between image intensities and features. Thus, the upsampled HR faces may not be sharp and realistic-looking enough. To make the upsampled HR faces realistic, we incorporate class similarity information that is provided by a discriminative network. We back-propagate the discriminative errors to the upsampling network. Our end-to-end solution allows fusing the pixel-wise, feature-wise and class-wise information in a manner robust to spatial transformations and obtaining a super-resolved output with much richer details.

Overall, our main contributions have four aspects:

- We present a novel end-to-end multiscale transformative discriminative network (MTDN) to super-resolve very low-resolution face images to HR face images of 128×128 pixels, where the upscaling factor ranges from 4× to 8×.

- We propose a unified framework which super-resolves LR faces at different resolutions, *i.e.*, from 16×16 to 32×32 pixels, and outputs aligned upscaled HR faces by a single deep neural network.

- In order to accept different sizes of LR input face images, we firstly divide an input image into a low-frequency component and a high-frequency residual one, and then design a two branch network to receive these two components for upsampling. In this manner, we do not need to discard the residuals of the downsample LR faces so as to fit the input size of deep neural networks, thus avoiding losing information of inputs.

- For tiny input images where landmark based methods inherently fail, our method is able to align and hallucinate an unaligned LR face image without requiring precise alignment in advance, which makes our method practical.

This paper is an extension of our previous conference papers [Yu and Porikli, 2016, 2017a,b]. In this paper, we propose a new unified framework to super-resolve LR faces at different resolutions. Since our previous methods need to downsample LR faces at different resolutions to a fixed resolution, this downsampling operation lose some high-frequency details of the LR inputs, *i.e.*, residual images. Thus, they may lead to suboptimal super-resolution results, as shown in our experimental part. Different from our previous works, the proposed network can preserve all the information of LR faces by our newly proposed multiscale network, thus achieving better super-resolution performance. In addition, we also conduct more comprehensive qualitative and quantitative experiments and discussions on each component of our proposed network.

6.4 Related Work

Super-resolution can be classified into two categories: generic super-resolution methods and class-specific super-resolution methods. When upsampling LR images, generic methods employ priors that ubiquitously exist in natural images without considering any image class information. Class-specific methods aim to exploit statistical information of objects in a certain class and they usually attain better results than generic methods, *e.g.*, the task of super-resolving LR face images.

Generic single image super-resolution methods generally have three types: interpolation based methods, image statistics based methods and learning-based methods. Interpolation based methods such as linear and non-linear upsampling are simple and computationally efficient, but they may produce overly smooth edges and fail to generate HR details as the upscaling factor increases. Image statistics based methods employ natural image priors to enhance the details of upsampled HR images, such as image gradients are sparse and follow heavy-tailed distributions [Tappen et al., 2003], but these methods are also limited to smaller magnification factors [Lin and Shum, 2006].

Learning-based methods demonstrate their potentials to exceed this limitation of the maximum upscaling factor by learning a mapping from a large number of LR/HR pairs [Lin et al., 2008]. Several methods [Glasner et al., 2009; Freedman and Fattal, 2010; Singh et al., 2014; Huang et al., 2015] exploit self-similarity of patches in an input image to generate HR patches. Freeman et al. [2002] and Hong Chang et al. [2004] construct LR and HR patch pairs from a training dataset, and then infer high-frequency details by searching the corresponding HR patch of the nearest neighbor of an input LR patch. Yang et al. [2010] employ sparse representation to construct the corresponding LR and HR dictionaries and then reconstruct HR output images by the sparse coding coefficients inferred from LR images. Gu et al. [2015] apply convolutional sparse coding instead of patch-based sparse coding to reconstruct HR images.

Deep learning based super-resolution methods have been also proposed. Dong et al. [2016a] incorporate convolutional neural networks to learn a mapping function between LR and HR patches from a large-scale dataset. Motivated by this idea, the follow-up works [Kim et al., 2016a; Ledig et al., 2017; Kim et al., 2016b; Shi et al., 2016; Lai et al., 2017; Tai et al., 2017] try to explore deeper network architectures to improve super-resolution performance. Since many different HR patches may correspond to one LR patch, output images may suffer from artifacts at the intensity edges. In order to reduce the ambiguity between the LR and HR patches, Bruna et al. [2016] explore the statistical information learned from a deep convolutional network to reduce ambiguity between LR and HR patches. Johnson et al. [2016] propose a perceptual loss to constrain the feature similarity by a pre-trained deep neural network. Ledig et al. [2017] employ the framework of generative adversarial networks (GAN) [Goodfellow et al., 2014] to enhance image details by combining an image intensity loss and an adversarial loss. Since those generic super-resolution methods do not take class-specific information into account, they still suffer over-

smoothed results when input sizes are tiny and magnification factors are large.

Class-specific super-resolution methods further exploit the statistical information in the image categories, thus leading to better performance. When the class is faces, they are also called face hallucination methods [Baker and Kanade, 2000; Liu et al., 2001; Baker and Kanade, 2002].

The seminal works [Baker and Kanade, 2000, 2002] build the relationship between facial HR and LR patches using Bayesian formulation such that high-frequency details can be transferred from the dataset for face hallucination. It can generate face images with richer details. However, artifacts also appear due to the possible inconsistency of the transferred HR patches. Wang and Tang [2005] apply PCA to LR face images, and then hallucinate HR face images by an Eigen-transformation of LR images. Although their method is able to magnify LR images by a large scaling factor, the output HR images suffer from ghosting artifacts when the HR images in the exemplar dataset are not precisely aligned. Liu et al. [2007] enforce linear constraints for HR face images using a subspace learned from the training set via PCA, and a patch-based Markov Random Field is proposed to reconstruct the high-frequency details in the HR face images. To mitigate artifacts caused by misalignments, a bilateral filtering is used as a post-processing step. Kolouri and Rohde [2015] employ optimal transport in combination with subspace learning to morph an HR image from the LR input. Their method still requires that face images in the dataset are precisely aligned and the test LR images have the same poses and facial expressions as the exemplar HR face images. Instead of imposing global constraints, Ma et al. [2010] super-resolve local HR patches by a weighted average of exemplar HR patches and the weights are learned from the corresponding LR patches. Rather than hallucinating HR patches in terms of image intensities, Li et al. [2014] resort to sparse representation on the local regions of faces. However, blocky artifacts may appear as magnification factors become large.

To handle various poses and expressions, Tappen and Liu [2012] integrate SIFT flow [Liu et al., 2011] to align facial components in LR images. Their method performs competently when the training face images are highly similar to the test face image in terms of identity, pose, and expression. Yang et al. [2013] and Yang et al. [2017a] first localize facial components, and then upsample each component by matching gradients with respect to the similar HR facial components in the exemplar dataset. However, these methods rely on accurate facial landmark points that are usually unavailable when the image size is very small. More comprehensive literature review of early face hallucination works can be referred to the work [Wang et al., 2014].

Deep learning based face hallucination methods are proposed to fully exploit the face structure and priors from emerging large-scale face datasets [Liu et al., 2015; Huang et al., 2007; Yang et al., 2016]. Zhou and Fan [2015] propose a convolutional neural network (CNN) to extract facial features and recover facial details from the extracted features. Yu and Porikli [2018] combine deconvolutional and convolutional layers to upsample LR face images, but they resort a post-processing step [Yu et al., 2014] to improve the visual quality of the super-resolved faces. Later, Yu and Porikli

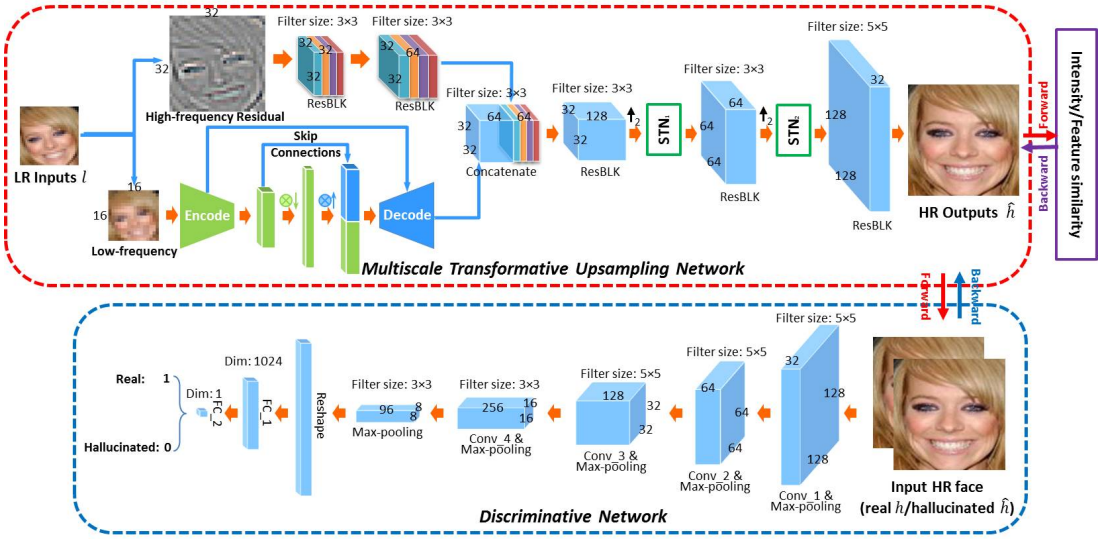


Figure 6.2: Our MTDN consists of two parts: an upsampling network (in the red frame) and a discriminative network (in the blue frame).

[2016] explore a discriminative generative network to super-resolve aligned LR face images in an end-to-end manner while Huang et al. [2017a] estimate wavelet coefficients for a face upsampled by a generative adversarial network and then reconstruct the HR image from the estimated coefficients. Xu et al. [2017] employ a multi-class adversarial loss in the framework of generative adversarial networks to super-resolve LR blurry face and text images. Dahl et al. [2017] exploit an autoregressive generative model [Van Den Oord et al., 2016] to hallucinate pre-aligned LR face images. In order to mitigate the ambiguity of the mappings between LR and HR faces, Yu et al. [2018] embed high-level semantic information, *i.e.*, face attributes, into the procedure of face hallucination. To relax the requirement of face alignment, Bulat and Tzimiropoulos [2018] present a constraint that the landmarks of the upsampled faces should be close to the landmarks detected in their ground-truth images. Since ground-truth landmarks are not provided in the training stage and erroneous localization of landmarks may lead to distorted upsampled face images, their results are only restricted to 64×64 pixels and facial details are not sharp enough. Zhu et al. [2016b] develop a cascade bi-network to super-resolve unaligned LR faces, where facial components are localized first and then upsampled. Chen et al. [2018] present a two-stage network, where low-frequency components of LR face are first super-resolved and then face priors (*i.e.*, facial component locations) are also employed to enrich facial details. However, those methods may produce ghosting artifacts when the facial component localization is erroneous. Towards the same goal, our previous works [Yu and Porikli, 2017a,b; Yu et al., 2018] embed multiple spatial transformer networks [Jaderberg et al., 2015] into the upsampling networks. However, those networks are trained in a fixed input resolution, and thus LR faces at different resolutions have to be resized (*i.e.*, downsampling) to meet the input resolution of the networks. Therefore, these

(a) HR (b) LR faces of different resolutions (c) Our results of (b) (d) HR (e) LR faces of different resolutions (f) Our results of (e)

Figure 6.3: Illustrations of our results with respect to the different resolutions of LR input images. (a)(d) Ground-truth HR face images. (b)(e) unaligned LR face images. From left to right, the resolutions of the images are 16×16, 24×24 and 32×32. (c) Our results of (b). From left to right, the corresponding PSNRs are 22.79 dB, 23.59 dB and 24.63 dB. (f) Our results of (e). From left to right, the corresponding PSNRs are 17.80 dB, 19.96 dB and 21.94 dB.

methods may lose information of input images and introduce extra ambiguity due to the downscaling operation.

6.5 Proposed Method: MTDN

6.5.1 Background

Our face hallucination method is motivated by the generative adversarial networks [Goodfellow et al., 2014] since they can generate an face image from random noise represented by a fairly low-dimensional vector. Specifically, the generative model $\mathcal{G}$ takes a noise vector z from a distribution $P_{noise}(z)$ as an input and then outputs an image $\hat{x}$. The discriminative model $\mathcal{D}$ takes an image stochastically chosen from either the generated image $\hat{x}$ or the real image x drawn from the training dataset with a distribution $P_{data}(x)$ as an input. $\mathcal{D}$ is trained to output a scalar probability, which is large for real images and small for generated images from $\mathcal{G}$. The generative model $\mathcal{G}$ is learned to maximize the probability of $\mathcal{D}$ making a mistake. Thus a minmax objective is used to train these two models simultaneously,

$$\min_{\mathcal{G}}\max_{\mathcal{D}}\mathbb{E}_{x\sim P_{data}(x)}\log\mathcal{D}(x)+\mathbb{E}_{z\sim P_{noise}(z)}\log(1-\mathcal{D}(\mathcal{G}(z))).$$

This equation encourages $\mathcal{G}$ to fit $P_{data}(x)$ so as to fool $\mathcal{D}$ with its generated samples $\hat{x}$. However, we cannot directly employ the above equation for the face hallucination task since GAN takes a fixed size noise vector as input to learn the distribution on the training dataset. In contrast, the input for our face super-resolution task is an LR face image, and its resolution is not fixed either. LR faces also undergo rotations, translations and scale changes.

In this paper, we propose a transformative discriminative neural network (MTDN) which achieves the image alignment and super-resolution simultaneously. Furthermore, our MTDN accepts LR input images in various sizes without losing image information. The entire pipeline is shown in Fig. 6.2.

6.5.2 Network Architecture

Our MTDN consists of two parts: a multiscale transformative upsampling network that combines autoencoder, spatial transformation network layers, upsampling layers and residual block layers, and a discriminative network that is composed of convolutional layers, max-pooling layers, and fully-connected layers. The multiscale transformative upsampling network is designed for receiving and super-resolving LR images at different resolutions while the discriminative network is developed to force the super-resolved faces to be realistic.

6.5.2.1 Multiscale Transformative Upsampling Network

Reception for LR Images in a Multiscale Manner: State-of-the-art CNN based super-resolution networks [Yu and Porikli, 2016, 2017a,b; Zhu et al., 2016b; Bulat and Tzimiropoulos, 2018; Chen et al., 2018] only accept LR inputs in a fixed resolution, *i.e.*, 16×16 pixels. When the resolutions of LR images are larger than the desired resolution, those methods need to downsample input images. However, downsampling input images may result in the loss of high-frequency details of LR inputs as well as more ambiguous mappings between LR and HR face images in super-resolution. In addition, we assume that the resolutions of LR images are smaller than 32×32 pixels. Otherwise, LR images can provide enough resolution for human observation and computer analysis. Hence, we only focus on LR images whose resolutions are smaller than 32×32 pixels in this paper.

Inspired by the Laplacian pyramid, we decompose an image into two components: a low-frequency part and a high-frequency part. We downsample an input image to 16×16 pixels as our low-frequency part, as illustrated in Fig. 6.1(i). The high-frequency part is obtained by subtracting the input image by the interpolated low-frequency components. Then, we upsample the high-frequency component to 32×32 pixels, as visible in Fig. 6.1(j). In this way, our transformative upsampling network can receive LR face images at different resolutions while preserving high-frequency residual details of the inputs for super-resolution.

In order to combine the information of the high-frequency and low-frequency branches together, we extract feature maps from the images of those tow branches and then concatenate the feature maps for further super-resolution. Specifically, we firstly employ an autoencoder with skip connections to extract features from the low-frequency component and then upsample the feature maps by a deconvolutional layer. After the deconvolutional layer, the resolution of the low-frequency branch has been increased as the same as the resolution of the high-frequency branch. Rather than directly combining the high-frequency residual component with the feature

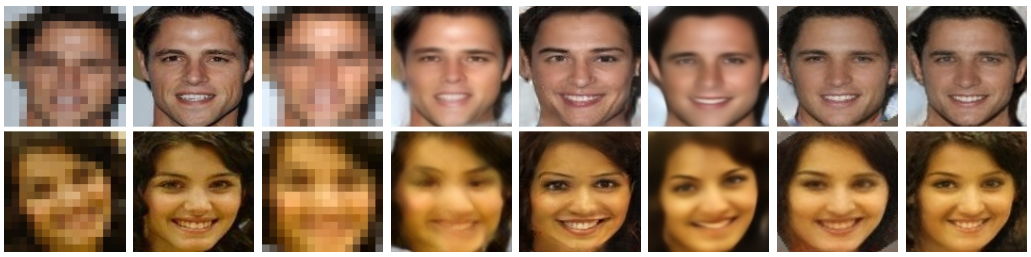
maps of the low-frequency component, we apply two cascaded residual blocks to extract features from the high-frequency component as well. Then, we concatenate the feature maps extracted from the high-frequency residual component with the upsampled feature maps of the low-frequency component and then employ a residual block to fuse the concatenated feature maps.

As shown in Fig. 6.3, our network is able to super-resolve LR face images at different resolutions. Note that, we do not need to fine-tune our network on images of different sizes. As expected, the PSNRs of our upsampled results become higher as the resolutions of LR faces increase. This indicates our network exploits all the information in LR input images for super-resolution.

Upsampling Layers: After obtaining the concatenated features maps of input images, we further super-resolve the feature maps by the deconvolutional layers and residual blocks. The deconvolutional layer, also known back-convolutional layer, can be made of a cascade of an upsampling layer and a convolutional layer, or a convolutional layer with a fractional stride [Zeiler et al., 2010; Zeiler and Fergus, 2014]. Therefore, the resolution of the output of the deconvolutional layers is larger than the resolution of its input. To reduce potential blocky artifacts caused by deconvolutional layers [Yu and Porikli, 2018] as well as increase the capacity of the network, we cascade a residual block after each deconvolutional layer as our upsampling layer.

Spatial Transformation Layers: The spatial transformation network (STN) is proposed by Jaderberg et al. [2015]. It can estimate the motion parameters of images, and warp images to the canonical view. In our architecture, the spatial transformation network layers are represented as the green boxes in Fig. 6.2. These layers contain three modules: a localization module, a grid generator module, and a sampler. The localization module consists of a number of hidden layers and outputs the transformation parameters of an input relative to the canonical view. The grid generator module creates a sampling grid according to the estimated parameters. Finally, the sampler module maps the input onto the generated grid by bilinear interpolation.

Since we focus on in-plane rotations, translations, and scale changes without requiring a 3D face model, we employ the similarity transformation for face alignment. Although STNs can warp images, it is not straightforward to use them directly to align very LR face images. As shown in Fig. 6.1(c), directly applying an STN to align LR images causes distortion artifacts due to the difficulty of spatial transformation estimation on very LR faces. There are several factors needed to be considered: (i) After the alignment of LR images, facial patterns are blurred due to the resampling of the aligned faces by bilinear interpolation. (ii) Since the resolution is very low and a wide range of poses exists, estimating spatial transformations on such small face images may lead to alignment errors. (iii) Due to the blur and alignment errors, the upsampling network may fail to generate realistic HR faces. (iv) If STNs are employed to the two branches separately, the estimated transformation parameters of these two branches may be different. This will result in misalignments between the low-frequency component and the high-frequency residual components. As a result of the misalignments, distortion artifacts or ghosting artifacts may appear in the final results. Therefore, we employ STNs to align the concatenated feature maps. In this



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6.4: Illustrations of different losses for super-resolution. (a) The input 16×16 LR images. (b) The original 128×128 HR images. (c) The aligned LR images. (d) The upsampled faces by SRGAN [Ledig et al., 2017]. Here, SRGAN is applied to the aligned LR faces. Since SRGAN is trained on generic images patches, we re-train SRGAN on whole face images. (e) The face images super-resolved by our previous method [Yu and Porikli, 2017b]. (f) The super-resolved faces by $\mathcal{L}_{pix}$. (g) The super-resolved faces by $\mathcal{L}_{pix} + \mathcal{L}_{feat}$. (h) The super-resolved faces by $\mathcal{L}_{pix} + \mathcal{L}_{feat} + \mathcal{L}_{\mathcal{U}}$. Here, we omit the trade-off weights for simplicity.

way, we can align the low-frequency and high-frequency parts simultaneously.

Instead of using a single STN to align LR face images, we employ multiple STN layers to line up the feature maps. Using multiple layers significantly reduces the load on each spatial transformation network and further reduces the errors of misalignments. In addition, resampling feature maps by multiple STN layers prevents from damaging or blurring input LR facial patterns. Since STN layers and the upsampling layers are interwoven together (rather than being two individual networks), the upsampling network can learn to eliminate the undesired effects of misalignment in the training stage.

6.5.2.2 Discriminative Network

In generic super-resolution [Kim et al., 2016a,b], only the ℓ_2 regression loss, also known as Euclidean distance loss, is employed to constrain the similarity between the upsampled HR images and their original HR ground-truth versions. However, as reported in our previous work [Yu and Porikli, 2016], deconvolutional layers supervised by a ℓ_2 loss tend to produce over-smoothed results. As seen in Fig. 6.4(f), the hallucinated faces are not sharp enough because the common parts learned by the upsampling network are averaged from similar components shared by different individuals. Thus, there is a quality gap between the real face images and the hallucinated faces. To bridge this gap, we inject class information. We integrate a discriminative network to distinguish whether the generated image is classified as an upright real face image or not. A similar idea is employed in the generative adversarial networks [Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015], which are designed to generate a new face. The architecture of the discriminative

network is shown in the blue frame of Fig. 6.2. It consists of convolutional, max-pooling, fully-connected and non-linear transformation layers. We employ a binary cross-entropy as the loss function to distinguish whether the input HR faces are sampled from super-resolved or real images. We backpropagate the discriminative error to revise the coefficients of the multiscale transformative upsampling network (for simplicity, we also refer to it as the upsampling network), which enforces the facial parts learned by the deconvolutional layers to be as sharp and authentic as real face images. Furthermore, the use of class information also facilitates the performance of the STN layers for face alignment since only upright faces are classified as valid faces. Therefore, our discriminative network also determines whether the faces are upright or not. As shown in Fig. 6.4(h), with the help of the discriminative information, the hallucinated face embodies more authentic, much sharper and better aligned details.

6.5.3 Training Details of MTDN

We construct LR and HR face image pairs $\{l_i, h_i\}$ as our training dataset, where h_i represents the aligned HR face images (only eyes are aligned), and l_i is the synthesized LR face images downsampled from h_i. Notice that, different from our previous works, the resolutions of input LR images l_i are different. As mentioned in Sec. 6.5.2.1, the input LR faces l_i are further decomposed into two components: the low-frequency component l_i^L of size 16×16 pixels and the high-frequency residual component l_i^H of size 32×32 pixels.

In training our MTDN, we not only employ the conventional pixel-wise intensity similarity, known as pixel-wise ℓ_2 loss, but also the feature-wise similarity, known as perceptual loss [Johnson et al., 2016]. The perceptual loss is able to enforce the upsampled facial characteristics to resemble their ground-truth counterparts. Even though pixel-wise and feature-wise similarity are applied in training our network, learning a mapping between LR and HR face images is still an ill-posed problem. Our network will tend to output blurry results to lower the training losses. Thus, in the testing stage, the upsampling network generates blurry faces. Similar to our previous works [Yu and Porikli, 2016, 2017a], the adversarial loss is also employed to attain visually appealing HR face images.

6.5.3.1 Pixel-wise Intensity Similarity Loss

We enforce the generated HR face $\hat{h}_i$ to be similar to its corresponding ground-truth h_i in terms of image intensities. Thus we employ a pixel-wise ℓ_2 regression loss $\mathcal{L}_{pix}$ to impose the appearance similarity constraint, expressed as:

$$\begin{aligned} \mathcal{L}_{pix} &= \mathbb{E}_{(\hat{h}_i, h_i)\sim p(\hat{h},h)} \|\hat{h}_i - h_i\|_F^2 \\ &= \mathbb{E}_{(l_i, h_i)\sim p(l,h)} \|\mathcal{U}_t(l_i^L, l_i^H) - h_i\|_F^2, \end{aligned} \tag{6.1}$$

where t and $\mathcal{U}$ are the parameters and the output of the upsampling network, $p(\hat{h}, h)$ represents the joint distribution of the frontalized HR faces and their corresponding

frontal HR ground-truths, $p(l,h)$ indicates the joint distribution of the LR and HR face images in the training dataset, and the LR input l_i is decomposed into l_i^L and l_i^H before fed into the upsampling network. Here, we do not distinguish the parameters of the upsampling layers and the STN layers because all the parameters are learned simultaneously. We employ t to represent all the parameters in our multiscale transformative upsampling network.

6.5.3.2 Feature-wise Similarity Loss

As illustrated in Fig. 6.4(f), the pixel-wise ℓ_2 loss leads to over-smoothed super-resolved results. Therefore, we employ a feature-wise similarity loss to force the super-resolved HR faces to share the same facial features as their ground-truth counterparts. The feature-wise loss $\mathcal{L}_{feat}$ measures Euclidean distance between the feature maps of super-resolved and ground-truth HR faces which are extracted by a deep neural network, written as:

$$\begin{aligned}\mathcal{L}_{feat} &= \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\Phi(\hat{h}_i)-\Phi(h_i)\|_F^2 \\ &= \mathbb{E}_{(l_i,h_i)\sim p(l,h)}\|\Phi(\mathcal{U}_t(l_i^L,l_i^H))-\Phi(h_i)\|_F^2,\end{aligned} \tag{6.2}$$

where $\Phi(\cdot)$ denotes feature maps extracted by the ReLU32 layer in VGG-19 [Simonyan and Zisserman, 2014], which gives good empirical performance in our experiments.

6.5.3.3 Class-wise Discriminative Loss

In order to achieve visually appealing results, we infuse class-specific discriminative information into our upsampling network by exploiting a discriminative network, similar to our previous works [Yu and Porikli, 2016, 2017a,b]. Since our goal is to output realistic HR faces, the upsampled face images should be able to fool the discriminative network. In other words, the upsampling network makes the discriminative network fail to distinguish generated faces from real ones. To do so, we enforce the super-resolved HR frontal faces to lie on the manifold of real HR face images. The discriminative network is used to classify real and super-resolved faces, and thus its objective function is written as:

$$\begin{aligned}\mathcal{L}_{\mathcal{D}} &= -\mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\left[\log \mathcal{D}_d(h_i)+\log(1-\mathcal{D}_d(\hat{h}_i))\right] \\ &= -\mathbb{E}_{h_i\sim p(h)}\log \mathcal{D}_d(h_i) - \mathbb{E}_{\hat{h}_i\sim p(\hat{h})}\log(1-\mathcal{D}_d(\hat{h}_i)) \\ &= -\mathbb{E}_{h_i\sim p(h)}\log \mathcal{D}_d(h_i) \\ &\quad -\mathbb{E}_{l_i\sim p(l)}\log(1-\mathcal{D}_d(\mathcal{U}_t(l_i^L,l_i^H))),\end{aligned} \tag{6.3}$$

where d represents the parameters of the discriminative network, $p(l)$, $p(h)$ and $p(\hat{h})$ indicate the distributions of the LR, HR ground-truth and upsampled faces respectively, and $\mathcal{D}_d(h_i)$ and $\mathcal{D}_d(\hat{h}_i)$ are the outputs of the discriminative network.

To make the discriminative network distinguish hallucinated faces from real ones, we minimize the loss $\mathcal{L}_{\mathcal{D}}$ and update the parameters d.

Meanwhile, our upsampling network aims to fool the discriminative network. It needs to generate realistic HR face images and make the discriminative network classify the super-resolved faces as real faces. Therefore, the objective function of our upsampling network is written as:

$$\begin{aligned}\mathcal{L}_{\mathcal{U}} &= -\mathbb{E}_{\hat{h}_i \sim p(\hat{h})} \log(\mathcal{D}(\hat{h}_i)) \\ &= -\mathbb{E}_{l_i \sim p(l)} \log(\mathcal{D}(\mathcal{U}_t(l_i^L, l_i^H))).\end{aligned} \tag{6.4}$$

By minimizing the loss $\mathcal{L}_{\mathcal{U}}$, we update the parameters t and thus the discriminative network will be prone to categorize the upsampled faces as real ones. These two discriminative losses in Eqn. 6.3 and Eqn. 6.4 are used to update our upsampling and discriminative networks respectively in an alternating fashion.

All the layers in our MTDN are differentiable and thus RMSprop [Hinton, 2012] is employed to update the parameters t and d. We update the parameters d by minimizing the loss $\mathcal{L}_{\mathcal{D}}$ as follows:

$$\begin{aligned}\Delta^{i+1} &= \gamma\Delta^i + (1-\gamma)(\frac{\partial \mathcal{L}_{\mathcal{D}}}{\partial d})^2, \\ d^{i+1} &= d^i - r\frac{\partial \mathcal{L}_{\mathcal{D}}}{\partial d}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}},\end{aligned} \tag{6.5}$$

where r and γ represent the learning rate and the decay rate respectively, i indicates the index of the iterations, Δ is an auxiliary variable, and ϵ is set to 10^{-8} to avoid division by zero.

Multiple losses, *i.e.*, $\mathcal{L}_{pix}$, $\mathcal{L}_{feat}$, and $\mathcal{L}_{\mathcal{U}}$, are used for learning the parameters of our upsampling network and the object function is expressed as:

$$\mathcal{L}_{\mathcal{T}} = \mathcal{L}_{pix} + \eta\mathcal{L}_{feat} + \lambda\mathcal{L}_{\mathcal{U}}, \tag{6.6}$$

where η and λ are the trade-off weights. We employ lower weights on the feature-wise and discriminative losses because we aim at super-resolving HR faces rather than generating random faces. Thus, λ and η are both set to 0.01. Then, the parameters of our upsampling network t are updated by the gradient descent as follows:

$$\begin{aligned}\Delta^{i+1} &= \gamma\Delta^i + (1-\gamma)(\frac{\partial \mathcal{L}_{\mathcal{T}}}{\partial t})^2, \\ t^{i+1} &= t^i - r\frac{\partial \mathcal{L}_{\mathcal{T}}}{\partial t}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}.\end{aligned} \tag{6.7}$$

As the iteration progresses, the output faces will be more similar to real faces. Therefore, we gradually reduce the impact of the discriminative network by decreasing λ,

$$\lambda^j = \max\{\lambda \cdot 0.995^j, \lambda/2\}, \tag{6.8}$$

Algorithm 2 Minibatch stochastic gradient descent training of MTDN

Input: minibatch size N, LR and HR face image pairs $\{l_i, h_i\}$, maximum number of iterations K.

1: **while** Iter $<$ K **do**
2: Choose one minibatch of LR and HR image pairs $\{l_i, h_i\}, i = 1, \ldots, N$.
3: Decompose LR images into the low-frequency and high-frequency components $\{l_i^L, l_i^H\}$.
4: Generate one minibatch of HR face images $\hat{h}_i$ from $\{l_i^L, l_i^H\}, i = 1, \ldots, N$, where $\hat{h}_i = \mathcal{U}_t(l_i^L, l_i^H)$.
5: Update the parameters of the discriminative network $\mathcal{D}_d$ by using Eqn. 6.3 and Eqn. 6.5.
6: Update the parameters of the multiscale transformative upsampling network $\mathcal{U}_t$ by using Eqn. 6.6 and Eqn. 6.7.
7: Update the trade-off weight λ by using Eqn. 6.8.
8: **end while**

Output: MTDN.

where j is the index of the epochs. Equation 6.8 not only increases the impact of the appearance similarity term but also preserves the class-specific discriminative information in the training phase. The training procedure of our MTDN is illustrated in Algorithm 2.

6.5.4 Hallucinating a Very LR Face Image

The discriminative network is only used for training of the upsampling network. In the testing phase, we first decompose an LR image into a low-frequency component image and its high-frequency residual image and then feed them into the upsampling network to obtain a super-resolved HR face. Because the ground-truth HR face images are upright in the training stage of the entire network, the output of the upsampling network will be an upright face image. As a result, our method does not require alignment of the very low-resolution images in advance. Our network provides an end-to-end mapping from an unaligned LR face image to an upright HR version, which mitigates potential artifacts caused by misalignments and facilitates achieving high-quality super-resolved HR face images.

6.5.5 Implementation Details

In Fig. 6.2, the STN layers are constructed by convolutional and ReLU layers (Conv+ReLU), max-pooling layers with a stride 2 (MP2) and fully connected layers (FC). In particular, STN_1 layer is cascaded by: MP2, Conv+ReLU (with the filter size: 128×20×5×5), MP2, Conv+ReLU (with the filter size: 20×20×5×5), FC+ReLU (from 80 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_2 is cascaded by: MP2, Conv+ReLU (with the filter size: 64×128×5×5), MP2, Conv+ReLU (with the filter size: 128×20×5×5), MP2, Conv+ReLU (with the filter size: 20×20×3×3), FC+ReLU

(from 180 to 20 dimensions) and FC (from 20 to 4 dimensions). We do not use zero-padding in the convolution operations.

In order to merge the low-frequency images with the information extracted from the high-frequency branch, we employ an autoencoder with skip connections. The encoder is composed of convolutional layers with a stride of 2 and zero-paddings. The decoder consists of deconvolutional layers with a stride of 2 and zero-paddings as well. The feature maps from the encoder and decoder are concatenated by skip connections. The residual block is composed of a convolutional layer with a kernel size 3×3, batch normalization, ReLU, a convolutional layer with a kernel size 1×1 and a high-pass connection.

In the following experimental part, some algorithms [Ma et al., 2010; Ledig et al., 2017; Kim et al., 2016b] require the alignments of LR inputs. Thus, we use STN_0 to align the LR inputs images (*i.e.*, 16×16 pixels) for those methods. The only difference between STN_0 and STN_1 is that the first MP2 operation in STN_1 is removed in STN_0 and the input channel is 3.

6.6 Experiments

In this section, we compare our method with the state-of-the-art methods [Ma et al., 2010; Kim et al., 2016a; Ledig et al., 2017; Zhu et al., 2016b; Yu and Porikli, 2017b] qualitatively and quantitatively. Kim et al. [2016a] employ very deep CNN to up-sample images. Ledig et al. [2017] use the generative adversarial framework to enhance super-resolved details. Ma et al. [2010] exploit position-patches in the dataset to reconstruct HR images. Zhu et al. [2016b] develop a deep CNN to localize facial components and then super-resolve them in a cascaded manner. Yu and Porikli [2017b] propose a single-scale face hallucination method, which also employs STN layers and deconvolutional layers for super-resolution.

6.6.1 Dataset

Our network is trained on the Celebrity Face Attributes (CelebA) dataset [Liu et al., 2015]. There are more than 200K face images in this dataset, and the images cover different pose variations and facial expressions. In training our network, we disregard these variations without grouping the face images into different pose and facial expression subcategories.

When generating the LR and HR face pairs, we crop the aligned HR face images from the CelebA dataset, and then resize them to 128×128 pixels as HR images. We manually transform the HR images including 2D translations, rotations and scale changes while constraining the faces in the image region, and then downsample the HR images to generate their corresponding LR images, where the resolutions of LR images are also randomly set between 16 to 32 pixels. We use 70%, 10% and 20% of LR and HR image pairs for training, validation and testing, respectively.

6.6.2 Qualitative Comparisons with the State-of-the-Art

Since our method is able to super-resolve an image with a substantial upscaling factor of 8×, for the methods that do not provide 8× [Kim et al., 2016a; Ledig et al., 2017], we retrain their network on face images with a magnification factor 8×. Furthermore, the resolutions of LR inputs are various, *i.e.*, 16×16∼32×32 pixels, but STNs can only accept an image in a fixed resolution due to the network architecture of its localization module. Considering some methods [Ma et al., 2010; Kim et al., 2016a; Ledig et al., 2017] require alignments before super-resolution and some approaches [Zhu et al., 2016b; Yu and Porikli, 2017b] only accept the input resolution of 16×16 pixels,the input images are resized to 16×16 pixels to meet the requirements. For fair comparisons and better illustration, we transform all the LR input images to the upright view as the inputs of the other methods. In this way, we compare the super-resolution performance with different algorithms for different magnification factors.

As shown in Fig. 6.5(c), traditional upsampling methods, *i.e.*, bicubic interpolation, cannot hallucinate authentic facial details. Since the resolution of inputs is very small, little information is contained in the input images. Simply interpolating input LR images cannot recover extra high-frequency details. As seen in Fig. 6.5(c), Fig. 6.6(c) and Fig. 6.7(c), the images upsampled by bicubic interpolation have some skew effects rather than laying in the upright view. This also indicates that aligning input images by STN_0 suffers from misalignments because it is difficult to estimate transformation parameters accurately from images in such a small size. On the contrary, we apply multiple STNs on the upsampled feature maps, which improves the alignment of the LR inputs. Therefore, our method outputs well-aligned faces. Moreover, with the help of our discriminative network, our method can achieve much sharper results.

Kim et al. [2016a] propose a very deep convolutional neural network based general purpose super-resolution method, dubbed VDSR. Since VDSR is trained on natural image patches, it may be not suitable to super-resolve face images. Furthermore, VDSR does not provide a magnification factor of 8×. Thus, we fine-tune VDSR with face images with an upscaling factor of 8×. However, VDSR is only composed of convolutional layers, and cannot address misalignments of LR faces. Hence, STN_0 is employed to align LR faces before super-resolution. As shown in Fig. 6.5(d), Fig. 6.6(d) and Fig. 6.7(d), VDSR fails to produce realistic facial details. This implies only using a pixel-wise loss as supervision leads to overly smoothed super-resolved results.

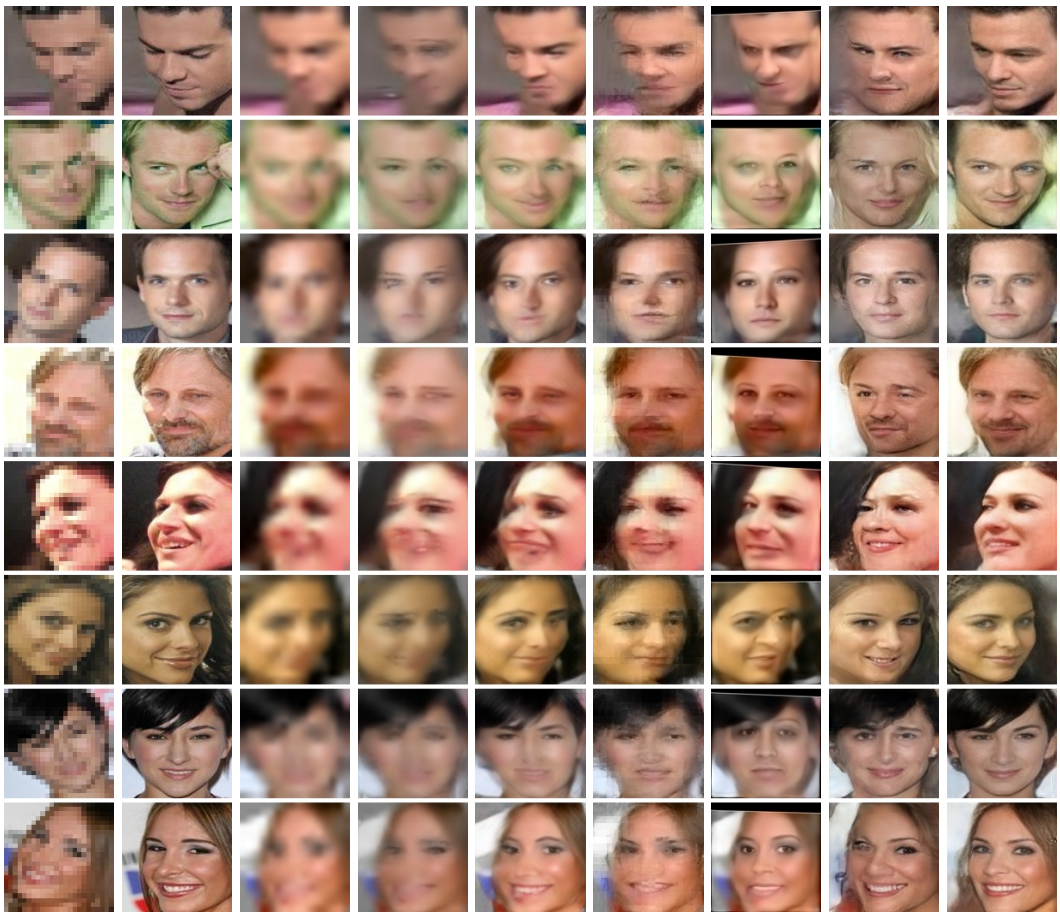
Ledig et al. [2017] develop a generic super-resolution method, known as SRGAN. SRGAN employs the framework of generative adversarial networks [Goodfellow et al., 2014; Radford et al., 2015] to enhance the visual quality. It is trained by using not only a pixel-wise ℓ_2 loss but also an adversarial loss. Similar to VDSR, original SRGAN is also trained on image patches, and thus it is hard to capture the global structure of face images. Therefore, we also retrain SRGAN with entire face images. As seen in Fig. 6.5(e), Fig. 6.6(e) and Fig. 6.7(e), SRGAN captures LR facial patterns and achieves sharper upsampled results compared to VDSR, but misalignments in

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 6.5: Comparisons with the state-of-the-art methods on the input images of size 16×16 pixels. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Kim *et al.*'s method [Kim et al., 2016a] (VDSR). (e) Ledig *et al.*'s method [Ledig et al., 2017] (SRGAN). (f) Ma *et al.*'s method [Ma et al., 2010]. (g) Zhu *et al.*'s method [Zhu et al., 2016b] (CBN). (h) Yu and Porikli's method [Yu and Porikli, 2017b] (TDAE). (i) Our method.

LR faces cause severe distortions and artifacts in the final hallucinated faces.

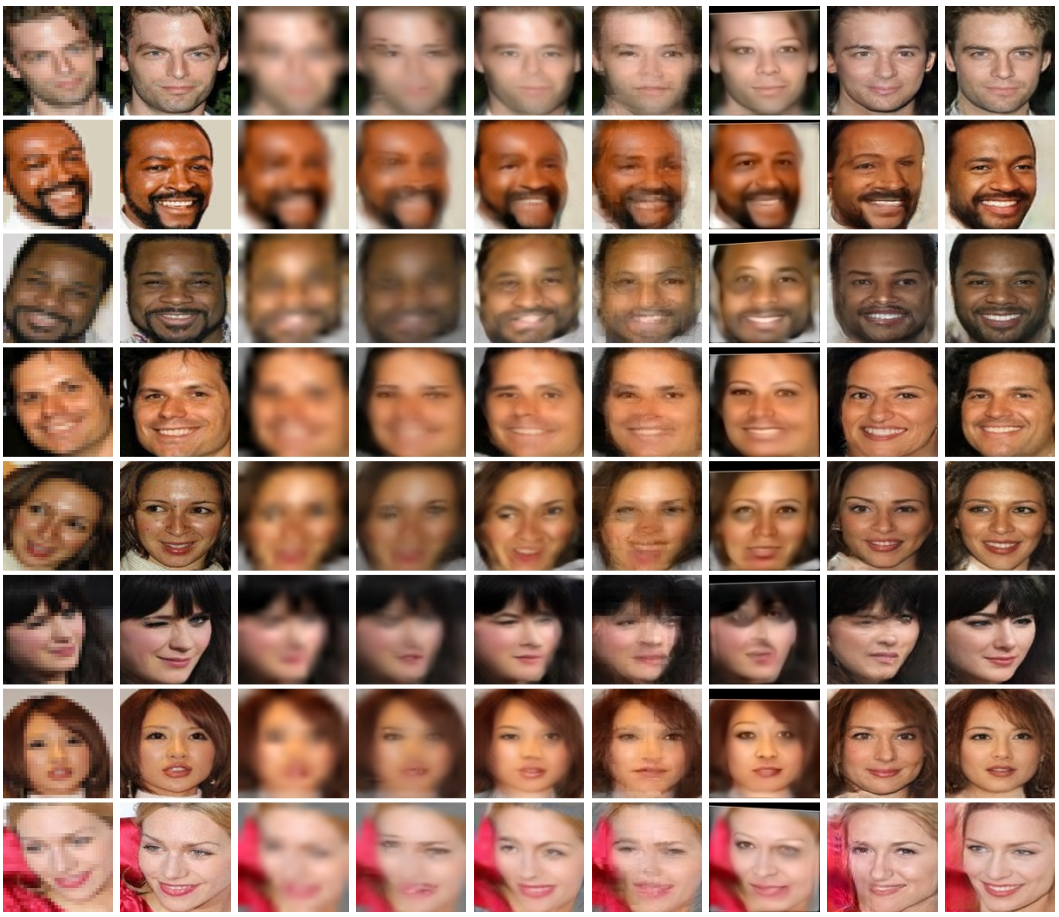
Ma et al. [2010] exploit position patches to hallucinate HR faces. Thus their method requires the LR inputs to be precisely aligned with the reference images in the training dataset. As seen in Fig. 6.5(f), Fig. 6.6(f) and Fig. 6.7(f), as the upscaling factor increases, the correspondences between LR and HR patches become more inconsistent. As a result, this method suffers from obvious blocky artifacts around the boundaries of different patches. In addition, when there are obvious alignment errors in the aligned LR faces or large poses exist, their method will output mixed and blurry facial components in their results.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 6.6: Comparisons with the state-of-the-art methods on the input images of size 24×24 pixels. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Kim *et al.*'s method [Kim et al., 2016a] (VDSR). (e) Ledig *et al.*'s method [Ledig et al., 2017] (SRGAN). (f) Ma *et al.*'s method [Ma et al., 2010]. (g) Zhu *et al.*'s method [Zhu et al., 2016b] (CBN). (h) Yu and Porikli's method [Yu and Porikli, 2017b] (TDAE). (i) Our method.

Zhu et al. [2016b] present a deep cascaded bi-branch network for face hallucination, named CBN, where one branch first localizes facial components, then aligns and upsamples LR facial components while the other branch is used to upsample global face profiles. However, when the inputs undergo large pose variations, CBN cannot localize facial components accurately, and thus produces severe artifacts as seen in Fig. 6.5(g), Fig. 6.6(g) and Fig. 6.7(g). In contrast, our method estimates the 2D deformations of LR faces and aligns them by multiple STNs in the procedure of super-resolution, where misalignments from the previous STN layer can be eliminated by the latter STN layer. Therefore, our results do not suffer the ghosting artifacts



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 6.7: Comparisons with the state-of-the-art methods on the input images of size 32×32 pixels. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Kim *et al.*'s method [Kim et al., 2016a] (VDSR). (e) Ledig *et al.*'s method [Ledig et al., 2017] (SRGAN). (f) Ma *et al.*'s method [Ma et al., 2010]. (g) Zhu *et al.*'s method [Zhu et al., 2016b] (CBN). (h) Yu and Porikli's method [Yu and Porikli, 2017b] (TDAE). (i) Our method.

as shown in Fig. 6.5(i), Fig. 6.6(i) and Fig. 6.7(i). Note that the facial component localization branch in CBN requires the input resolution to be fixed, *i.e.*16×16 pixels. Therefore, if the resolutions of input images are larger than 16×16 pixels, CBN needs to downsample input images first. In that case, CBN may lose high-frequency information of inputs and achieves suboptimal hallucination results.

Yu and Porikli [2017b] design a transformative discriminative autoencoder, called TDAE, to upsample noisy and unaligned LR face images. However, TDAE only takes LR images in a fixed resolution, and it has to downsample LR images to a lower-resolution when the resolutions of input images are larger than the required

Table 6.1: Quantitative comparisons on the entire test dataset

Methods	Bicubic	VDSR	SRGAN	Ma *et al.*	CBN	TDAE	IBSR	Ours
PSNR	19.23	20.13	19.08	19.11	18.78	20.83	21.45	**21.98**
SSIM	0.56	0.57	0.57	0.54	0.54	0.57	0.59	**0.62**

resolution, *i.e.*, 16×16 pixels. Therefore, TDAE will lose details of input images and may generate inaccurate facial characteristics, such as gender reversal as visible in the second row of Fig. 6.6(h) and the third row of Fig. 6.7(h). Furthermore, benefiting from the feature-wise loss, our MTDN is able to hallucinate facial characteristics akin to the ground-truth HR faces. Furthermore, TDAE is trained mainly on near-frontal face images. It does not super-resolves LR faces in large poses well. In contrast, we enlarge the training dataset with more examples and more challenging poses to train our MTDN. Therefore, our network attains better super-resolution performance.

As shown in Fig. 6.5(g), Fig. 6.6(g) and Fig. 6.7(g), our method reconstructs authentic facial details and the reconstructed faces have different poses and facial expressions. Since our method applies multiple STNs on feature maps to align face images, we can achieve better alignment results without damaging input LR facial patterns. Furthermore, our method does not warp input images directly, so there are no blank regions in our results. Since our network is able to receive LR images at different resolutions without discarding residual images, our method can exploit information better than the other methods. Notice that, we only use a single network to super-resolve all the LR face images in various resolutions.

6.6.3 Quantitative Results

We report the quantitative comparison results using the average Peak Single-to-Noise Ratio (PSNR) and Structural SIMilarity scores (SSIM) on the entire test dataset in Tab. 6.1. Note that, in the test dataset the resolutions of LR input face images ranges from 16×16 to 32×32. We use all the methods to upsample LR face images to the HR images of size 128×128 pixels and then compare the upsampled HR faces with their corresponding ground-truths. As mentioned in Sec. 6.6.2, all the other methods need to downsample input images to 16×16 pixels.

As indicated in Tab. 6.1, our MTDN attains the best PSNR and SSIM results and outperforms the second best with a large margin of 1.15 dB in PSNR. Note that, our previous work TDAE [Yu and Porikli, 2017b] also interweaves STN layers as well as deconvolutional layers to upsample unaligned face images, but it only accepts input images in a fixed resolution, *i.e.*16×16 pixels, and thus achieves the second best performance. This indicates that TDAE loses important high-frequency information of LR images in the downsampling operation. As indicated in Tab. 6.1, by using the multi-scale strategy and the two branch architecture network, we can preserve all the information of the LR inputs in super-resolution and thus obtain superior

Table 6.2: Quantitative evaluations on different STN layers

STNs	STN_1	STN_2	Ours
PSNR	21.47	21.74	**21.98**
SSIM	0.62	0.62	**0.62**

Table 6.3: Quantitative evaluations on different losses

Losses	$\mathcal{L}_{pix}$	$\mathcal{L}_{pix+feat}$	$\mathcal{L}_{pix+\mathcal{U}}$	$\mathcal{L}_{\mathcal{T}}$
PSNR	22.34	22.09	21.71	**21.98**
SSIM	0.65	0.65	0.61	**0.62**

performance.

6.7 Discussions

6.7.1 Impacts of Residual Branch

As indicated by the quantitative result of TDAE in Tab. 6.1, the downsampling operation leads to suboptimal super-resolution performance. Since our MTDN also employs extra residual blocks, the improvement of the performance may be caused by the increased capacity of the network. In order to evaluate the impacts of the high-frequency residual branch, similar to our previous methods [Yu and Porikli, 2016, 2017b], we only employ one branch, *i.e.*, the low-frequency branch, to upsample LR input face images. Note that, we do not need to re-train our MTDN network. As shown in Tab. 6.5, the performance of only using low-frequency branch is marked by noHF, and its performance degrades 0.71 dB in PSNR. It indicates that the high-frequency residual information extracted from the input images contains useful clues for super-resolution. Thus, providing more high-frequency details improves face super-resolution performance.

6.7.2 Effects of Different Losses

As mentioned in Sec. 6.5.3, there are three different losses employed to train our network, *i.e.*, pixel-wise and feature-wise ℓ_2 losses and a class-wise discriminative loss. Pixel-wise ℓ_2 loss is used to constrain the appearance similarity. As reported in our previous work [Yu and Porikli, 2016] and as indicated in Tab. 6.3, the upsampling network which is trained only by a pixel-wise ℓ_2 loss to super-resolve LR faces obtains the highest PSNR but produces over-smoothed results as shown in Fig. 6.4(f).

The feature-wise loss is able to make the super-resolved results sharper without suffering over-smoothness because it forces the high-order moments of upsampled faces, *i.e.*, feature maps of faces, to be similar to their ground-truths. In addition,

Table 6.4: Quantitative evaluations on different input resolutions

Resolutions	16×16	24×24	32×32
PSNR	20.97	22.16	22.23
SSIM	0.59	0.63	0.63

Table 6.5: Quantitative evaluations on different components in our MTDN

Modules	NoAE	NoSkip	NoHF	Ours
PSNR	21.16	21.57	21.27	**21.98**
SSIM	0.62	0.61	0.60	**0.62**

we also incorporate a class-wise discriminative loss to force the upsampling network to generate realistic faces. Since the class-specific loss is not used to measure the similarity between two images, too large discriminative loss will distort our super-resolution performance. Therefore, there is a trade-off between the upsampling and discriminative networks and we gradually decrease the influence of the discriminative network as iterations progress.

Because PSNR is designed to measure the similarity of appearance intensities but does not reflect visual quality of reconstructed images, using the feature-wise and class-wise losses decreases the PSNR, as seen in Tab. 6.3 but improves the visual quality significantly, as visible in Fig. 6.4.

6.7.3 Impacts of Multiple STN Layers

As illustrated in Fig. 6.2, we apply two STN layers to align feature maps in our network. Our previous works [Yu and Porikli, 2017a,b] only use one branch to upsample LR faces and they align feature maps at the resolution of 16×16 pixels. However, our MTDN has two branches and the resolutions of these two branch inputs are different. Therefore, we apply STN layers after the concatenation layer, where the resolution of the feature maps is 32×32 pixels. In this manner, all feature maps can be aligned simultaneously. As mentioned in [Jaderberg et al., 2015], using multiple STNs can achieve more accurate alignment. Due to the GPU memory limitation, we cannot apply an STN layer to align the feature maps of size 128×128 pixels. Hence, we only employ two STN layers to the feature maps of size 32×32 and 64×64 pixels in our network. As shown in Tab. 6.2, we demonstrate the contributions of different STN layers to the final performance. Table 6.2 also indicates that using multiple STN layers can improve face alignment, thus obtaining better face hallucination performance.

(a) (b) (c) (d) (e)

Figure 6.8: Comparisons of different variants of our network. (a) The input 16×16 LR images. (b) The original 128×128 HR images. (c) Results of the network without using the autoencoder. (d) Results of IBSR. (e) Our results.

Figure 6.9: Real-world cases. The top row: real-world LR faces captured in the wild. The bottom row: our super-resolved results.

6.7.4 Effects of Autoencoder in Low-frequency Branch

Different from our previous works [Yu and Porikli, 2016, 2017a, 2018], our MTDN does not super-resolve LR faces directly by deconvolutional layers. Since our method needs to fuse two branch images together, we first extract feature maps from the two branch input images separately. In order to make the resolutions of the feature maps from the two branches compatible, we upsample the feature maps of the low-frequency branch to 32×32 pixels. In particular, we apply an autoencoder with skip connections to extract features and then upsample features by a deconvolutional layer in the low-frequency branch while residual blocks are applied to extract features from the high-frequency branch. Since the resolution of the low-frequency branch is very small, the autoencoder does not require much GPU memory but increases the capacity of our network.

We replace the autoencoder with a convolutional layer and use a deconvolutional layer to upsample the LR feature maps in the low-frequency branch, and we represent this variant as noAE in Tab. 6.5. As demonstrated in Tab. 6.5, the performance of noAE degrades 0.82 dB compared to our MTDN. Therefore, by increasing the network capacity, *i.e.*, the employment of the autoencoder, our MTDN achieves better quantitative super-resolution performance. Furthermore, the upsampled faces also achieve better visual quality by using the autoencoder, as shown in Fig. 6.8. It al-

so demonstrates that our autoencoder can extract feature maps better than a single convolutional layer. Since skip connections are employed in the autoencoder, we can also preserve the spatial information from the encoder to the decoder. Removal of the skip connections in our network, marked as NoSkip, causes 0.41 dB degradation in PSNR, as shown in Tab. 6.5. However, we do not observe significant deterioration in visual quality.

6.7.5 PSNR and SSIM at Different Input Resolutions

Since our test dataset consists of LR face images at different resolutions, it cannot reflect the performance of our network as the input resolutions increase. Hence, we generate another test dataset where each HR face image corresponds to three different LR image versions, *i.e.*, 16×16, 24×24 and 32×32 pixels. We group and super-resolve input LR images according to their resolutions and then measure the performance of our network in each group. As indicated in Tab. 6.4, our network generates better super-resolved results in terms of PSNR as the input resolution increases. It implies our proposed two branch network can fully exploit input information when more information is provided in LR input images.

6.7.6 Interpolation before Super-resolution

There is another option for preserving all the information in the LR input images: we can first resize the different LR image sizes to 32×32 pixels by bicubic interpolation and then super-resolve the interpolated images. We name this super-resolution approach as IBSR. As reported in previous generic super-resolution methods [Kim et al., 2016a; Ledig et al., 2017], using convolutional and deconvolutional layers can achieve better super-resolution performance than traditional interpolation methods, *e.g.*bicubic interpolation. Therefore, we use an autoencoder and a deconvolutional layer to upsample low-frequency part as well as residual blocks to extract features from high-frequency residuals. After obtaining the feature maps from the low-frequency and high-frequency branches, we fuse those feature maps by a residual block. In this fashion, we achieve 128 channel features maps of size 32×32 for further super-resolution rather than only 3 channel interpolated images in IBSR. Hence, our network architecture can achieve better performance qualitatively and quantitatively, as demonstrated in Fig. 6.8(d) and Tab. 6.1.

6.7.7 Real World Cases

Since it is easy to obtain real-world LR face images but very difficult to attain their corresponding HR images, we use bicubic downsampling to mimic the degradation process. Although our network is trained on CelebA dataset, our model can also super-resolve real-world LR face images effectively, as seen in Fig. 6.9. In Fig. 6.9, we randomly choose LR face images from 16×16 pixels to 32×32 pixels in WiderFace dataset [Yang et al., 2016] where LR faces are captured in the wild. As visible in real-world LR faces, the mosaic artifacts and noise are obvious, which can degrade

the super-resolution performance. We believe with proper data augmentation our network is able to super-resolve real-world LR faces even better.

6.8 Conclusion

We present a novel and capable multiscale transformative discriminative network to super-resolve very small LR face images. By designing a two branch input neural network, our network can upsample LR images in various resolutions without discarding the residuals of resized input images. In this manner, our method is able to utilize all the information from inputs for face super-resolution. Furthermore, our algorithm can increase the input LR image size significantly, *e.g.* $8\times$, and reconstruct much richer facial details. Since our method does not require any alignments of LR faces and learns an end-to-end mapping between LR and HR face images, it preserves well the global structure of faces and is more practical.

Chapter 7

Face Super-resolution Guided by Facial Component Heatmaps

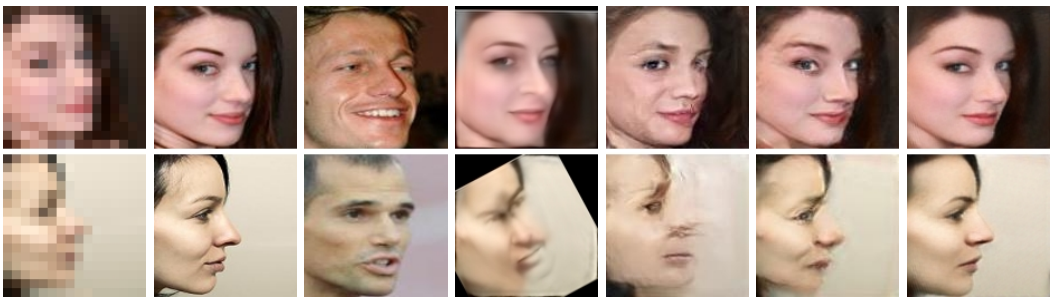
7.1 Foreword

In previous chapters, our networks are designed to super-resolve low-resolution face images undergoing different 2D transformations, such as rotational and translational misalignments. Even though our networks can hallucinate faces in different poses, those low-resolution input faces are nearly frontal. When the input faces undergo large pose variations, such as side-view poses, our previous methods may fail to upsample those faces authentically. There may be two possible reasons: one is that we do not use sufficient data to train our networks, and the other is that our proposed networks cannot recognize those facial components inherently. In this chapter, we first demonstrate that our previously presented networks cannot recognize facial components when input low-resolution faces exhibit large poses. Thus, they fail to upsample faces in large poses. Then, we propose a multi-task upsampling network to super-resolve low-resolution face images while localizing facial components on the fly. Since our proposed network can localize facial components, it can super-resolve facial components explicitly. Therefore, our network is able to super-resolve faces in a wide range of poses.

This chapter has been published as a conference paper: Xin Yu, Basura Fernando, Bernard Ghanem, Fatih Porikli, Richard Hartley: Face Super-Resolution Guided by Facial Component Heatmaps. In *European Conference on Computer Vision (ECCV)*, 217-233, 2018.

7.2 Abstract

State-of-the-art face super-resolution methods leverage deep convolutional neural networks to learn a mapping between low-resolution (LR) facial patterns and their corresponding high-resolution (HR) counterparts by exploring local appearance information. However, most of these methods do not account for facial structure and suffer from degradations due to large pose variations and misalignments. In this paper, we propose a method that explicitly incorporates structural information



(a) (b) (c) (d) (e) (f) (g)

Figure 7.1: Comparison of state-of-the-art face super-resolution methods on very low-resolution (LR) face images. Columns: (a) Unaligned LR inputs. (b) Original HR images. (c) Nearest Neighbors (NN) of aligned LR faces. Note that image intensities are used to find NN. (d) CBN [Zhu et al., 2016b]. (e) TDAE [Yu and Porikli, 2017b]. (f) TDAE†. We retrain the original TDAE with our training dataset. (g) Our results.

of faces into the face super-resolution process by using a multi-task convolutional neural network (CNN). Our CNN has two branches: one for super-resolving face images and the other branch for predicting salient regions of a face coined *facial component heatmaps*. These heatmaps encourage the upsampling stream to generate super-resolved faces with higher-quality details. Our method not only uses low-level information (*i.e.*, intensity similarity), but also middle-level information (*i.e.*, face structure) to further explore spatial constraints of facial components from LR inputs images. Therefore, we are able to super-resolve very small unaligned face images (16×16 pixels) with a large upscaling factor of 8×, while preserving face structure. Extensive experiments demonstrate that our network achieves superior face hallucination results and outperforms the state-of-the-art.

7.3 Introduction

Face images provide crucial clues for human observation as well as computer analysis [Fasel and Luettin, 2003; Zhao et al., 2003]. However, the performance of most existing facial analysis techniques, such as face alignment [Xiong and De la Torre, 2013; Bulat and Tzimiropoulos, 2017a] and identification [Taigman et al., 2014], degrades dramatically when the resolution of a face is adversely low.

Face super-resolution (FSR) [Baker and Kanade, 2000], also known as face hallucination, provides a viable way to recover a high-resolution (HR) face image from its low-resolution (LR) counterpart and has attracted increasing interest in recent years. Modern face hallucination methods [Zhou and Fan, 2015; Yu and Porikli, 2016, 2017b; Zhu et al., 2016b; Cao et al., 2017; Dahl et al., 2017] employ deep learning and achieve state-of-the-art performance. These methods explore image intensity correspondences between LR and HR faces from large-scale face datasets. Since

near-frontal faces prevail in popular large-scale face datasets [Liu et al., 2015; Huang et al., 2007], deep learning based FSR methods may fail to super-resolve LR faces under large pose variations, as seen in the examples of Fig. 7.1. In fact, in these examples, the face structure has been distorted and facial details are not fully recovered by state-of-the-art super-resolution methods.

A naive idea to remedy this issue is to augment training data with large pose variations (*i.e.*, [Zafeiriou et al., 2017]) and then retrain the neural networks. As shown in Fig. 7.1(f), this strategy still leads to suboptimal results where facial details are missing or distorted due to erroneous localization of LR facial patterns. This limitation is common in intensity-based FSR methods that only exploit local intensity information in super-resolution and do not take face structure or poses into account. We postulate that methods that explicitly exploit information about the locations of facial components in LR faces have the capacity to improve super-resolution performance.

Another approach to super-resolve LR face images is to localize facial components in advance and then upsample them [Yang et al., 2013; Zhu et al., 2016b] progressively. However, localizing these facial components with high accuracy is generally a difficult task in very LR images, especially under large pose variations. As shown in Fig. 7.1(e), the method of Zhu et al. [2016b] fails to localize facial components accurately and produces an upsampled face with severe distortions. Therefore, directly detecting facial components or landmarks in LR faces is suboptimal and may lead to ghosting artifacts in the final result.

In contrast to previous methods, we propose a method that super-resolves LR face images while predicting face structure in a collaborative manner. Our intuition is that, although it is difficult to accurately detect facial landmarks in LR face images, it is possible to localize facial components (not landmarks) and identify the visibility of the components on the super-resolved faces or the intermediate upsampled feature maps because they can provide enough resolution for localization. Obtaining the locations of facial components can in turn facilitate face super-resolution.

Driven by this idea, we propose a multi-task deep neural network to upsample LR images. In contrast to the state-of-the-art FSR methods [Yu and Porikli, 2017b; Zhu et al., 2016b; Cao et al., 2017; Dahl et al., 2017], our network not only super-resolves LR images but also estimates the spatial positions of their facial components. Then the estimated locations of the facial components are regarded as a guidance map which provides the face structure in super-resolution. Here, face structure refers to the locations and visibility of facial components as well as the relationship between them and we use heatmaps to represent the probability of the appearance of each component. Since the resolution of the input faces is small, (*i.e.*, 16×16 pixels), localizing facial components is also very challenging. Instead of detecting facial components in LR images, we opt to localize facial components on super-resolved feature maps. Specifically, we first super-resolve features of input LR images, and then employ a spatial transformer network [Jaderberg et al., 2015] to align the feature maps. The upsampled feature maps are used to estimate the heatmaps of facial components. Since the feature maps are aligned, the same facial components may appear at the corresponding positions closely. This also provides an initial estimation

for the component localization. Furthermore, we can also largely reduce the training examples for localizing facial components when input faces or feature maps are pre-aligned. For instance, we only use 30K LR/HR face image pairs for training our network, while a state-of-the-art face alignment method [Bulat and Tzimiropoulos, 2017a] requires about 230K images to train a landmark localization network.

After obtaining the estimated heatmaps of facial components, we concatenate them with the upsampled feature maps to infuse the spatial and visibility information of facial components into the super-resolution procedure. In this fashion, higher-level information beyond pixel-wise intensity similarity is explored and used as an additional prior in FSR. As shown in Fig. 7.1(g), our presented network is able to upsample LR faces in large poses while preserving the spatial structure of upsampled face images.

Overall, the contributions of our work can be summarized as:

- We present a novel multi-task framework to super-resolve LR face images of size 16×16 pixels by an upscaling factor of $8\times$, which not only exploits image intensity similarity but also explores the face structure prior in face super-resolution.
- We not only upsample LR faces but also estimate the face structure in the framework. Our estimated facial component heatmaps provide not only spatial information of facial components but also their visibility information, which cannot be deduced from pixel-level information.
- We demonstrate that the proposed two branches, *i.e.*, upsampling and facial component estimation branches, collaborate with each other in super-resolution, thus achieving better face hallucination performance.
- Due to the design of our network architecture, we are able to estimate facial component heatmaps from the upsampled feature maps, which provides enough resolutions and details for estimation. Furthermore, since the feature maps are aligned before heatmap estimation, we can largely reduce the number of training images to train the heatmap estimation branch.

To the best of our knowledge, our method is the first attempt to use a multi-task framework to super-resolve very LR face images. We not only focus on learning the intensity similarity mappings between LR and HR facial patterns, similar to [Yu and Porikli, 2017b; Dahl et al., 2017; Ma et al., 2010], but also explore the face structure information from images themselves and employ it as an additional prior for super-resolution.

7.4 Related Work

Exploiting facial priors, such as spatial configuration of facial components, in face hallucination is the key factor different from generic super-resolution tasks. Based

on the usage of the priors, face hallucination methods can be roughly grouped into global model based and part based approaches.

Global model based approaches aim at super-resolving an LR input image by learning a holistic appearance mapping such as PCA. Wang and Tang [2005] learn subspaces from LR and HR face images respectively, and then reconstruct an HR output from the PCA coefficients of the LR input. Liu et al. [2007] employ a global model for the super-resolution of LR face images but also develop a local nonparametric model, *i.e.*, markov random field (MRF), to augment the facial details and reduce ghosting artifacts caused by the misalignments in LR images. Kolouri and Rohde [2015] employ optimal transport techniques to morph an HR output by interpolating exemplar HR faces whose downsampled versions are close to the LR input in terms of distances in the LR face subspace. In order to learn a good global model, LR inputs are required to be precisely aligned and to share similar poses to the exemplar HR images. When large pose variations and misalignments exit in LR inputs, these methods are prone to produce severe artifacts.

Considering pose and expression variations in both LR and HR face images, it is difficult to hallucinate HR faces by employing only one global appearance model. Thus, part based methods are proposed to super-resolve individual facial regions separately. They reconstruct the HR counterparts of LR inputs based on either reference patches or facial components in the training dataset. Baker and Kanade [2002] search the best mapping between LR and HR patches and then use the matched HR patches to recover high-frequency details of aligned LR face images. Motivated by this idea, some works [Ma et al., 2010; Yang et al., 2010; Li et al., 2014] average weighted position patches extracted from multiple aligned HR images to upsample aligned LR face images in either the image intensity domain or sparse coding domain, while Jin and Bouganis [2015] exploit a patch-wise mixture of probabilistic PCA priors to reconstruct HR faces. However, patch based methods also require LR inputs to be aligned in advance and may produce blocky artifacts when the upscaling factor is too large. Instead of using position patches, Tappen and Liu [2012] super-resolve HR facial components by warping the reference HR images and the warping transformation is estimated by SIFT flow [Liu et al., 2011] between the LR input and LR training exemplars. Yang et al. [2013] localize facial components in the LR images by a facial landmark detector and then reconstruct missing high-frequency details from similar HR reference components. Because facial component based methods need to extract facial parts in LR images and then align them to exemplar images accurately, their performance degrades dramatically when the resolutions of input faces become unfavorably small.

Recently, deep learning techniques have been applied to the face hallucination field and achieved significant progress. Yu and Porikli [2016] present a discriminative generative network to hallucinate aligned LR face images. Their follow-up works [Yu and Porikli, 2017a,b] interweave multiple spatial transformer networks [Jaderberg et al., 2015] with the deconvolutional layers to handle unaligned LR faces. Zhou and Fan [2015] extract features from a blurry LR face image by a convolutional neural network (CNN) and then reconstruct a sharp HR face image from them. Xu et al.

[2017] employ the framework of generative adversarial networks [Goodfellow et al., 2014; Radford et al., 2015] to recover blurry LR face images while enhancing the facial details by a multi-class discriminative loss. Dahl et al. [2017] leverage the framework of PixelCNN [Van Den Oord et al., 2016] to super-resolve very low-resolution faces. Since the above deep convolutional networks only consider local information in super-resolution without taking the holistic face structure into account, they may distort face structure when super-resolving non-frontal LR faces. Zhu et al. [2016b] employ a cascade bi-network, dubbed CBN, to upsample very low-resolution and unaligned faces, where the low-frequency parts are upsampled by a convolutional network and the high-frequency parts, *i.e.*, facial components, are firstly localized by a pre-defined model and then upsampled by the another network. Since CBN needs to localize facial components in LR images, CBN may produce ghosting faces when there are localization errors. Concurrent to our work, the algorithms [Bulat and Tzimiropoulos, 2018; Chen et al., 2018] also employ facial structure in face hallucination. In contrast to their works, we propose a multi-task network which can be trained in an end-to-end manner. In particular, our network not only estimates the facial heatmaps but also employs them for achieving high-quality super-resolved results.

7.5 Our Proposed Method

Our network mainly consists of two parts: a multi-task upsampling network and a discriminative network. Our multi-task upsampling network (MTUN) is composed of two branches: an upsampling branch and a facial component heatmap estimation branch. Our upsampling branch consists of an autoencoder, deconvolutional layers and one spatial transformer layer [Jaderberg et al., 2015]. Different from [Yu and Porikli, 2017b], we employ an autoencoder to reduce image noise while extracting high-frequency details from LR inputs before upsampling without requiring much memory. In order to explore the face structure information from LR inputs, we propose a facial component heatmap estimation branch (HEB). The predicted heatmaps will be fed into the upsampling branch as additional mid-level structure information for super-resolution. Benefiting from our HEB, we not only impose face structure in super-resolution but also estimate the structure on the fly rather than localizing facial components in very LR input images beforehand. Thus, HEB is the key part of our algorithm which is distinct from previous works [Zhu et al., 2016b; Yu and Porikli, 2016, 2017b]. The discriminative network enforces the generated HR faces to lie on the manifold of real HR face images. Figure 7.2 illustrates the overall architecture of our proposed network. The entire network is trained in an end-to-end fashion.

7.5.1 Facial Component Heatmap Estimation

When the resolution of input images is too small, facial components will be even smaller and thus it is very difficult for state-of-the-art facial landmark detectors to localize facial landmarks in very low-resolution images accurately. However, we

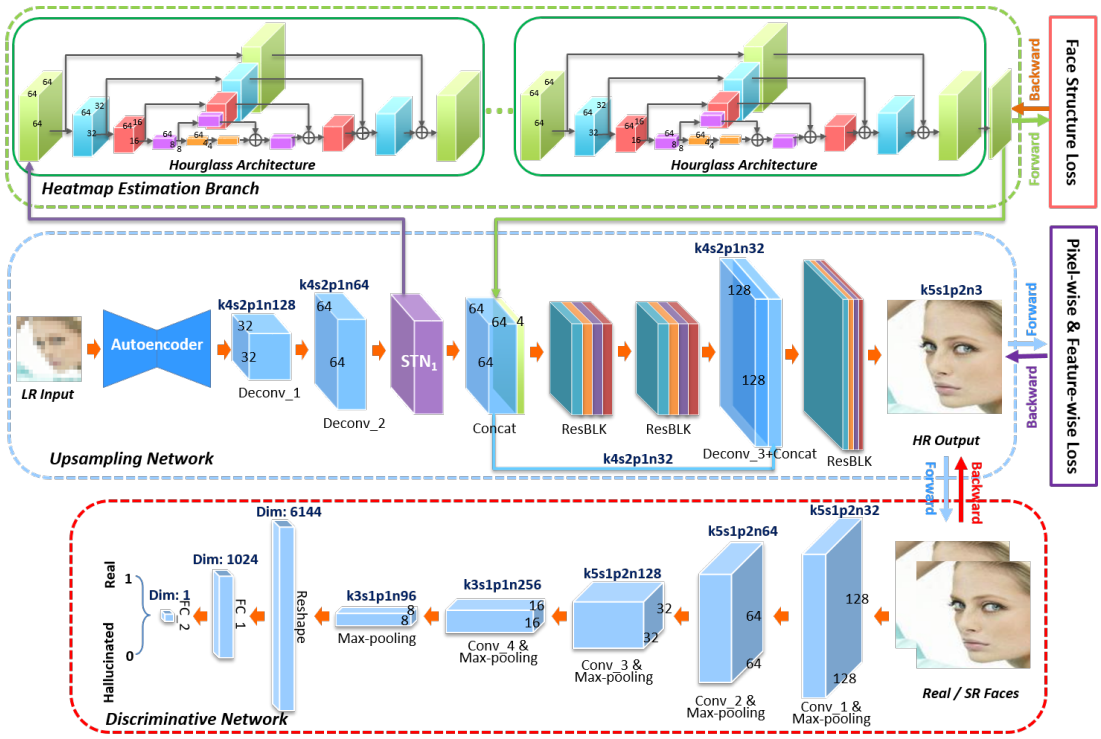
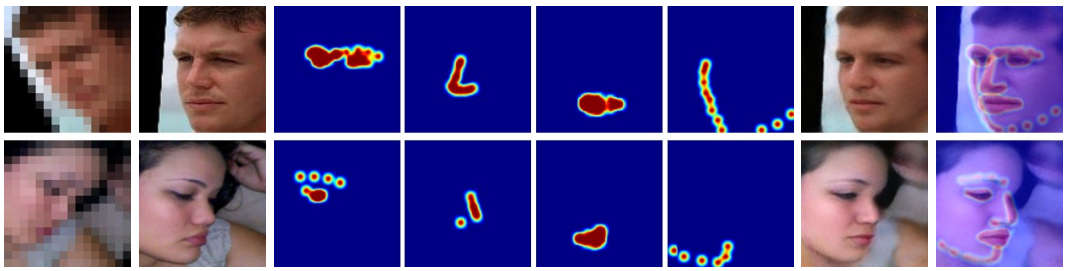


Figure 7.2: The pipeline of our multi-task upsampling network. In the testing phase, the upsampling branch (blue block) and the heatmap estimation branch (green block) are used.

propose to predict facial component heatmaps from super-resolved feature maps rather than localizing landmarks in LR input images, because the upsampled feature maps contain more details and their resolutions are large enough for estimating facial component heatmaps. Moreover, since 2D faces may exhibit a wide range of poses, such as in-plane rotations, out-of-plane rotations and scale changes, we may need a large number of images for training HEB. For example, Bulat and Tzimiropoulos [2017a] require over $200K$ training images to train a landmark detector, and there is still a gap between the accuracy of [Bulat and Tzimiropoulos, 2017a] and human labeling. To mitigate this problem, our intuition is that when the faces are roughly aligned, the same facial components lie in the corresponding positions closely. Thus, we employ a spatial transformer network (STN) to align the upsampled features before estimating heatmaps. In this way, we not only ease the heatmap estimation but also significantly reduce the number of training images used for learning HEB.

We use heatmaps instead of landmarks based on three reasons: (i) localizing each facial landmark individually is difficult in LR faces even for humans and erroneous landmarks would lead to distortions in the final results. On the contrary, it is much easier to localize each facial components as a whole. (ii) Even state-of-the-art landmark detectors may fail to output accurate positions in high-resolution images, such as in large pose cases. However, it is not difficult to estimate a region represented



(a) (b) (c) (d) (e)

Figure 7.3: Visualization of estimated facial component heatmaps. Columns: (a) Unaligned LR inputs. (b) HR images. (c) Ground-truth heatmaps generated from the landmarks of HR face images. (d) Our results. (e) The estimated heatmaps overlying over our super-resolved results. Note that, we overlap four estimated heatmaps together and upsample the heatmaps to fit our upsampled results.

by a heatmap in those cases. (iii) Furthermore, our goal is to provide clues of the spatial positions and visibility of each component rather than the exact shape of each component. Using heatmaps as a probability map is more suitable for our purpose.

In this paper, we use four heatmaps to represent four components of a face, *i.e.*, eyes, nose, mouth and chain, respectively. We exploit 68 point facial landmarks to generate the ground-truth heatmaps. Specifically, each landmark is represented by a Gaussian kernel and the center of the kernel is the location of the landmark. By adjusting the standard variance of Gaussian kernels in accordance with the resolutions of feature maps or images, we can generate a heatmap for each component. The generated ground-truth heatmaps are shown in Fig. 7.3(c). Note that, when self-occlusions appear, some components are not visible and they will not appear in the heatmaps. In this way, heatmaps not only provides the locations of components but also their visibility in the original LR input images.

In order to estimate facial component heatmaps, we employ the stacked hourglass network architecture [Newell et al., 2016]. It exploits a repeated bottom-up and top-down fashion to process features across multiple scales and is able to capture various spatial relationships among different parts. As suggested by Newell et al. [2016], we also use the intermediate supervision to improve the performance. The green block in Fig. 7.2 illustrates our facial component heatmap estimation branch. We feed the aligned feature maps to HEB and then concatenate the estimated heatmaps with the upsampled feature maps for super-resolving facial details. In order to illustrate the effectiveness of HEB, we resize and then overlay the estimated heatmaps over the output images as visible in Fig. 7.3(e). The ground-truth heatmaps are shown in Fig. 7.3(c) for comparison.

7.5.2 Network Architecture

7.5.2.1 Multi-task Upsampling Network

Figure 7.2 illustrates the architecture of our proposed multi-task upsampling network (MTUN) in the blue and green blocks. MTUN consists of two branches: an upsampling branch (blue block) and a facial component heatmap estimation branch (green block). The upsampling branch firstly super-resolves features of LR input images and then aligns the feature maps. When the resolution of the feature maps is large enough, the upsampled feature maps are fed into HEB to estimate the locations and visibility of facial components. Thus we obtain the heatmaps of the facial components of LR inputs. The estimated heatmaps are then concatenated with the upsampled feature maps to provide the spatial positions and visibility information of facial components for super-resolution.

In the upsampling branch, the network is composed of a convolutional autoencoder, deconvolutional layers and an STN. The convolutional autoencoder is designed to extract high-frequency details from input images while removing image noise before upsampling and alignment, thus increasing the super-resolution performance. The deconvolutional layers are employed to super-resolve the feature maps. Since input LR faces undergo in-plane rotations, translations and scale changes, STN is employed to compensate for those affine transformations, thus facilitating facial component heatmap estimation.

After obtaining aligned upsampled feature maps, those feature maps are used to estimate facial component heatmaps by an HEB. We construct our HEB by a stacked hourglass architecture [Newell et al., 2016], which consists of residual blocks and upsampling layers, as shown in the green block of Fig. 7.2.

Our multi-task network aims at super-resolving input face images as well as predicting heatmaps of facial components in the images. As seen in Fig. 7.4(c), when we only use the upsampling branch to super-resolve faces without using HEB, the facial details are blurred and some facial components, *e.g.*, mouth and nose, are distorted in large poses. Furthermore, the heatmap supervision also forces STN to align the upsampled features more accurately, thus improving super-resolution performance. Therefore, these two tasks collaborate with each other and benefit from each other as well. As shown in Fig. 7.4(f), our multi-task network achieves better super-resolved results.

7.5.2.2 Discriminative Network

Recent works [Yu and Porikli, 2016, 2017b; Xu et al., 2017; Ledig et al., 2017] demonstrate that only using Euclidean distance (ℓ_2 loss) between the upsampled faces and the ground-truth HR faces tends to output over-smoothed results. Therefore, we incorporate a discriminative objective into our network to force super-resolved HR face images to lie on the manifold of real face images.

As shown in the red block of Fig. 7.2, the discriminative network is constructed by convolutional layers and fully connected layers similar to [Radford et al., 2015].

(a) (b) (c) (d) (e) (f) (g) (h)

Figure 7.4: Comparisons of different losses for the super-resolution. Columns: (a) Unaligned LR inputs. (b) Original HR images. (c) $\mathcal{L}_p$. (d) $\mathcal{L}_p + \mathcal{L}_f$. (e) $\mathcal{L}_p + \mathcal{L}_f + \mathcal{L}_{\mathcal{U}}$. (f) $\mathcal{L}_p + \mathcal{L}_h$. (g) $\mathcal{L}_p + \mathcal{L}_f + \mathcal{L}_h$. (h) $\mathcal{L}_p + \mathcal{L}_f + \mathcal{L}_{\mathcal{U}} + \mathcal{L}_h$. For simplicity, we omit the trade-off weights.

It is employed to determine whether an image is sampled from real face images or hallucinated ones. The discriminative loss, also known as adversarial loss, is back-propagated to update our upsampling network. In this manner, we can super-resolve more authentic HR faces, as shown in Fig. 7.4(h).

7.5.3 Loss Function

7.5.3.1 Pixel-wise Loss

Since the upsampled HR faces should be similar to the input LR faces in terms of image intensities, we employ the Euclidean distance, also known as pixel-wise ℓ_2 loss, to enforce this similarity as follows:

$$\mathcal{L}_p(w) = \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\hat{h}_i - h_i\|_F^2 = \mathbb{E}_{(l_i,h_i)\sim p(l,h)}\|\mathcal{U}_w(l_i) - h_i\|_F^2, \tag{7.1}$$

where $\hat{h}_i$ and $\mathcal{U}_w(l_i)$ both represent the upsampled faces by our MTUN, w is the parameters of MTUN, l_i and h_i denote the LR input image and its HR ground-truth counterpart respectively, $p(l,h)$ represents the joint distribution of the LR and HR face images in the training dataset, and $p(\hat{h},h)$ indicates the joint distribution of the upsampled HR faces and their corresponding HR ground-truths.

7.5.3.2 Feature-wise Loss

As mentioned in [Yu and Porikli, 2016; Ledig et al., 2017; Xu et al., 2017], only using pixel-wise ℓ_2 loss will produce over-smoothed super-resolved results. In order to achieve high-quality visual results, we also constrain the upsampled faces to share the same features as their HR counterparts. The objective function is expressed as:

$$\mathcal{L}_f(w) = \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\psi(\hat{h}_i) - \psi(h_i)\|_F^2 = \mathbb{E}_{(l_i,h_i)\sim p(l,h)}\|\psi(\mathcal{U}_w(l_i)) - \psi(h_i)\|_F^2, \tag{7.2}$$

where $\psi(\cdot)$ denotes feature maps of a layer in VGG-19 [Simonyan and Zisserman, 2014]. We use the layer ReLU32, which gives good empirical results in our experiments.

7.5.3.3 Discriminative Loss

Since super-resolution is inherently an under-determined problem, there would be many possible mappings between LR and HR images. Even imposing intensity and feature similarities may not guarantee that the upsampling network can output realistic HR face images. We employ a discriminative network to force the hallucinated faces to lie on the same manifold of real face images, and our goal is to make the discriminative network fail to distinguish the upsampled faces from real ones. Therefore, the objective function for the discriminative network $\mathcal{D}$ is formulated as:

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}(d) &= \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\left[\log \mathcal{D}_d(h_i) + \log(1-\mathcal{D}_d(\hat{h}_i))\right] \\ &= \mathbb{E}_{h_i\sim p(h)}[\log \mathcal{D}_d(h_i)] + \mathbb{E}_{\hat{h}_i\sim p(\hat{h}))}[\log(1-\mathcal{D}_d(\hat{h}_i))] \\ &= \mathbb{E}_{h_i\sim p(h)}[\log \mathcal{D}_d(h_i)] + \mathbb{E}_{l_i\sim p(l))}[\log(1-\mathcal{D}_d(\mathcal{U}_w(l_i)))],\end{aligned} \tag{7.3}$$

where d represents the parameters of the discriminative network $\mathcal{D}$, $p(h)$, $p(l)$ and $p(\hat{h})$ indicate the distributions of the real HR, LR and super-resolved faces respectively, and $\mathcal{D}_d(h_i)$ and $\mathcal{D}_d(\hat{h}_i)$ are the outputs of $\mathcal{D}$. To make our discriminative network distinguish the real faces from the upsampled ones, we maximize the loss $\mathcal{L}_{\mathcal{D}}(d)$ and the loss is back-propagated to update the parameters d.

In order to fool the discriminative network, our upsampling network should produce faces as much similar as real faces. Thus, the objective function of the upsampling network is written as:

$$\mathcal{L}_{\mathcal{U}}(w) = \mathbb{E}_{(\hat{h}_i)\sim p(\hat{h})}\left[\log \mathcal{D}_d(\hat{h}_i)\right] = \mathbb{E}_{l_i\sim p(l)}\left[\log \mathcal{D}_d(\mathcal{U}_w(l_i))\right]. \tag{7.4}$$

We minimize Eqn. 7.4 to make our upsampling network generate realistic HR face images. The loss $\mathcal{L}_{\mathcal{U}}(w)$ is back-propagated to update the parameters w.

7.5.3.4 Face Structure Loss

Unlike previous works [Yu and Porikli, 2017b; Xu et al., 2017; Yu and Porikli, 2016], we not only employ image pixel information (*i.e.*, pixel-wise and feature-wise losses) but also explore the face structure information during super-resolution. In order to achieve spatial relationships between facial components and their visibility, we estimate the heatmaps of facial components from the upsampled features as follows:

$$\mathcal{L}_h(w) = \mathbb{E}_{(l_i,h_i)\sim p(l,h)}\frac{1}{M}\sum_{k=1}^{M}\frac{1}{N}\sum_{j=1}^{N}\|\mathcal{H}_j^k(h_i) - \mathcal{H}_j^k(\tilde{\mathcal{U}}_w(l_i))\|_2^2, \tag{7.5}$$

where M is the number of the facial components, N indicates the number of Gaussian kernels in each component, $\tilde{\mathcal{U}}_w(l_i)$ is the intermediate upsampled feature maps by $\mathcal{U}$, $\mathcal{H}_j^k$ represents the j-th kernel in the k-th heatmap, and $\mathcal{H}_j^k(h_i)$ and $\mathcal{H}_j^k(\tilde{\mathcal{U}}_w(l_i))$ denote the ground-truth and estimated kernel positions in the heatmaps. Due to self-occlusions, some parts of facial components are invisible and thus N varies according to the visibility of those kernels in the heatmaps. Note that, the parameters w not only refer to the parameters in the upsampling branch but also those in the heatmap estimation branch.

7.5.3.5 Training Details

In training our discriminative network $\mathcal{D}$, we only use the loss $\mathcal{L}_\mathcal{D}(d)$ in Eqn. 7.3 to update the parameters d. Since the discriminative network aims at distinguishing upsampled faces from real ones, we maximize $\mathcal{L}_\mathcal{D}(d)$ by stochastic gradient ascent.

In training our multi-task upsampling network $\mathcal{U}$, multiple losses, *i.e.*, $\mathcal{L}_p$, $\mathcal{L}_f$, $\mathcal{L}_\mathcal{U}$ and $\mathcal{L}_h$, are involved to update the parameters w. Therefore, in order to achieve authentic super-resolved HR face images, the objective function $\mathcal{L}_\mathcal{T}$ for training the upsampling network $\mathcal{U}$ is expressed as:

$$\mathcal{L}_\mathcal{T} = \mathcal{L}_p + \alpha\mathcal{L}_f + \beta\mathcal{L}_\mathcal{U} + \mathcal{L}_h, \tag{7.6}$$

where α, β are the trade-off weights. Since our goal is to recover HR faces in terms of appearance similarity, we set α and β to 0.01. We minimize $\mathcal{L}_\mathcal{T}$ by stochastic gradient descent. Specifically, we use RMSprop optimization algorithm [Hinton, 2012] to update the parameters w and d. The discriminative network and upsampling network are trained in an alternating fashion. The learning rate r is set to 0.001 and multiplied by 0.99 after each epoch. We use the decay rate 0.01 in RMSprop.

7.5.4 Implementation Details

In our multi-task upsampling network, we employ similarity transformation estimated by STN to compensate for in-plane misalignments. In Fig. 7.2, STN is built by convolutional and ReLU layers (Conv+ReLU), max-pooling layers with a stride 2 (MP2) and fully connected layers (FC). Specifically, our STN is composed of MP2, Conv+ReLU (k5s1p0n20), MP2, Conv+ReLU (k5s1p0n20), MP2, FC+ReLU (from 80 to 20 dimensions) and FC (from 20 to 4 dimensions), where k, s and p indicate the sizes of filters, strides and paddings respectively, and n represents the channel number of the output feature maps. Our HEB is constructed by stacking four hourglass networks and we also apply intermediate supervision to the output of each hourglass network. The residual block is constructed by BN, ReLU, Conv (k3s1p1nN$_i$), BN, ReLU and Conv (k1s1p0nN$_o$), where N$_i$ and N$_o$ indicate the channel numbers of input and output feature maps.

In the experimental part, some algorithms require alignment of LR inputs, *e.g.*, [Ma et al., 2010]. Hence, we employ an STN$_0$ to align the LR face images to the upright position. STN$_0$ is composed of Conv+ReLU (k5s1p0n64), MP2, Conv+ReLU

(k5s1p0n20), FC+ReLU (from 80 to 20 dimensions), and FC (from 20 to 4 dimensions).

7.6 Experimental Results

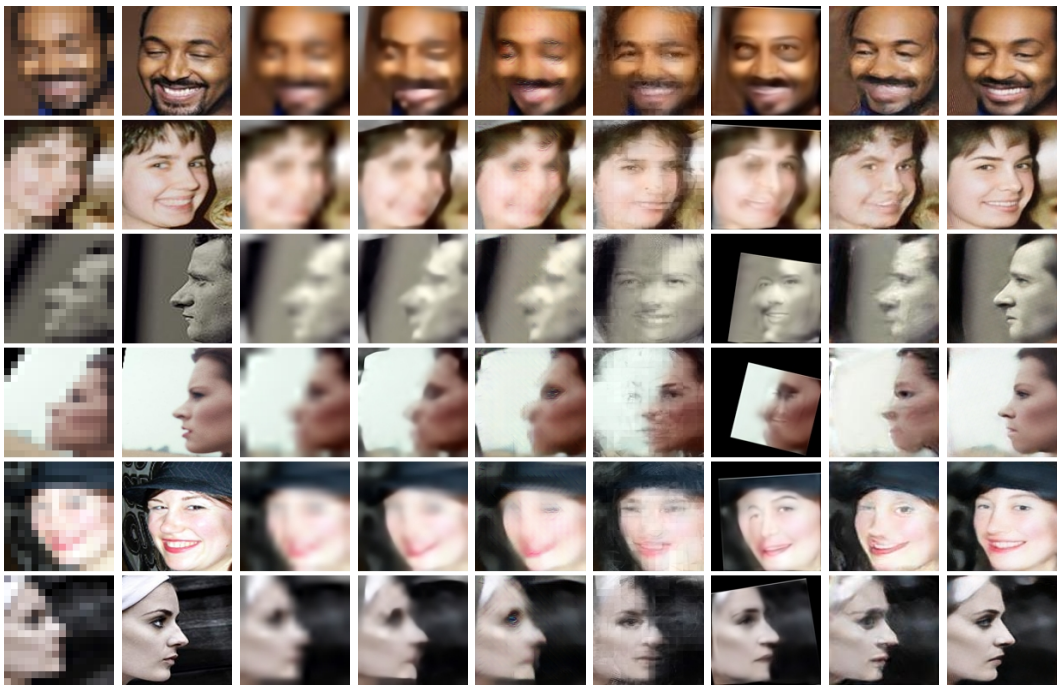
In order to evaluate the performance of our proposed network, we compare with the state-of-the-art methods [Kim et al., 2016a; Ledig et al., 2017; Ma et al., 2010; Zhu et al., 2016b; Yu and Porikli, 2017b] qualitatively and quantitatively. Kim et al. [2016a] employ a very deep convolutional network to super-resolve generic images, known as VDSR. The method of Ledig et al. [2017], dubbed SRGAN, is a generic super-resolution method, which employs the framework of generative adversarial networks and is trained with pixel-wise and adversarial losses. Ma et al. [2010] exploit position patches in the dataset to reconstruct HR images. The method of Zhu et al. [2016b], known as CBN, first localizes facial components in LR input images and then super-resolves the localized facial parts. Yu and Porikli [2017b] upsample very low-resolution unaligned face images by a transformative discriminative autoencoder (TDAE).

7.6.1 Dataset

Although there are large-scale face datasets [Liu et al., 2015; Huang et al., 2007], they do not provide structural information, *i.e.*, facial landmarks, for generating ground-truth heatmaps. In addition, we found that most of faces in the celebrity face attributes (CelebA) dataset [Liu et al., 2015], as one of the largest face datasets, are near-frontal. Hence, we use images from the Menpo facial landmark localization challenges (Menpo) [Zafeiriou et al., 2017] as well as images from CelebA to generate our training dataset. Menpo [Zafeiriou et al., 2017] provides face images in different poses and their corresponding 68 point landmarks or 39 point landmarks when some facial parts are invisible. Because Menpo only contains about $8K$ images, we also collect another $22K$ images from CelebA. The landmarks of CelebA are firstly localized by two state-of-the-art facial landmark detectors [Bulat and Tzimiropoulos, 2017a; Xiong and De la Torre, 2013] and then the erroneous localizations will be removed manually. Menpo is also aligned to the coordinates of CelebA. Then we crop the aligned faces and then resize them to 128×128 pixels as our HR ground-truth images h_i. Our LR face images l_i are generated by transforming and downsampling the HR faces to 16×16 pixels. We choose 80 percent of image pairs for training and 20 percent of image pairs for testing.

7.6.2 Qualitative Comparisons with SoA

Since Ma et al. [2010] need to align input LR faces before super-resolution and Yu and Porikli [2017b] automatically output upright HR face images, we align LR faces by a spatial transformer network STN_0 for a fair comparison and better illustration. The upright HR ground-truth images are also shown for comparison.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 7.5: Comparisons with the state-of-the-art methods. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of [Kim et al., 2016a] (VDSR). (e) The method of Ledig et al. [2017] (SRGAN). (f)The method of Ma et al. [2010]. (g) The method of Zhu et al. [2016b] (CBN). (h) The method of Yu and Porikli [2017b] (TDAE). Since TDAE is not trained with near-frontal face images, we retrain it with our training dataset. (i) Our method.

Bicubic interpolation only upsamples image intensities from neighboring pixels instead of generating new contents for new pixels. As shown in Fig. 7.5(c), bicubic interpolation fails to generate facial details.

VDSR only employs a pixel-wise ℓ_2 loss in training and does not provide an upscaling factor $8\times$. We apply VDSR to an LR face three times by an upscaling factor $2\times$. As shown in Fig. 7.5(d), VDSR fails to generate authentic facial details and the super-resolved faces are still blurry.

SRGAN is able to super-resolve an image by an upscaling factor of $8\times$ directly and employs an adversarial loss to enhance details. However, SRGAN does not take the entire face structure into consideration and thus outputs ringing artifacts around facial components, such as eyes and mouth, as shown in Fig. 7.5(e).

Ma *et al.*'s method is sensitive to misalignments in LR inputs because it hallucinates HR faces by position-patches. As seen in Fig. 7.5(f), obvious blur artifacts and ghosting facial components appear in the hallucinated faces. As the upscaling factor increases, the correspondences between LR and HR patches become inconsistent.

Table 7.1: Quantitative comparisons on the entire test dataset

Methods	PSNR	SSIM
Bicubic	18.83	0.57
VDSR [Kim et al., 2016a]	18.65	0.57
SRGAN [Ledig et al., 2017]	18.57	0.55
Ma et al. [2010]	18.66	0.53
CBN [Zhu et al., 2016b]	18.49	0.55
TDAE [Yu and Porikli, 2017b]	18.87	0.52
TDAE† [Yu and Porikli, 2017b]	21.39	0.62
Ours†	22.69	0.66
Ours‡	22.83	0.65
Ours	**23.14**	**0.68**

Thus, the super-resolved face images suffer severe blocky artifacts.

CBN first localizes facial components in LR faces and then super-resolves facial details and entire face images by two branches. As shown in Fig. 7.5(g), CBN generates facial components inconsistent with the HR ground-truth images in near-frontal faces and fails to generate realistic facial details in large poses. This indicates that it is difficult to localize facial components in LR faces accurately.

TDAE employs ℓ_2 and adversarial losses and is trained with near-frontal faces. Due to various poses in our testing dataset, TDAE fails to align faces in large poses. For a fair comparison, we retrain the decoder of TDAE with our training dataset. As visible in Fig. 7.5(h), TDAE still fails to realistic facial details due to various poses and misalignments.

Our method reconstructs authentic facial details as shown in Fig. 7.5(i). Our facial component heatmaps not only facilitate alignment but also provide spatial configuration of facial components. Therefore, our method is able to produce visually pleasing HR facial details similar to the ground-truth faces while preserving face structure.

7.6.3 Quantitative Comparisons with SoA

We also evaluate the performance of all methods quantitatively on the entire test dataset by the average PSNR and the structural similarity (SSIM) scores. Table 7.1 indicates that our method achieves superior performance compared to other methods, *i.e.*, outperforming the second best with a large margin of **1.75** dB in PSNR. Note that, the average PSNR of TDAE for its released model is only 18.87 dB because it is trained with near-frontal faces. Even after retaining TDAE, indicated by TDAE†, its performance is still inferior to our results. It also implies that our method localizes facial components and aligns LR faces more accurately with the help of our estimated

Table 7.2: Ablation study of HEB

	Position		**Depth**			
	$\mathcal{R}16$	$\mathcal{R}32$	$\mathcal{S}1$	$\mathcal{S}2$	$\mathcal{S}3$	$\mathcal{S}4$
PSNR	21.97	21.98	22.32	22.91	22.93	**23.14**
SSIM	0.63	0.64	0.64	0.67	0.67	**0.68**

Table 7.3: Ablation study on the loss

	w/o $\mathcal{L}_h$			**w/ $\mathcal{L}_h$**		
	$\mathcal{L}_p$	$\mathcal{L}_{p+f}$	$\mathcal{L}_{p+f+\mathcal{U}}$	$\mathcal{L}_p$	$\mathcal{L}_{p+f}$	$\mathcal{L}_{p+f+\mathcal{U}}$
PSNR	21.43	21.57	21.55	23.23	23.35	23.14
SSIM	0.66	0.66	0.65	0.69	0.69	0.68

heatmaps.

7.7 Analysis and Discussion

Effectiveness of HEB: As shown in Fig. 7.4(c), Fig. 7.4(d) and Fig. 7.4(e), we demonstrate that the visual results without HEB suffer from distortion and blur artifacts. By employing HEB, we can localize the facial components as seen in Fig. 7.3, and then recover realistic facial details. Furthermore, HEB provides the spatial locations of facial components and an additional constraint for face alignments. Thus we achieve higher reconstruction performance as shown in Tab. 7.3.

Feature Sizes for HEB: In our network, there are several layers which can be used to estimate facial component heatmaps, *i.e.*, feature maps of sizes 16, 32, 64 and 128, respectively. We employ HEB at different layers and demonstrate the influence of the sizes of feature maps. Due to GPU memory limitations, we only compare the super-resolution performance of using features of sizes 16 ($\mathcal{R}16$), 32 ($\mathcal{R}32$) and 64 ($\mathcal{S}4$) to estimate heatmaps. As shown in Tab. 7.2, as the resolution of feature maps increases, we obtain better super-resolution performance. Therefore, we employ the upsampled feature maps of size 64×64 to estimate heatmaps.

Depths of HEB: Table 7.2 demonstrates the performance influenced by the stack number of hourglass networks. Due to the limitation of GPU memory, we only conduct our experiments on the stack number ranging from 1 to 4. As indicated in Tab. 7.2, the final performance improves as the stack number increases. Hence, we set the stack number to 4 for our HEB.

Loss Functions: Table 7.3 also indicates the influences of different losses on the super-resolution performance. As indicated in Tab. 7.3 and Fig. 7.4, using the face structure loss improves the super-resolved results qualitatively and quantitatively. The feature-wise loss improves the visual quality and the discriminative loss makes

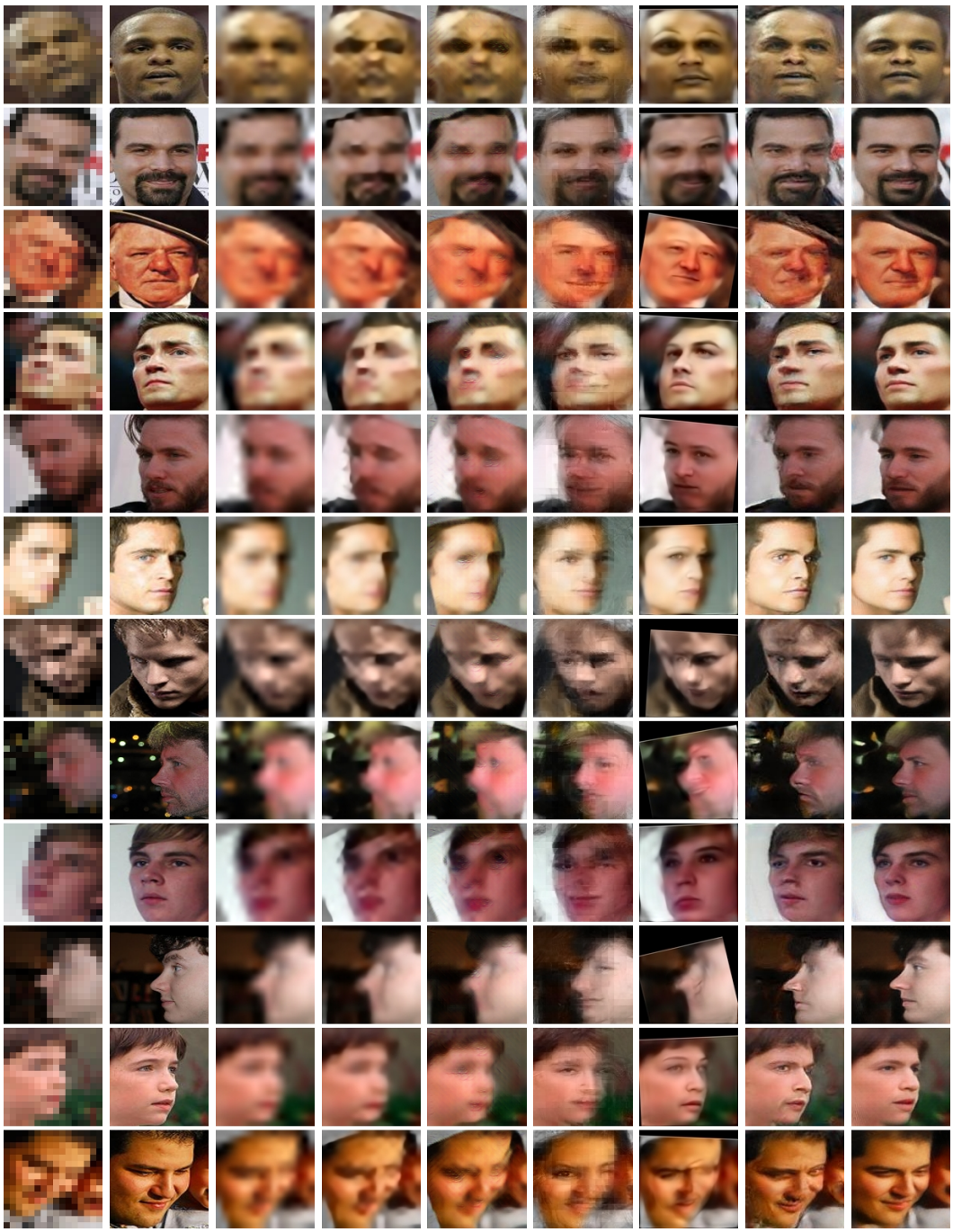
the hallucinated faces sharper and more realistic, as shown in Fig. 7.4(h).
Skip Connection and Autoencoder: Considering there are estimation errors in the heatmaps, fusing feature maps with erroneous heatmaps may lead to distortions in the final outputs. Hence, we employ a skip connection to correct the errors in Fig. 7.2. As indicated in Tab. 7.1, using the skip connection, we can improve the final quantitative result by 0.45 dB in PSNR. The result without using skip connection is indicated by Ours†. We also remove our autoencoder and upsample LR inputs directly and the result is denoted as Ours‡. As shown in Tab. 7.1, we achieve 0.31 dB improvement with the help of the autoencoder.

7.8 Conclusion

We present a novel multi-task upsampling network to super-resolve very small LR face images. We not only employ the image appearance similarity but also exploit the face structure information estimated from LR input images themselves in the super-resolution. In this manner, we preserve the spatial relationships between facial components, thus producing more authentic face images. With the help of our facial component heatmap estimation branch, our method super-resolves faces in different poses without distortions caused by erroneous facial component localization in LR inputs.

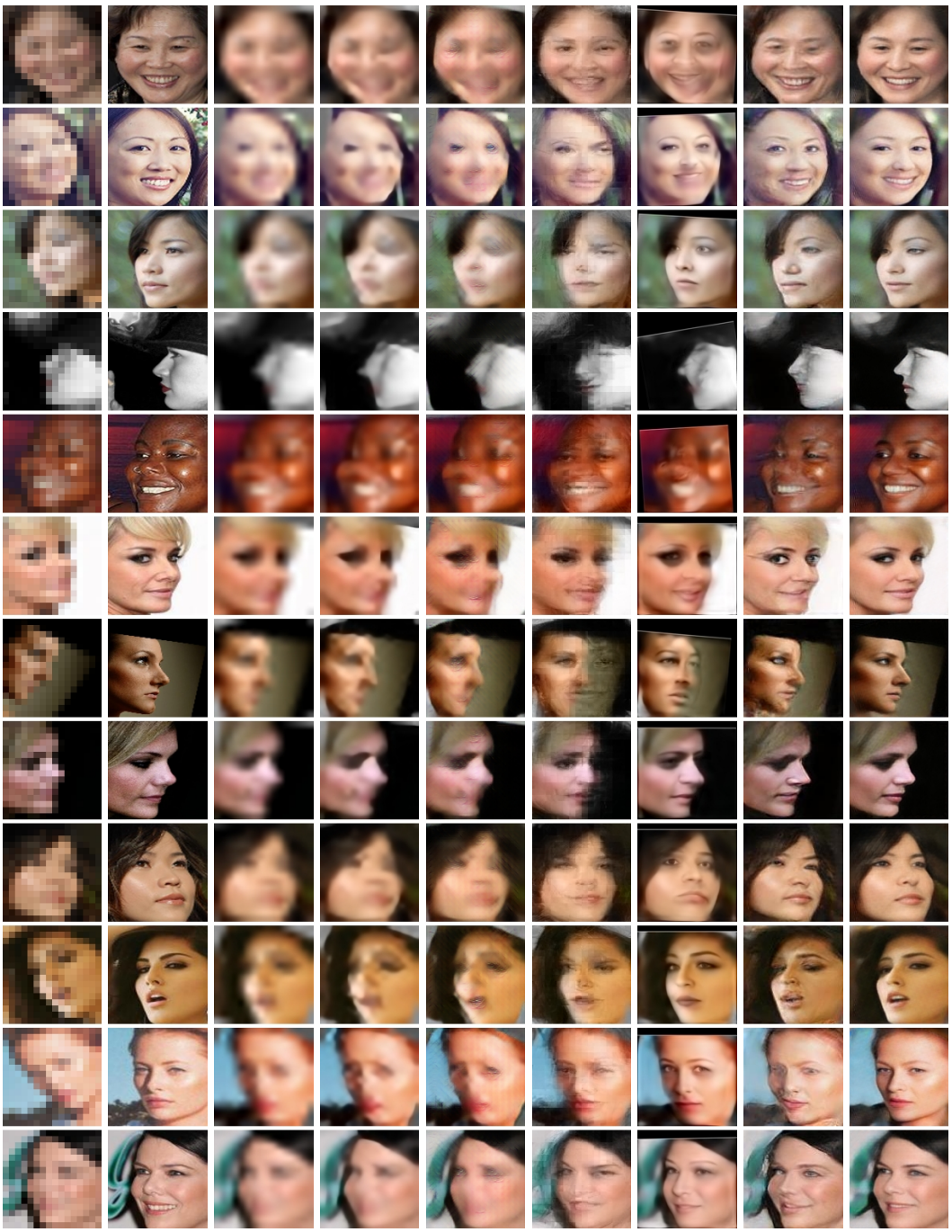
7.9 Appendix

Here, we also provide additional experimental results.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 7.6: Comparisons with the state-of-the-art methods. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Kim et al. [2016a] (VDSR). (e) The method of Ledig et al. [2017] (SRGAN). (f) The method of Ma et al. [2010]. (g) The method of Zhu et al. [2016b] (CBN). (h) The method of Yu and Porikli [2017b] (TDAE). Since TDAE is not trained on near-frontal face images, we retrain it on our training dataset. (i) Our method.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 7.7: Comparisons with the state-of-the-art methods. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) The method of Kim et al. [2016a] (VDSR). (e) The method of Ledig et al. [2017] (SRGAN). (f) The method of [Ma et al., 2010]. (g) The method of Zhu et al. [2016b] (CBN). (h) The method of Yu and Porikli [2017b] (TDAE). (i) Our method.

Chapter 8

Semantic Face Hallucination: Super-Resolving Very Low-Resolution Face Images with Supplementary Attributes

8.1 Foreword

Previous chapters propose different schemes to super-resolve faces in different situations, such as noisy and unaligned low-resolution faces and faces in large poses, but the inherently ill-posed nature of super-resolution may still lead to inaccurate upsampled faces, especially when the magnification factor is very large, *e.g.*, $8\times$. Thus, our previously proposed methods may suffer from generating reverse genders as well as rejuvenating ages of face images. In this chapter, we intend to embed facial attribute information into face super-resolution network. By leveraging high-level semantic information, we can mitigate the uncertainty caused by one to many mappings in face super-resolution and thus achieve more accurate face hallucination results.

This chapter has been published as a conference paper: Xin Yu, Basura Fernando, Richard Hartley, Fatih Porikli: Super-Resolving Very Low-Resolution Face Images with Supplementary Attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 908-917, 2018. Furthermore, we extended this work and submitted it to *IEEE Transactions on Pattern Analysis and Machine Intelligence* as a journal paper: Xin Yu, Basura Fernando, Richard Hartley, Fatih Porikli: Semantic Face Hallucination: Super-Resolving Very Low-Resolution Face Images with Supplementary Attributes.

8.2 Abstract

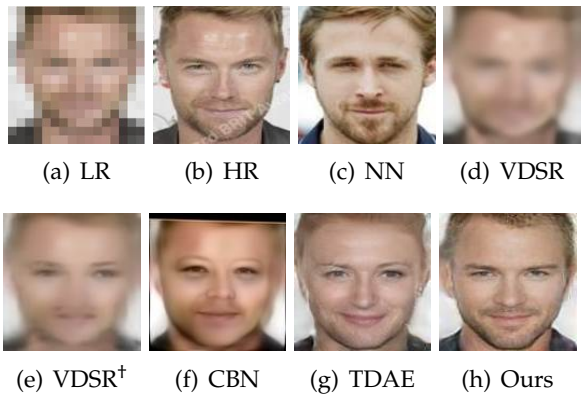
Given a tiny face image, existing face hallucination methods aim at super-resolving its high-resolution (HR) counterpart by learning a mapping from an exemplary dataset. Since a low-resolution (LR) input patch may correspond to many HR can-

didate patches, this ambiguity may lead to distorted HR facial details and wrong attributes such as gender reversal and rejuvenation. An LR input contains low-frequency facial components of its HR version while its residual face image, defined as the difference between the HR ground-truth and interpolated LR images, contains the missing high-frequency facial details. We demonstrate that supplementing residual images or feature maps with additional facial attribute information can significantly reduce the ambiguity in face super-resolution. To explore this idea, we develop an attribute-embedded upsampling network, which consists of an upsampling network and a discriminative network. The upsampling network is composed of an autoencoder with skip-connections, which incorporates facial attribute vectors into the residual features of LR inputs at the bottleneck of the autoencoder, and deconvolutional layers used for upsampling. The discriminative network is designed to examine whether super-resolved faces contain the desired attributes or not and then its loss is used for updating the upsampling network. In this manner, we can super-resolve tiny (16×16 pixels) unaligned face images with a large upscaling factor of 8× while reducing the uncertainty of one-to-many mappings remarkably. By conducting extensive evaluations on a large-scale dataset, we demonstrate that our method achieves superior face hallucination results and outperforms the state-of-the-art.

8.3 Introduction

Face images provide important information for human visual perception as well as computer analysis [Fasel and Luettin, 2003; Zhao et al., 2003]. Depending on the imaging conditions, the resolution of a face area may be unfavorably low, thus raising a critical issue that would directly impede our understanding. Motivated by this challenge, recovering high-resolution (HR) face images from their low-resolution (LR) counterparts, also known as face hallucination, has received increasing attention recently [Yu and Porikli, 2016, 2017b; Zhu et al., 2016b; Cao et al., 2017]. State-of-the-art face hallucination methods try to explore and utilize image domain priors for super-resolution. Even though they are trained on large-scale datasets benefiting from the development of deep learning techniques, ill-posed nature of the problem, which induces inherent ambiguities such as one-to-many correspondence between a given LR face and its possible HR counterparts, would still lead to drastically flawed outputs especially when the magnification factor is very large.

For instance, as shown in Fig. 8.1, the hallucinated details generated by the state-of-the-art face super-resolution methods [Zhu et al., 2016b; Yu and Porikli, 2017b] are semantically and perceptually inconsistent with the ground-truth HR image, and inaccuracies range from unnatural blur to attribute mismatches including the wrong facial hair and mixed gender features just to count a few. Note that Zhu *et al.*'s method [Zhu et al., 2016b], dubbed CBN, exploits facial structure information to super-resolve facial components while Yu and Porikli's method [Yu and Porikli, 2017b], known as TDAE, employ a class-specific discriminative prior. These methods explore either the low-level class-specific feature similarity or mid-level structure



(a) LR (b) HR (c) NN (d) VDSR

(e) VDSR† (f) CBN (g) TDAE (h) Ours

Figure 8.1: Comparison with the state-of-the-art CNN based face hallucination methods. (a) 16×16 LR input image. (b) 128×128 HR original image (not used in training). (c) The corresponding HR image of the nearest neighbor of the given LR image in the dataset after compensating for misalignments. (d) Result of VDSR [Kim et al., 2016a], which is a CNN based generic super-resolution method. (e) Result of VDSR† [Kim et al., 2016a] retrained with LR and HR face image pairs. (f) Result of CBN [Zhu et al., 2016b]. (g) Result of TDAE [Yu and Porikli, 2017b]. (h) Our result.

information as a spatial constraint in face super-resolution. However, they cannot capture high-level facial characteristic information and thus generate semantically inaccurate upsampled facial details in the outputs.

Unlike previous works, we aim to utilize high-level semantic information, *i.e.*, facial attributes, to reduce the ambiguity when super-resolving very low-resolution faces. However, a direct embedding of the binary facial attribute vector as an additional input channel to the network would still yield degraded results (see Fig. 8.3(c)). A simple combination of low-level visual information (an LR image) with high-level semantic information (attributes) in the input layer does not prevent ambiguity or provide consistent LR-HR mappings. We also note that the low-frequency facial components are visible in the LR input while the missing high-frequency details are often contained in the corresponding residual between the HR face image and the upsampled LR image (*e.g.*interpolated by Bicubic interpolation). Thus, our intuition is to incorporate facial attribute information into the residual features that are extracted from LR inputs (as seen in the yellow block of Fig. 8.2) for super-resolution of high-frequency facial details.

Driven by our observations above, we present a novel LR face image upsampling network that is able to embed facial attributes into face super-resolution. In contrast to previous face super-resolution networks [Baker and Kanade, 2000; Ma et al., 2010; Yu and Porikli, 2016, 2017a,b; Zhu et al., 2016b; Yu and Porikli, 2018], our network employs an autoencoder with skip connections to amalgamate visual features obtained from LR face images and semantic cues provided from facial attributes. It progressively upsamples the concatenated feature maps through its deconvolution-

al layers. Inspired by the architecture of StackGAN [Zhang et al., 2017b; Yan et al., 2016], we also employ a discriminative network that is used to examine whether a super-resolved face image is similar to authentic face images as well as the attributes extracted from the upsampled faces are faithful to the input attributes. As a result, our discriminative network can guide the upsampling network to incorporate the semantic information in the overall process. In this manner, the ambiguity in hallucination can be significantly reduced. Furthermore, since we apply the attribute information into the LR residual feature maps rather than concatenating it to the low-resolution input images, we can learn more consistent mappings between L-R and HR facial patterns. This allows us to generate realistic high-resolution face images as shown in Fig. 8.1(h).

Above all, the contributions of our work can be summarized as:

- We present a new semantics-embedded face hallucination framework to super-resolve LR face images. Instead of directly upsampling LR face images, we first encode LR images with facial attributes and then super-resolve the encoded feature maps.
- We propose an autoencoder with skip connections to extract residual feature maps from LR inputs and concatenate the residual feature maps with attribute information. This allows us to fuse visual and semantic information to achieve better visual results.
- Even though our network is trained to super-resolve very low-resolution face images, the upsampled HR faces can be further modified by tuning the face attributes in order to add or remove particular attributes. This property significantly increases the flexible of our face super-resolution method rather than only outputting a deterministic upsampled face.
- To the best of our knowledge, our method is the first attempt to utilize high-level semantic information, *i.e.*, facial attribute, into face super-resolution, effectively reducing the ambiguity caused by the inherent nature of this task, especially when the upscaling factor is very challenging, *i.e.*$8\times$.

8.4 Related Work

Since our work not only relates to traditional face hallucination methods but also has a close relationship with generative adversarial networks (GANs) [Goodfellow et al., 2014], we briefly review the related literatures in these two fields.

Face hallucination methods can be roughly grouped into three categories: global model based, part based, and deep learning based. Global model based methods upsample a whole LR input image, often by a learned mapping between LR and HR face images such as Principal Component Analysis (PCA). The seminal works [Baker and Kanade, 2000, 2002] progressively transfer the pixels of HR faces to the given LR face in a Gaussian Pyramid by maximizing a posteriori estimate of the ground-truth

HR face. Wang and Tang [2005] learn a linear mapping between LR and HR face subspaces, and then reconstruct an HR output with the coefficients estimated from the LR input. Liu et al. [2007] not only establish a global model for upsampling LR inputs by PCA but also exploit a local nonparametric model, *i.e.*, Markov Random Field (MRF), to enhance the facial details as well as mitigate blocky and ghosting artifacts in the upsampled faces. Kolouri and Rohde [2015] morph an HR output from the exemplar HR faces whose downsampled versions are similar to the LR input by optimal transport and subspace learning techniques. Global model based methods require LR inputs to be precisely aligned and share similar poses to exemplar HR images. However, aligning LR faces is difficult when the resolutions of LR faces are very low (*e.g.*, 16×16 pixels). Therefore, global model based algorithms produce severe artifacts when there are misalignments and pose variations in LR inputs.

Aimed at addressing pose variations, part based methods super-resolve individual facial regions separately. They either exploit reference patches or facial components to reconstruct the HR counterparts of LR inputs. Ma et al. [2010] blend position patches extracted from multiple aligned HR images to super-resolve aligned LR face images. In order to suppress image noise, the works [Yang et al., 2010; Li et al., 2014] reconstruct the position patches in LR faces by sparse coding. Tappen and Liu [2012] use SIFT flow [Liu et al., 2011] to align the facial components of LR images and reconstruct HR facial details by warping the reference HR images. Yang et al. [2013] employ a facial landmark detector to localize facial components in the LR images and then reconstruct details from the similar HR reference components. Because part based methods need to extract and align facial parts in LR images accurately, their performance degrades dramatically when LR faces are tiny. More comprehensive survey of traditional face super-resolution methods can be referred to the literature review [Wang et al., 2014].

Recently, deep learning based models achieve significant progress in several image processing tasks and are now pushing forward the state-of-the-art in super-resolution. For instance, Yu and Porikli [2018] employ deconvolutional layers to super-resolve aligned LR faces and convolutional layers to remove potential blocky artifacts. Their method also resorts an unsharp filter to enhance the edges of hallucinated faces. In order to train an end-to-end upsampling network, Yu and Porikli [2016] introduce a discriminative generative network to super-resolve aligned tiny LR face images. Instead of restoring image intensities of HR faces, Huang et al. [2017a] estimate wavelet coefficients of an upsampled HR face in the framework of generative adversarial networks. Then the upsampled HR face is reconstructed from the estimated wavelet coefficients. Zhou and Fan [2015] first extract feature maps from a blurry LR face image by a convolutional neural network (CNN) and then reconstruct a sharp HR version from the extracted feature maps. Xu et al. [2017] design a multi-class adversarial loss to super-resolve aligned LR blurry faces and text images in the framework of generative adversarial networks. Dahl et al. [2017] exploit an autoregressive generative model, also known as Pixel-RNN [Van Den Oord et al., 2016], to upscale pre-aligned LR face images.

To relax the requirement of face alignments, Yu and Porikli [2017a] interweave

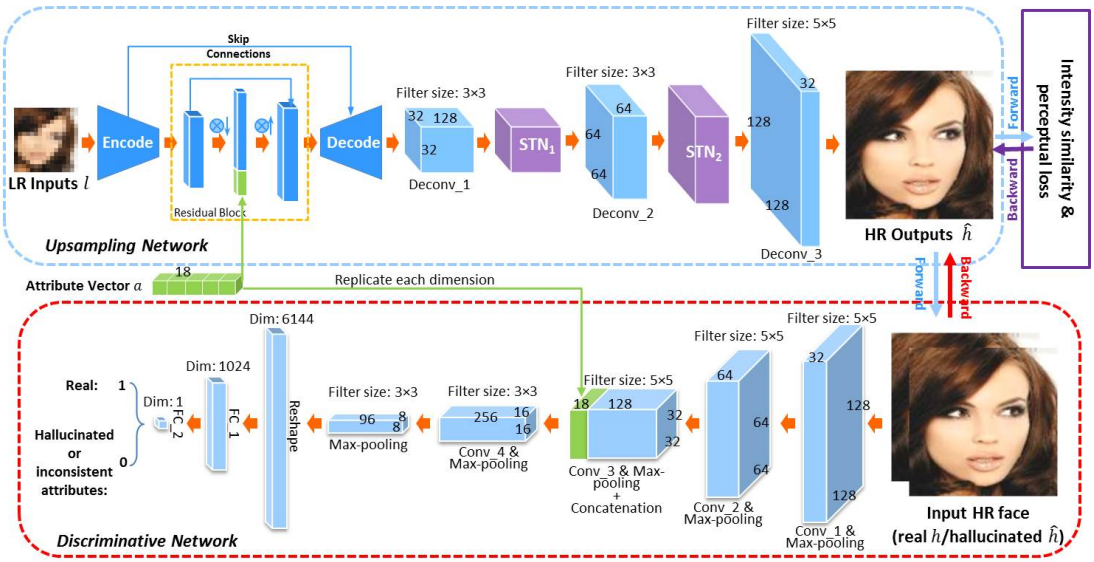


Figure 8.2: The architecture of our attribute embedded upsampling network. The network consists of two parts: an upsampling network and a discriminative network. The upsampling network takes LR faces and attribute vectors as inputs while the discriminative network takes real/super-resolved HR face images and attribute vectors as inputs.

multiple spatial transformer networks [Jaderberg et al., 2015] with the deconvolutional layers. In this manner, their method can align LR faces while super-resolving them simultaneously. Based on the observation that mild distortions and artifacts in upsampled HR faces can be mitigated in their downsampled versions, their follow-up work [Yu and Porikli, 2017b] develops a decoder-encoder-decoder structure to super-resolve noisy and unaligned LR faces. Zhu et al. [2016b] develop a cascade bi-network to localize facial components first and then super-resolve the unaligned LR faces. Chen et al. [2018] propose a two-stage network, where low-frequency components of LR faces are first super-resolved and then face priors (*i.e.*, facial component locations) are used to enrich facial details. Bulat and Tzimiropoulos [2018] employ a constraint that the landmarks of the upsampled faces should be close to the landmarks detected in their ground-truth images to handle various poses. However, due to the inherent under-determined nature of super-resolution, they may still produce results unfaithful to the ground-truths, such as gender reversal and face rejuvenation.

The method of Lee et al. [2018], concurrent with our work, also employs attributes in face super-resolution, where a feature extractor network is used to extract and combine the features of attributes and LR faces. However, their discriminative network is only designed to distinguish whether the upsampled faces are realistic or not and there is no mechanism to exam whether the attributes are successfully embedded or not.

Image generation also has a close relationship to face hallucination when gen-

(a) (b) (c) (d) (e) (f) (g) (h)

Figure 8.3: Ablation study of our network. (a) 16×16 LR input image. (b) 128×128 HR ground-truth image, its ground-truth attributes are male and old. (c) Result without using an autoencoder. Here, the attribute vectors are replicated and then concatenated with the LR input directly. (d) Result without using skip connections in the autoencoder. (e) Result by only using an ℓ_2 loss. (f) Result without using the attribute embedding but with a standard discriminative network. In this case, the network is similar to the decoder in [Yu and Porikli, 2017b]. (g) Result without using the perceptual loss. (h) Our final result.

erated images are faces. Goodfellow et al. [2014] propose a generative adversarial network (GAN) to construct images from noise, but the resolution of constructed images is limited (*i.e.*48×48 pixels) due to difficulty in training. Later, variants of GANs [Denton et al., 2015; Radford et al., 2015; Zhao et al., 2016; Arjovsky et al., 2017; Berthelot et al., 2017] have been proposed to increase the resolutions and quality of generated images. Rather than generating face images from noise, Reed et al. [2016] and Zhang et al. [2017b] generate images based on textual inputs. Yan et al. [2016] use a conditional CNN to generate faces based on attribute vectors. Perarnau et al. [2016] develop an invertible conditional GAN to generate new faces by manipulating facial attributes of the input images, while Shen and Liu [2016] change attributes of an input image on its residual image by training two generative networks in a complementary fashion. Since their methods aim at generating new face images rather than super-resolving faces, they may change the identity information. In contrast, our work focuses on obtaining HR faces faithful to LR inputs. We employ the attribute information to reduce the uncertainty in face hallucination rather than producing new face images.

8.5 Super-resolution with Attribute Embedding

Each low-resolution face image may correspond to many high-resolution face candidates during the process of increasing their resolutions. To reduce the ambiguity encountered in the super-resolution process, we present an upsampling network that takes LR faces and semantic information (*i.e.*, facial attributes) as inputs and outputs super-resolved HR faces. The entire network consists of two parts: an upsampling network and a discriminative network. The upsampling network is used for embedding facial attributes into LR input images as well as upsampling the fused feature maps. The discriminative network is used to constrain the input attributes to be encoded and the hallucinated face images to be similar to real ones. The entire ar-

chitecture of our network is illustrated in Fig. 8.2.

8.5.1 Attribute Embedded Upsampling Network

The upsampling network is composed of a facial attribute embedding autoencoder and upsampling layers (as shown in the blue frame). Previous works [Yu and Porikli, 2018, 2016, 2017a,b] only take LR images as inputs and then super-resolve them by deconvolutional layers. They do not make use of any valuable semantic information into account during super-resolution. Indeed, obtaining semantic information such as facial attributes for face images is not difficult, yet it is logical to make use of semantic information, especially for face images. For instance, we can deduce gender information from the outfits. Unlike previous works, we incorporate low-level visual and high-level semantic information in face super-resolution to reduce the ambiguity of the mappings between LR and HR images.

Rather than concatenating LR input images with attribute vectors directly, in our proposed attribute embedding network we employ a convolutional-deconvolutional autoencoder with skip connections [Long et al., 2015] to fuse visual features and attribute vectors. Due to the skip connections, we can utilize residual features obtained from LR input images to incorporate the attribute vectors. Specifically, at the bottleneck of the autoencoder, we concatenate the attribute vector with the residual feature vector as illustrated in the green and blue vectors of Fig. 8.2. As shown in Fig. 8.3(d), when we encode attributes with the feature maps of LR faces at the bottleneck of the autoencoder without using the skip connections instead of residual feature maps, artifacts appear in the smooth regions of the super-resolved result. After combining the residual feature vectors of LR inputs with the attribute vectors, we employ deconvolutional layers to upsample the concatenated feature maps. Since LR input images may undergo misalignments, such as in-plane rotations, translations and scale changes, we use spatial transformer networks (STNs) [Jaderberg et al., 2015] to compensate for misalignments similar to [Yu and Porikli, 2017a,b], as shown in the purple blocks in Fig. 8.2. Since STNs employ bilinear interpolation to re-sample images, they will blur LR input images, as reported in [Yu and Porikli, 2017a]. Therefore, we only employ STNs in the upsampling layers.

To constrain the appearance similarity between the super-resolved faces and their HR ground-truth counterparts, we exploit a pixel-wise Euclidean distance loss, also known as pixel-wise ℓ_2 loss, and a feature-wise ℓ_2 loss, dubbed perceptual loss [Johnson et al., 2016]. The pixel-wise ℓ_2 loss is employed to enforce image intensity similarity between the upsampled HR faces and their ground-truth images. As reported in [Yu and Porikli, 2016], deconvolutional layers supervised by an ℓ_2 loss tend to output over-smoothed results as shown in Fig. 8.3(e). Since the perceptual loss measures Euclidean distance between features of two images, we use it to constrain feature similarity between the upsampled faces and their ground-truth ones. We use VGG-19 [Simonyan and Zisserman, 2014] to extract features from images (please refer to Sec. 8.5.3 for more details). Without the help of the perceptual loss, the network tends to produce ringing artifacts to mimic facial details, such as wrinkles, as seen in

Fig. 8.3(g).

8.5.2 Discriminative Network

In order to force the upsampling work to encode facial attribute information, we employ a conditional discriminative network. Specifically, the discriminative network is designed to distinguish whether the attributes of super-resolved face images are faithful to the desired attributes embedded in the upsampling network or not and is used to constrain the upsampled images to be similar to HR real face images too.

Even though our autoencoder concatenates attribute vectors with residual feature maps of the LR inputs, the upsampling network may simply learn to ignore them, *e.g.*, the weights corresponding to the semantic information are zeros. Therefore, we need to design a discriminator network to enforce semantic attribute information into the generative process. As shown in Fig. 8.3(f), by employing a standard discriminative network [Yu and Porikli, 2016; Radford et al., 2015], the output HR face still looks like a female face even if the expected figure should be an old male. It implies that the attribute information is not well embedded. Therefore, simply embedding a semantic vector into LR inputs may increase the ambiguity or deviate the learned mapping between the LR and correct HR face images.

We present a discriminative network to enforce the input attribute information to be embedded in LR inputs, thus generating the desired attributes in the hallucinated face images. As shown in the red frame of Fig. 8.2, our discriminative network is constructed by convolutional layers and fully connected layers. HR face images (real and upsampled faces) are fed into the network while attribute information is also fed into the middle layer of the network as conditional information. Here, an attribute vector is replicated and then concatenated with the feature maps of images. Because CNN filters in the first layers mainly extract low-level features while filters in higher layers extract image patterns or semantic information [Zeiler and Fergus, 2014], we concatenate the attribute information with the extracted feature maps on the third layer, which yields good empirical results in our experiments. If the extracted features do not comply with the input attribute information, the discriminative network ought to pass that information to the upsampling network. Our discriminative network is a binary classifier which is trained with a binary cross-entropy loss. With the help of the discriminative network, the attribute information can be embedded into the upsampling network. As shown in Fig. 8.3(h), our final result is faithful to the age and gender of the ground-truth image.

8.5.3 Training Procedure

Our face super-resolution network is trained in an end-to-end fashion. We use an LR face image denoted by l_i and its ground-truth attribute label vector a_i as the inputs and the corresponding HR ground-truth face image h_i as the target. Note that, since our network aims at super-resolving very low-resolution face images rather than manipulating facial attributes of HR face images, we only feed the correct attributes

of LR face images into the upsampling network in the training phase.

In training the entire network, we employ a binary cross-entropy loss to update our discriminative network and then train the upsampling network using a pixel-wise ℓ_2 loss, a perceptual loss and the discriminative loss obtained from our discriminative network. Therefore, we first update the parameters of the discriminative network and then the parameters of the upsampling network because the upsampling network relies on the loss back-propagated from the discriminative network to update its weights.

8.5.3.1 Training Discriminative Network

Our discriminative network is designed to embed attribute information into the upsampling network as well as to force the super-resolved HR face images to be authentic. Similar to [Yan et al., 2016; Zhang et al., 2017b], our goal is to make the discriminative network be able to tell whether super-resolved faces contains the desired attributes or not but fail to distinguish hallucinated faces from real ones. Hence, in order to train the discriminative network, we take real HR face images h_i and their corresponding ground-truth attributes a_i as positive sample pairs $\{h_i, a_i\}$. Negative data is constructed from super-resolved HR faces $\hat{h}_i$ by our upsampling network and their ground-truth attributes a_i as well as real HR faces and mismatched attributes $\tilde{a}_i$. Therefore, the negative sample pairs consist of both $\{\hat{h}_i, a_i\}$ and $\{h_i, \tilde{a}_i\}$. The objective function for the discriminative network $\mathcal{L}_D$ is expressed as:

$$
\begin{aligned}
\mathcal{L}_{\mathcal{D}} = & - \mathbb{E}\left[\log \mathcal{D}_d(h, a)\right] \\
& - \mathbb{E}\left[\log(1 - \mathcal{D}_d(\hat{h}, a)) + \log(1 - \mathcal{D}_d(h, \tilde{a}))\right] \\
= & - \mathbb{E}_{(h_i, a_i)\sim p(h,a)}\left[\log \mathcal{D}_d(h_i, a_i)\right] \\
& - \mathbb{E}_{(h_i, \tilde{a}_i)\sim p(h,\tilde{a})}\left[\log(1 - \mathcal{D}_d(h_i, \tilde{a}_i))\right] \\
& - \mathbb{E}_{(\hat{h}_i, a_i)\sim p(\hat{h},a)}\left[\log(1 - \mathcal{D}_d(\hat{h}_i, a_i))\right] \\
= & - \mathbb{E}_{(h_i, a_i)\sim p(h,a)}\left[\log \mathcal{D}_d(h_i, a_i)\right] \\
& - \mathbb{E}_{(h_i, \tilde{a}_i)\sim p(h,\tilde{a})}\left[\log(1 - \mathcal{D}_d(h_i, \tilde{a}_i))\right] \\
& - \mathbb{E}_{(l_i, a_i)\sim p(l,a)}\left[\log(1 - \mathcal{D}_d(\mathcal{U}_t(l_i, a_i), a_i))\right],
\end{aligned}
\tag{8.1}
$$

where d represents the parameters of the discriminative network $\mathcal{D}$, $\mathcal{D}_d(h_i, a_i)$, $\mathcal{D}_d(\hat{h}_i, a_i)$ and $\mathcal{D}_d(h_i, \tilde{a}_i)$ are the outputs of $\mathcal{D}$, $\mathcal{U}_t(l_i)$ is the output of our upsampling network and t represents the parameters of our upsampling network. In addition, $p(h, a)$ represents the joint distribution of positive sample pairs, $p(\hat{h}, a)$ as well as $p(h, \tilde{a})$ represent the joint distributions of negative sample pairs, and $p(l, a)$ represents the joint distribution of the LR input faces and their ground-truth attributes.

Since all the layers in our discriminative network are differentiable, back-propagation is used to calculate the gradients with respect to the parameters of the discriminative

network d. Thus, we minimize $\mathcal{L}_{\mathcal{D}}$ by RMSprop [Hinton, 2012] as follows:

$$\begin{aligned} \Delta^{i+1} &= \gamma\Delta^{i} + (1-\gamma)(\frac{\partial\mathcal{L}_{\mathcal{D}}}{\partial d})^2, \\ d^{i+1} &= d^{i} - r\frac{\partial\mathcal{L}_{\mathcal{D}}}{\partial d}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}, \end{aligned} \tag{8.2}$$

where r and γ represent the learning rate and the decay rate respectively, i indicates the index of the iterations, Δ is an auxiliary variable, and ϵ is set to 10^{-8} to avoid division by zero.

8.5.3.2 Training Upsampling Network

Since our upsampling network aims at super-resolving LR input images, we only feed our upsampling network with LR face images l_i and their corresponding attributes a_i as inputs. To constrain the upsampled faces to be similar to the HR ground-truth face images, we employ a pixel-wise ℓ_2 loss on image intensities, expressed as:

$$\begin{aligned} \mathcal{L}_{pix} &= \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\hat{h}_i - h_i\|_F^2 \\ &= \mathbb{E}_{(l_i,a_i,h_i)\sim p(l,a,h)}\|\mathcal{U}_t(l_i,a_i) - h_i\|_F^2, \end{aligned} \tag{8.3}$$

where $p(\hat{h},h)$ is the joint distribution of the upsampled faces and their ground-truth counterparts and $p(l,h,a)$ represents the joint distribution of the LR and HR face images and their corresponding attributes in the training dataset.

As mentioned in Sec. 8.5.1, we also employ a perceptual loss $\mathcal{L}_{feat}$ to enforce the feature similarity between the super-resolved faces and their corresponding ground-truths, written as:

$$\begin{aligned} \mathcal{L}_{feat} &= \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\Phi(\hat{h}_i) - \Phi(h_i)\|_F^2 \\ &= \mathbb{E}_{(l_i,a_i,h_i)\sim p(l,a,h)}\|\Phi(\mathcal{U}_t(l_i,a_i)) - \Phi(h_i)\|_F^2, \end{aligned} \tag{8.4}$$

where $\Phi(\cdot)$ denotes feature maps extracted by the ReLU32 layer in VGG-19 [Simonyan and Zisserman, 2014], which gives good empirical performance in our experiments.

To enforce the upsampling network to encode the attribution information, a discriminative loss $\mathcal{L}_{dis}$ is also exploited as follows:

$$\begin{aligned} \mathcal{L}_{dis} &= -\mathbb{E}_{(\hat{h}_i,a_i)\sim p(\hat{h},a)}\log(\mathcal{D}_d(\hat{h}_i,a_i)) \\ &= -\mathbb{E}_{(l_i,a_i)\sim p(l,a)}\log(\mathcal{D}_d(\mathcal{U}_t(l_i,a_i),a_i)), \end{aligned} \tag{8.5}$$

where $p(\hat{h},a)$ indicates the joint distribution of the upsampled faces and their corresponding attributes.

All the above three losses are used to update the parameters of our upsampling

Algorithm 3 Training procedure of our entire network

Input: minibatch size N, LR and HR face image pairs $\{l_i, h_i\}$ and their corresponding attributes a_i, maximum number of iterations K.

1: **while** iter $< K$ **do**
2: Choose one minibatch of LR and HR image pairs $\{l_i, h_i\}$ and their corresponding attributes, $i = 1, \ldots, N$.
3: Generate one minibatch of HR face images $\hat{h}_i$ from $\{l_i, a_i\}, i = 1, \ldots, N$, where $\hat{h}_i = \mathcal{U}_t(l_i, a_i)$.
4: Generate mismatched attributes $\tilde{a}_i$ from a_i by randomly permuting one dimension in an attribute vector.
5: Generate positive sample pairs $\{h_i, a_i\}$ and negative sample pairs $\{\hat{h}_i, a_i\}$ and $\{h_i, \tilde{a}_i\}$.
6: Update the parameters of the discriminative network $\mathcal{D}_d$ by using Eqn. 8.1 and Eqn. 8.2.
7: Update the parameters of the upsampling network $\mathcal{U}_t$ by using Eqn. 8.6 and Eqn. 8.7.
8: **end while**

Output: Our attribute embedded upsampling network.

network, and the total loss $\mathcal{L}_{\mathcal{U}}$ is expressed as:

$$\mathcal{L}_{\mathcal{U}} = \mathcal{L}_{pix} + \alpha \mathcal{L}_{feat} + \beta \mathcal{L}_{dis}, \tag{8.6}$$

where α is a weight term which trades off between the image intensity similarity and the feature similarity, and β is a weight which trades off between the appearance similarity and the attribute similarity. Here, we also employ RMSprop to update the parameters of our upsampling network:

$$\begin{aligned} \Delta^{i+1} &= \gamma \Delta^i + (1-\gamma)(\frac{\partial \mathcal{L}_{\mathcal{U}}}{\partial t})^2, \\ t^{i+1} &= t^i - r \frac{\partial \mathcal{L}_{\mathcal{U}}}{\partial t} \frac{1}{\sqrt{\Delta^{i+1} + \epsilon}}. \end{aligned} \tag{8.7}$$

After updating the upsampling network, we can obtained upsampled face images in better quality. Hence, we use HR faces hallucinated by the newly updated upsampling network to train the discriminative network again. By updating these two network alternatingly, we can achieve realistic super-resolved face images including correct attributes. The entire training procedure is illustrated in Algorithm 3.

8.5.4 Super-Resolving LR Inputs with Attributes

The discriminative network $\mathcal{D}$ is only required in the training phase. In the super-resolving (testing) phase, we take LR face images and their corresponding attributes as the inputs of the upsampling network $\mathcal{U}$, and the outputs of $\mathcal{U}$ are the hallucinated HR face images. In addition, although the attributes are binary values, *i.e.*, either 0 or

1, in training, the attributes can be further scaled, such as negative values or values exceeding 1, to manipulate the final super-resolved results according to the users' descriptions in the testing phase.

8.5.5 Implementation Details

The detailed architectures of the upsampling and discriminative networks are illustrated in Fig. 8.2. We employ convolutional layers with kernels of size 4×4 in a stride 2 in the encoder and deconvolutional layers with kernels of size 4×4 in a stride 2 in the decoder. The feature maps in our encoder will be passed to the decoder by skip connections. We also use the same architectures of the STN layers in [Yu and Porikli, 2017b] to align feature maps. Specifically, the STN layers are constructed by convolutional and ReLU layers (Conv+ReLU), max-pooling layers with a stride 2 (MP2) and fully connected layers (FC). STN_1 layer is cascaded by: MP2, Conv+ReLU (with the filter size: $128\times20\times5\times5$), MP2, Conv+ReLU (with the filter size: $20\times20\times5\times5$), FC+ReLU (from 80 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_2 is cascaded by: MP2, Conv+ReLU (with the filter size: $64\times128\times5\times5$), MP2, Conv+ReLU (with the filter size: $128\times20\times5\times5$), MP2, Conv+ReLU (with the filter size: $20\times20\times3\times3$), FC+ReLU (from 180 to 20 dimensions) and FC (from 20 to 4 dimensions). We do not use zero-padding in the convolution operations.

We set the learning rate to 0.001 and multiplied by 0.95 after each epoch, and α is set to 0.01. As suggested by Yu and Porikli [2017b], we also set β to 0.01 and gradually decrease it by a factor 0.995, thus emphasizing the importance of the appearance similarity. On the other hand, in order to guarantee the attributes to be embedded in the training phase, we stop decreasing β when it is lower than 0.005.

8.6 Experiments

We evaluate our network qualitatively and quantitatively, and compare with the state-of-the-art methods [Kim et al., 2016a; Ma et al., 2010; Zhu et al., 2016b; Yu and Porikli, 2017b; Ledig et al., 2017]. Kim *et al.*'s method [Kim et al., 2016a], dubbed VDSR, is a generic CNN based super-resolution method. Ledig *et al.*'s method [Ledig et al., 2017], also known as SRGAN, is also a generic CNN based super-resolution method, which employs an adversarial loss to enhance the super-resolved details. Since VDSR and SRGAN are trained on natural images, they may not capture LR facial patterns well for face super-resolution. We retrain VDSR and SRGAN on entire face images for fair comparisons. Ma et al. [2010] exploit position-patches in the exemplary dataset to reconstruct HR images. Zhu et al. [2016b] employ a cascaded deep convolutional neural network to hallucinate facial components of LR face images. Yu and Porikli [2017b] use a decoder-encoder-decoder structure to super-resolve unaligned LR faces.

Table 8.1: Quantitative evaluations on the test dataset.

Method	PSNR	SSIM
Bicubic	19.23	0.56
VDSR [Kim et al., 2016a]	19.58	0.57
VDSR† [Kim et al., 2016a]	20.12	0.57
SRGAN† [Ledig et al., 2017]	19.06	0.57
Ma et al. [2010]	19.11	0.54
CBN [Zhu et al., 2016b]	18.77	0.54
TDAE [Yu and Porikli, 2017b]	20.40	0.57
Ours	**21.82**	**0.62**

8.6.1 Dataset

We use the Celebrity Face Attributes (CelebA) dataset [Liu et al., 2015] to train our network because CelebA dataset contains over 220K face images and also provides 40 binary-value attributes for each face image. Unlike previous face generation methods [Perarnau et al., 2016; Yan et al., 2016; Shen and Liu, 2016], our network focuses on super-resolving LR faces by exploiting facial attributes. Hence, we only choose the attributes related to facial details, such as gender, age and beard information, rather than the attributes which can be directly extracted from LR faces, such as hair and skin colors, and are not related to facial details, such as wearing hats, glasses and earrings. In particular, we select the 18 attributes from the 40 attributes, including 5 o'clock shadow, arched eyebrow, bags under eyes, big lips, big nose, bushy eyebrows, double chin, goatee, heavy makeup, high cheekybone, male, mouth open, mustache, narrow eyes, no beard, pointy nose, sideburns and young. In this way, we reduce the potential inconsistency between visual and semantic information imposed by the supplementary attributes.

When generating the LR and HR face pairs, we select 170K cropped face images from the CelebA dataset, and then resize them to 128×128 pixels as HR images. We manually transform the HR images, including rotations, translations and scale changes, and then downsample HR images to 16×16 pixels to attain their corresponding LR images. We use 160K LR and HR face pairs and their corresponding attributes for training, 2K LR and HR image pairs and their attributes for validation, and 2K LR face images and their ground-truth attributes for testing.

8.6.2 Qualitative Comparison with the SoA

Some algorithms [Ma et al., 2010; Kim et al., 2016a; Ledig et al., 2017] need the alignments of LR inputs before face super-resolution while the method of Yu and Porikli [2017b] automatically generates upright HR face images. For a fair comparison and better illustration, we employ a spatial transformer network STN_0 to align LR faces. The aligned upright HR ground-truth images are shown for comparison. As reported

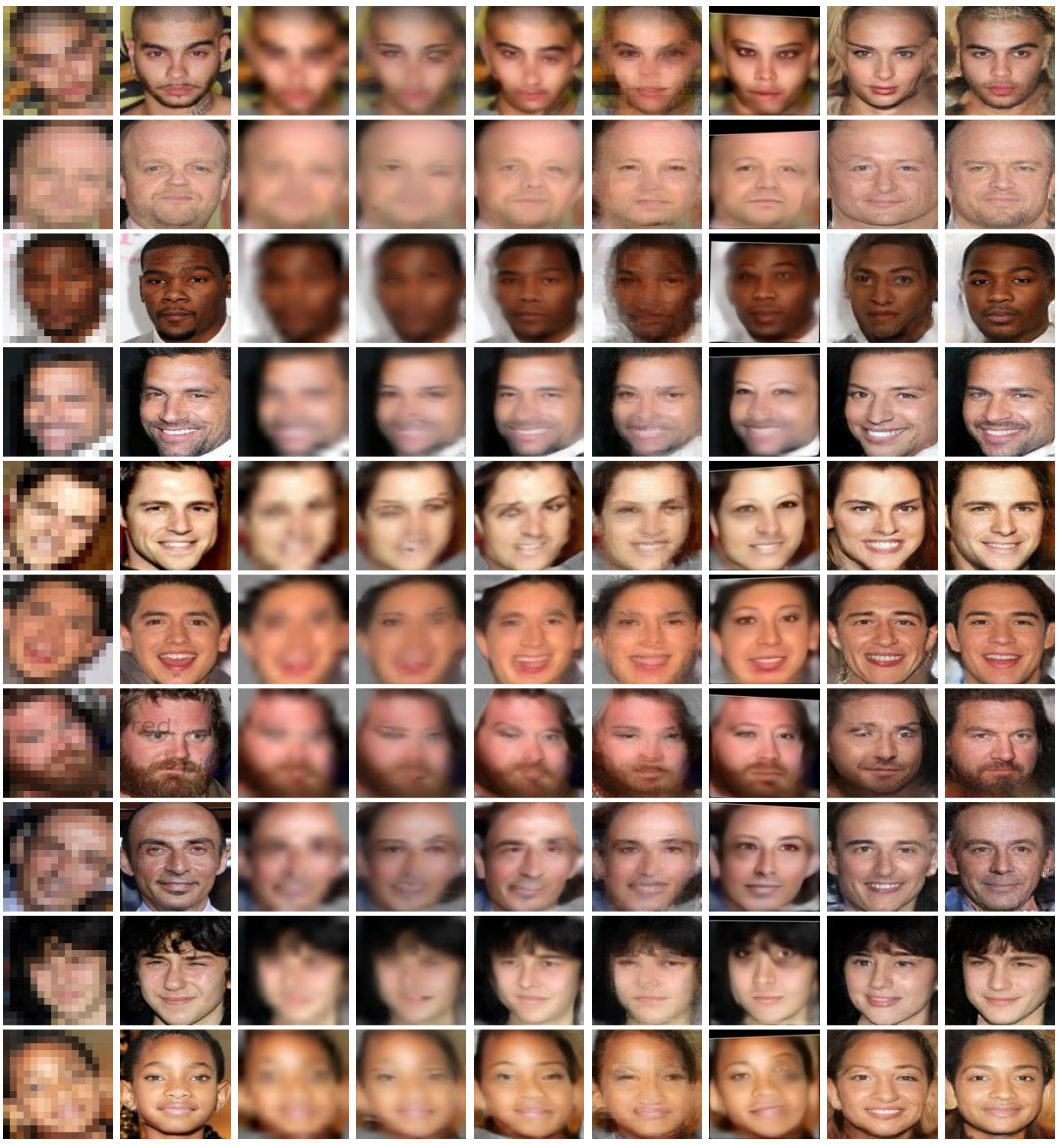


Figure 8.4: Comparison with the state-of-the-arts methods on male images. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Results of Kim et al. [2016a] (VDSR). (e) Results of Ledig et al. [2017] (SRGAN). (f) Results of Ma et al. [2010]. (g) Results of Zhu et al. [2016b] (CBN). (h) Results of Yu and Porikli [2017b] (TDAE). (i) Our results.

in [Yu and Porikli, 2017a,b], LR faces aligned by STN_0 may still suffer misalignments. Therefore, we employ multiple STNs in the upsampling network to reduce misalignments similar to [Yu and Porikli, 2017a,b]. The only difference between STN_0 and STN_1 is that the first MP2 operation in STN_1 is removed in STN_0 and the input

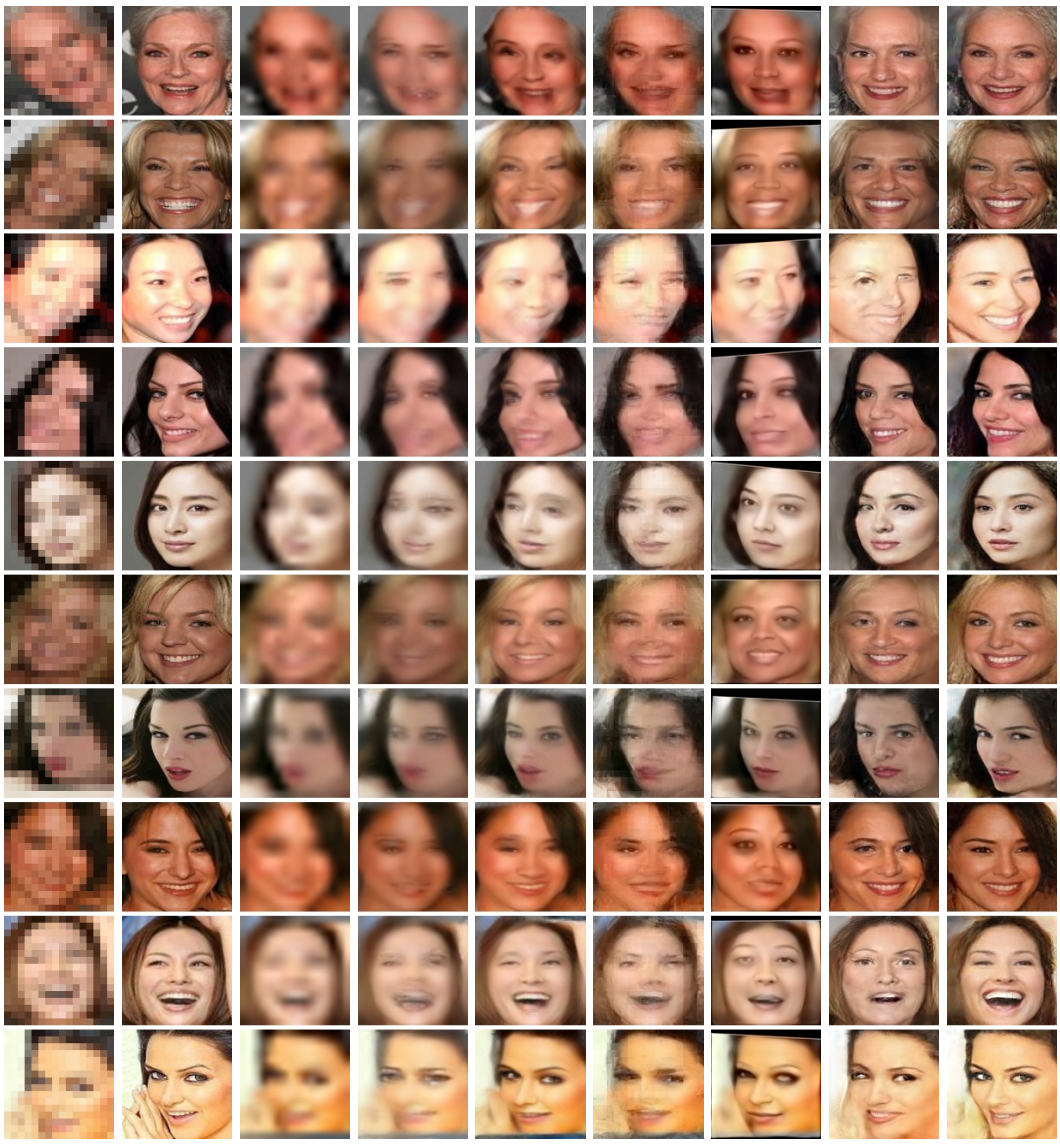
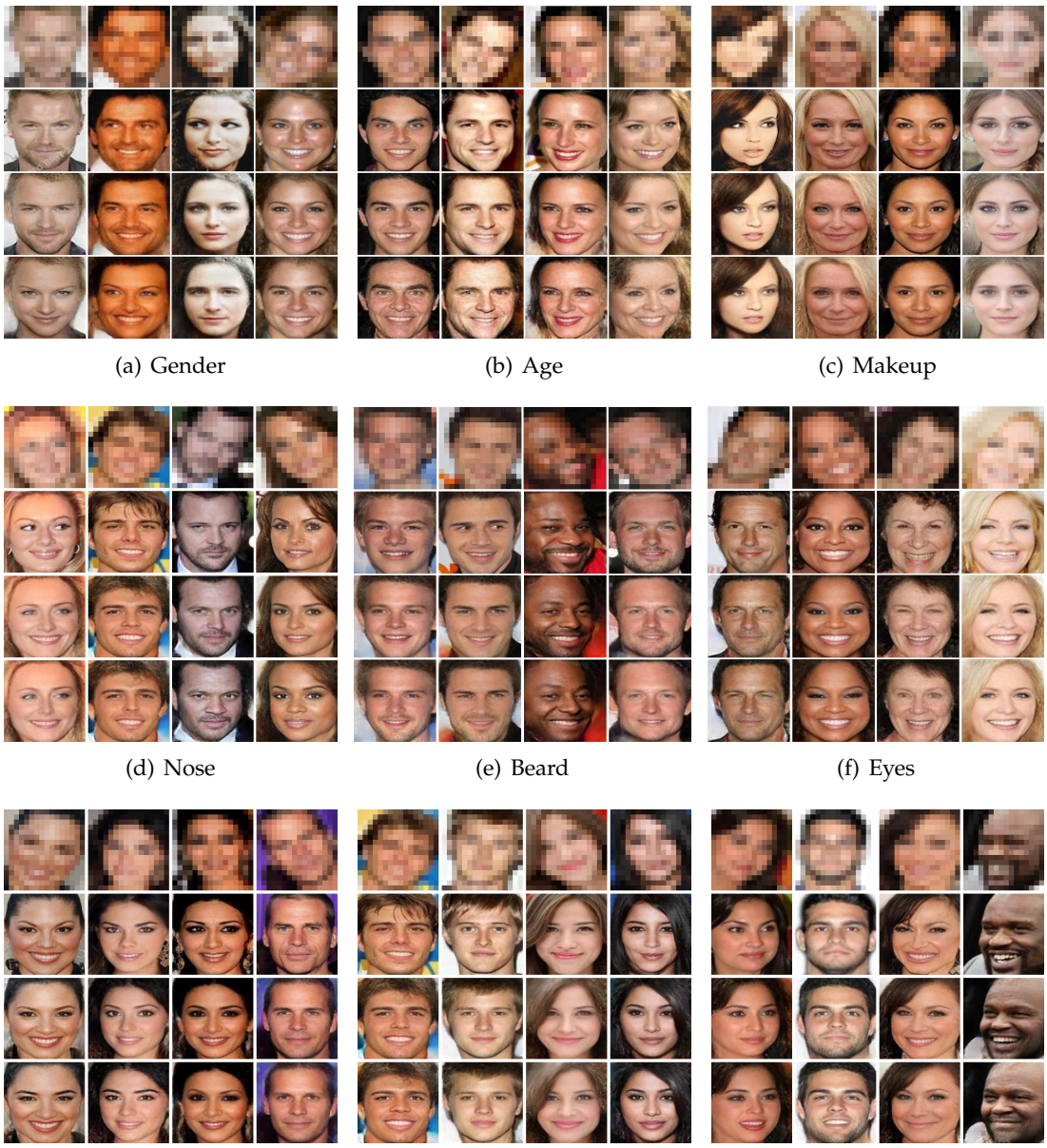


Figure 8.5: Comparison with the state-of-the-arts methods on female images. (a) Unaligned LR inputs. (b) Original HR images. (c) Bicubic interpolation. (d) Results of Kim et al. [2016a] (VDSR). (e) Results of Ledig et al. [2017] (SRGAN). (f) Results of Ma et al. [2010]. (g) Results of Zhu et al. [2016b] (CBN). (h) Results of Yu and Porikli [2017b] (TDAE). (i) Our results.

channel is 3.

Bicubic upsampling only interpolates new pixels from neighboring pixels rather than hallucinating new contents for new pixels. Furthermore, the resolution of our input face images is very small, and little information is contained in the input im-



(a) Gender (b) Age (c) Makeup

(d) Nose (e) Beard (f) Eyes

(g) Eyebrows (h) Lips (i) Mouth

Figure 8.6: Our method can fine-tune the super-resolved results by adjusting the attributes. From top to bottom: the LR input faces, the HR ground-truth faces, our results with ground-truth attributes, our results by adjusting attributes. (a) Reversing genders of super-resolved faces. (b) Aging upsampled faces. (c) Removing makeups. (d) Changing noses. (The first two columns: making noses pointy, and the last two columns: making noses bigger.) (e) Adding and removing beard. (f) Narrowing and opening eyes. (g) Making and removing bushy Eyebrows. (h) Making lips bigger. (i) Opening and closing mouths.

Table 8.2: Classification results impacted by tuning attributes.

Attributes	GT Attr. Acc.	Increased Attr. Acc.	Decreased Attr. Acc.
Male	100%	100%	0%
Young	100%	100%	0%
Makeup	91%	100%	2.9%
Big nose	42%	100%	8.3%
Beard	100%	100%	0%
Narrow eyes	67%	100%	0%
Bushy eyebrows	88%	100%	0%
Big lips	56%	94%	0%
Mouth open	100%	100%	0%

ages. As shown in Fig. 8.4(c) and Fig. 8.5(c), conventional bicubic interpolation fails to generate facial details. The upsampled faces also suffer from obvious skew artifacts. This indicates that it is difficult to align very low-resolution faces accurately by a single STN_0.

Kim et al. [2016a] present a deep CNN for generic purpose super-resolution, known as VDSR. Because VDSR is trained on natural image patches and does not provide an upscaling factor of $8\times$, it cannot capture the global face structure, as shown in Fig. 8.1(d). We re-train the model with an upscaling factor of $8\times$ on face images, marked as $\text{VDSR}^\dagger$. As shown in Fig. 8.4(d) and Fig. 8.5(d), this method also suffers from the distortion artifacts in the results due to misalignments. Furthermore, since $\text{VDSR}^\dagger$ is only trained by a pixel-wise ℓ_2 loss, it outputs overly smoothed results as seen in Fig. 8.4(d) and Fig. 8.5(d).

Ledig et al. [2017] develop a CNN based generic super-resolution method, dubbed SRGAN. In order to avoid producing overly smoothed super-resolved results, SRGAN employs an adversarial loss [Goodfellow et al., 2014; Radford et al., 2015]. Since original SRGAN is also trained on generic image patches, we also fine-tune SRGAN with entire face images for a fair comparison, named as $\text{SRGAN}^\dagger$. As seen in Fig. 8.4(e) and Fig. 8.5(e), SRGAN is able to capture LR facial patterns and achieves sharper upsampled results compared to VDSR. However, misalignments in LR faces result in severe distortions in the final results.

Ma et al. [2010] super-resolve HR faces by position-patches from HR exemplar face images. Thus, their method is sensitive to misalignments in LR inputs. As seen in Fig. 8.4(f) and Fig. 8.4(f), there are obvious blur artifacts along the profiles of hallucinated faces. In addition, the correspondences between LR and HR patches become inconsistent as the upscaling factor increases. Hence, severe blocky artifacts appear on the boundaries of different patches.

Zhu et al. [2016b] develop a cascaded bi-network (CBN) to super-resolve very low-resolution face images. CBN firstly localizes facial components in LR faces and then super-resolves facial details by a local network and entire face images by a global

Table 8.3: Ablation study on our proposed network

	EN_s	inAttr	woAttr	woAE	noSkip	Ours
PSNR	20.03	21.43	21.64	21.03	21.21	**21.82**
SSIM	0.55	0.60	0.60	0.58	0.58	**0.62**

network. As shown in the first and fifth rows of Fig. 8.4(g), CBN is able to generate HR facial components, but it also hallucinates feminine facial details in male face images, *e.g.*, eye lines appear in male faces as seen in the fifth row of Fig. 8.4(g). Furthermore, CBN fails to super-resolve faces of senior people, as shown in the first row of Fig. 8.5(g). As the upscaling factor increases, the facial details in LR faces become more ambiguous. Therefore, it is difficult to recover the facial details of senior people, such as wrinkles and age spots which are even hard to observe in LR faces.

Yu and Porikli [2017b] exploit a transformative discriminative autoencoder (T-DAE) to upsample very low-resolution face images. They also employ deconvolutional layers to upsample LR faces as well as STN layers to align LR faces, but their discriminative network is only used to force the upsampling network to produce sharper results without imposing any high-level semantic information, *e.g.*, facial attributes, in super-resolution. As visible in Fig. 8.4(h) and Fig. 8.5(h), their method also reverses the genders of the upsampled faces as well as suffers from facial rejuvenation.

In contrast, our method is able to reconstruct authentic facial details as shown in Fig. 8.4(i) and Fig. 8.5(i). Even though there are different poses, facial expressions and ages in the input faces, our method still produces visually pleasing HR faces which are similar to the ground-truth faces without suffering gender reversal and facial rejuvenation. For instance, we can super-resolve faces of senior persons as illustrated in the second row of Fig. 8.4(i) and the first rows of Fig. 8.5(i) as well as the child face in the last row of Fig. 8.4(i).

8.6.3 Quantitative Comparison with the SoA

We quantitatively measure the performance of all the methods on the entire test dataset by the average Peak Single-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) scores. Table 8.1 demonstrates that our method achieves superior performance in comparison to other methods, outperforming the second best with a large margin of **1.42** dB in PSNR.

As indicated in Tab. 8.1, after retraining VDSR and SRGAN with face images, they achieve higher PSNRs but still output inferior quantitative results compared with our results. TDAE [Yu and Porikli, 2017b] also employs multiple STNs to align LR face images and achieves the second best results. Note that TDAE employs three networks to super-resolve face images, which is much larger than our network. This also indicates that the ambiguity is significantly reduced by imposing attribute infor-

Table 8.4: Embedding attributes into different layers of $\mathcal{D}$

Layers	$\mathcal{D}_1$	$\mathcal{D}_2$	$\mathcal{D}_3$	$\mathcal{D}_4$
PSNR	21.59	21.76	**21.82**	21.63
SSIM	0.62	0.62	**0.62**	0.61

mation into the super-resolution procedure rather than by increasing the capacity of a neural network. Therefore, our method is able to achieve better quantitative results.

8.7 Discussions

8.7.1 Attribute Manipulation in Super-Resolution

Given an LR face image, previous deep neural network based face hallucination methods [Yu and Porikli, 2016, 2017b; Zhu et al., 2016b] only produce a certain HR face image. There is no freedom for those methods to fine-tune the final results. In contrast, our method can output different super-resolved results by adjusting the attribute vectors. As shown in Fig. 8.6, by changing the gender attribute we can hallucinate face images either from male to female or from female to male. Our method can manipulate the age of the upsampled faces, *i.e.*, more wrinkles and age spots, by changing the age attribute, as seen in Fig. 8.6(b). Because gender and age information may become ambiguous in LR face images, combining that semantic information in super-resolution can produce more accurate results. In addition, after obtaining super-resolved faces, our method is still able to post-edit the upsampled facial details in accordance with the desired attributes. For instance, our method removes the eye lines and shadows in Fig. 8.6(c), makes noses bigger in Fig. 8.6(d), removes and adds beard in Fig. 8.6(e), opens and closes eyes in Fig. 8.6(f), makes eyebrows bushy in Fig. 8.6(g), makes lips bigger in Fig. 8.6(h) as well as opens and closes mouths in Fig. 8.6(i) by manipulating the corresponding attribute vectors. Therefore, infusing semantic information into LR face images significantly increases the flexibility of our method.

To demonstrate our upsampling network is able to embed attributes into the upsampled HR faces successfully, we choose 9 different attributes, *i.e.*, gender, age, makeup, big nose, beard, open eyes, bushy eyebrows, big lips and open mouth, and train a attribute classifier for each attribute. Note that, some of our selected 18 attributes are coupled together, such as goatee and beard information, and some attributes may not be always consistent with human observation and are even hard to distinguish in upsampled faces in our experiments, such as eye bags. Therefore, we conduct the quantitative evaluations on the above 9 attributes as visible in Fig. 8.6 rather than all the selected attributes. By increasing and decreasing the corresponding attribute values, the true positive accuracies are changed accordingly, as illustrated in Tab. 8.2. This indicates that the attribute information has been successfully embedded in super-resolution.

Table 8.5: Quantitative evaluations of impact of different losses

Losses	$\mathcal{L}_{pix}$	$\mathcal{L}_{pix}$+$\mathcal{L}_{feat}$	$\mathcal{L}_{pix}$+$\mathcal{L}_{dis}$	Ours
PSNR	22.45	22.31	20.96	21.82
SSIM	0.66	0.65	0.57	0.62

8.7.2 Learn to Encode Attribute Vectors in Hallucination

Since our network directly accepts binary-value attributes, an option to improve the embedding might be using a shared CNN branch EN_s to encode attribute vectors. In the training stage, the encoding branch EN_s will be updated as well in order to embed attributes into the upsampling network. Because the output of EN_s, *i.e.*, the embedded attribute vector, is the input of both the upsampling network and the discriminative network, the ℓ_2 and perceptual losses from the upsampling network $\mathcal{U}$ and the discriminative loss from the discriminative network $\mathcal{D}$ are used to update EN_s. Therefore, although the upsampling network and the discriminative network are updated alternatingly, EN_s is updated in every iteration.

In training our discriminative network, the discriminative labels for the faces upsampled by $\mathcal{U}$ are set to 0 regardless of the attribute information, the labels for real faces with matched attributes are set to 1, and the labels for real faces with mismatched attributes are set to 0. Different from the previous training protocol [Yu and Porikli, 2016, 2017b], the discriminative loss is not only used to update the discriminative network but also employed to update the embedding branch EN_s. We only use one binary cross-entropy loss to update the discriminative network $\mathcal{D}$, but the training errors of $\mathcal{D}$ may come from either the face images or the mismatched attributes. Since the binary cross-entropy loss is not able to distinguish whether the faces are hallucinated or the attributes does not match the faces, it may cause ambiguity in the procedure of backpropagation.

On the other hand, in training our upsampling network, only the upsampled faces with their corresponding ground-truth attributes are fed into the discriminative network and the discriminative labels are set to 1. Note that, in training $\mathcal{D}$, the discriminative labels for super-resolved faces with their attributes should be 1 while in training $\mathcal{U}$, the labels are set to 0. Similar to previous works [Yu and Porikli, 2016, 2017a,b], the discriminative loss should be only used to update the upsampling network to make the super-resolved faces realistic, but here it is also used to update the encoding network EN_s. Thus, it is difficult for EN_s to learn a consistent encoder due to the contradicted discriminative labels in training $\mathcal{D}$ and $\mathcal{U}$. Therefore, the super-resolution performance using EN_s decreases 1.79 dB as indicated in Tab. 8.3 and the hallucinated faces suffer from obvious artifacts, as seen in Fig. 8.7(c). Therefore, we directly feed a binary-value attribute vector into our upsampling and discriminative network.

(a) (b) (c) (d) (e) (f)

Figure 8.7: Discussions on the variants of our network. (a) 16×16 LR input images. (b) 128×128 HR ground-truth images. (c) Result of using a shared CNN branch EN_s to encode attributes in super-resolution. (d) Result of using all neutral attributes. (e) Result without embedding attribute information. (f) Our result.

8.7.3 Performance with/without Autoencoder

As shown in Fig. 8.3(c), we demonstrate that it is not suitable to concatenate high-level semantic information with low-level image pixels directly. Specifically, we remove the autoencoder, replicate the attribute vector to the image size, and then concatenate the replicated attributes with the input LR image. In this way, all semantic labels will be applied to the whole images by the low-level convolutional filters. However, low-level filters are mainly responsible to extract image edges or corners [Zeiler and Fergus, 2014]. It is unsuitable to employ low-level filters to fuse high-level semantic information and low-level visual information. This is also verified by the quantitative result, donated as woAE, in Tab. 8.3.

On the contrary, we first encode the LR input faces by an encoder and then fuse the high-level semantic information, *i.e.*, attribute vectors, with the high-level feature maps extracted by the encoder. In this manner, the attribute labels are better associated with the feature maps qualitatively and quantitatively, as shown in Fig 8.3(h) and Tab. 8.3.

8.7.4 Performance with/without Skip-Connections

As shown in Fig. 8.2, we also employ skip-connections to pass low-frequency components of LR inputs to the decoder. In this fashion, we only focus on embedding the supplementary attributes into high-frequency facial details as well as preserve spatial information of LR input faces. Here, the low-frequency components are not strict low-frequency components of LR faces but relatively low-frequency compared to the components in the residual branch, *i.e.*, high-frequency components. Without using skip-connections, the network will fuse the facial attributes with all the frequency

components of LR faces. As seen in Fig. 8.3(d), the hallucinated faces suffer from obvious artifacts at the smooth regions after removing the skip-connections. Therefore, the attribute information should be fused into high-frequency components of LR faces rather than low-frequency ones. We also demonstrate the quantitative result without using the skip-connections, denoted as noSkip, in Tab. 8.3. As indicated in Tab. 8.3, with the help of the skip-connections, our super-resolution performance increases 0.60 dB in PSNR.

8.7.5 Performance with Inaccurate Attributes

When super-resolving very low-resolution face images, we may not always obtain all the 18 ground-truth attributes. Therefore, we may use inaccurate attribute information in face hallucination. In this case, we set undetermined attributes to 0.5 as neutral attributes in super-resolution because an attribute is set either 1 or 0 in training. In an extreme case, we do not know any information about attributes. Hence, we use the neutral value for all the attributes in super-resolution, marked as inAttr, and the quantitative result is shown in Tab. 8.3. Figure 8.7(d) also illustrates that our network can still generate high-quality results with inaccurate attributes.

8.7.6 Performance with/without Attribute Embedding

To demonstrate the influence of embedding attributes in face hallucination, we remove the branches of feeding attributes into $\mathcal{U}$ and $\mathcal{D}$ for comparisons, and denote this variant as woAttr. As shown in Fig. 8.7(e), the final results upsampled by woAttr suffer from gender reversal and expression changes. The average PSNR without embedding attributes decreases 0.18 dB, as indicated in Tab. 8.3. Furthermore, we also employ two pretrained attribute classifiers, *i.e.*, gender and age, to recognize the attributes recovered by our network and woAttr. For the age classification results, the error rate of our proposed network is 0 while the error rate of woAttr is 23.4%. For the gender classification results, the error rate of our proposed network is 0 while the error rate of woAttr is 6%. These experiments demonstrate that our method effectively reduces ambiguity in face hallucination by embedding supplementary attributes.

8.7.7 Impact of Embedding Layers in $\mathcal{D}$

As mentioned in Sec. 8.5.2, we embed attribute vectors into the third layer of the discriminative network. Here, we also demonstrate the quantitative results of embedding attributes into different layers of the discriminative network, (*i.e.*, 1st, 2nd, 3rd and 4th convolutional layers). As reported in our previous work [Yu and Porikli, 2017a], overly smoothed upsampled results tend to achieve higher PSNR but their visual quality is inferior. Therefore, we compare the quantitative results when these variants generate similar visual quality results. As shown in Tab. 8.4, we achieve the best performance when embedding attribute vectors into the third layer of D.

8.7.8 Impact of Different Losses

As seen in Fig. 8.3, we only show the impact of different losses on the visual results. In Tab. 8.5, we also show the quantitative results of our network trained by using different losses. When only employing the pixel-wise ℓ_2 loss, the average PSNR is higher but the visual results suffer from severe blurriness, as shown in Fig. 8.3(e). To avoid generating overly smoothed results, the feature-wise ℓ_2 loss is used in training the network. Due to the lack of the guidance of high-level semantic information in super-resolution, the network trained by using the pixel-wise and feature-wise losses still suffers from notorious ambiguity, such as gender reversal or facial rejuvenation. Using the discriminative loss $\mathcal{L}_{dis}$ and the pixel-wise ℓ_2 loss is able to embed the attribute information in the upsampled face images, but the facial characteristics may not be fully captured. Thus, the upsampling network generates ringing artifacts to mimic facial details, as shown in Fig. 8.3(g). By employing these three losses altogether, our network is able to achieve the best visual quality. Similar to the phenomenon mentioned in our previous work [Yu and Porikli, 2016], using the discriminative loss is a trade-off between the quantitative performance and the visual quality. Therefore, we set the weight for the discriminative loss to 0.001.

8.8 Conclusions

We introduced an attribute embedded discriminative network to super-resolve very low-resolution (16×16 pixels) unaligned face images by a large magnification factor 8× in an end-to-end fashion. With the help of the conditional discriminative network, our network successfully embeds facial attribute information into the upsampling network to reduce the inherit ambiguity in super-resolution. After training, our network is not only able to super-resolve LR faces but also fine-tune the upsampled results by adjusting the attribute information. In this manner, our network can generate HR face images much closer to their corresponding ground-truth ones, thus achieving superior face hallucination performance.

Chapter 9

Can We See More? Joint Frontalization and Hallucination of Unaligned Tiny Faces

9.1 Foreword

Previous chapters mainly focus on super-resolving high-resolution face images from the low-resolution inputs. When the face images are in large poses, such as profile views, there is less information available for human observation and computer analysis compared to the frontal ones. Thus, profile faces bring difficulties to the state-of-the-art face recognition systems or even human perception. Benefiting from the great power of deep neural networks, some works are developed to frontalize high-resolution side-view faces. In this fashion, they can provide more information for human observation as well as computer processing. Inspired by this idea, we also attempt to super-resolve low-resolution faces while frontalizing the upsampled face images simultaneously, thus providing more information for human and machine perception.

This chapter has been submitted to *IEEE Transactions on Pattern Analysis and Machine Intelligence* as a journal paper: Xin Yu, Basura Fernando, Fatih Porikli, Richard Hartley: Hallucinating Unaligned Face Images by Multiscale Transformative Discriminative Networks.

9.2 Abstract

In popular TV programs (such as CSI), a very low-resolution face image of a person, who is not even looking at the camera in many cases, is digitally super-resolved to a degree that suddenly the person's identity is made visible and recognizable. Of course, we suspect that this is merely a cinematographic special effect and such a magical transformation of a single image is not technically possible. Or, is it? In this paper, we push the boundaries of super-resolving (hallucinating to be more accurate) a tiny, non-frontal face image to understand how much of this is possible by leverag-

ing the availability of large datasets and deep networks. To this end, we introduce a novel Transformative Adversarial Neural Network (TANN) to jointly frontalize very-low resolution (*i.e.* 16×16 pixels) out-of-plane rotated face images (including profile views) and aggressively super-resolve them (8×), regardless of their original poses and without using any 3D information. TANN is composed of two components: a transformative upsampling network which embodies encoding, spatial transformation and deconvolutional layers, and a discriminative network that enforces the generated high-resolution frontal faces to lie on the same manifold as real frontal face images. We evaluate our method on a large set of synthesized non-frontal face images to assess its reconstruction performance. Extensive experiments demonstrate that TANN generates both qualitatively and quantitatively superior results achieving over 4 dB improvement over the state-of-the-art.

9.3 Introduction

Recovering high-resolution (HR) face images from their low-resolution (LR) counterparts, known as face hallucination, has received significant attention in recent years. Existing face hallucination methods mainly focus on super-resolving nearly frontal faces, which provide critical perceptual information for the human visual system [Hassner et al., 2015]. However, in most cases, LR faces may not necessarily be frontal. Super-resolving such non-frontal LR faces requires either frontalizing them first and then applying existing face hallucination techniques, or super-solving first (which highly depends on an available pose-specific exemplar dataset) and then frontalizing. Nevertheless, both of these options are naturally very challenging.

Conventional and emerging face frontalization methods [Blanz and Vetter, 1999; Yang et al., 2011; Hassner, 2013; Taigman et al., 2014; Sagonas et al., 2015; Hassner et al., 2015; Thies et al., 2016] often rely on facial landmarks for warping 2D face images onto 3D models, and thus require the input images to have a sufficient resolution where such landmarks are detectable. This renders them ineffective for tiny face images. Without a proper frontalization, directly employing face hallucination methods [Baker and Kanade, 2000, 2002; Liu et al., 2001; Wang and Tang, 2005; Liu et al., 2007; Hennings-Yeomans et al., 2008; Ma et al., 2010; Yang et al., 2010; Li et al., 2014; Kolouri and Rohde, 2015; Wang et al., 2014; Yu and Porikli, 2016, 2018] may cause severe artifacts due to large pose variations and misalignments. As shown in Fig. 9.1 and Fig. 9.3, for very low-resolution non-frontal face images, applying either face frontalization followed by hallucination, or hallucination followed by frontalization produces degraded results.

In this paper, we aim to *jointly* frontalize and hallucinate a given input face image so as to avoid the artifacts produced by either of these tasks individually. To do so, we present a new Transformative Adversarial Neural Network (TANN) that automatically frontalizes the LR faces while hallucinating the frontalized LR feature maps by an upscaling factor of 8× in an end-to-end fashion. Considering that an LR input face may undergo large pose variations and misalignments as seen in Fig. 9.1, our

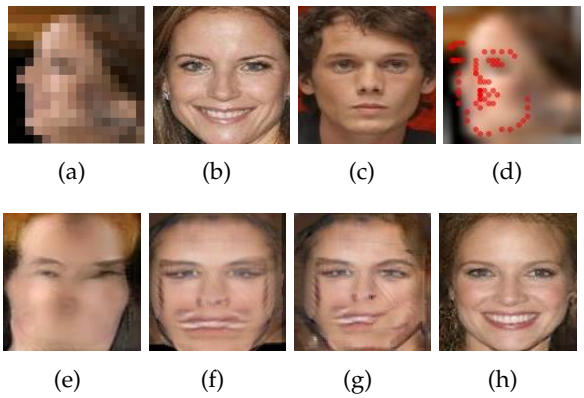


Figure 9.1: Comparison with the combination of face hallucination [Yu and Porikli, 2017b] and frontalization [Hassner et al., 2015] methods. (a) 16×16 LR non-frontal input image. (b) 128×128 HR original frontal image (not available in training). (c) The best possible match to the given LR image in the dataset after compensating for in-plane rotations by STN_0 [Jaderberg et al., 2015]. (d) Detected landmarks by the method of Zhu and Ramanan [2012] after bicubic upsampling. (e) Result obtained by applying [Hassner et al., 2015] first and then [Yu and Porikli, 2017b]. In [Yu and Porikli, 2017b], the first decoder and encoder are used to reduce image noise. Hereby, we only use the second decoder of Yu and Porikli [2017b] for super-resolving LR faces. (f) Result obtained by applying [Yu and Porikli, 2017b] first and then [Hassner et al., 2015]. (g) Image generated by [Yu and Porikli, 2017b], which is retrained with LR non-frontal and HR frontal face images. (h) Our result.

motivation is to force a non-frontal LR face to share the same latent representation of its corresponding frontal LR face and then super-resolve the latent representation. Thus, we first design a transformative subnetwork to encode a non-frontal LR face into a latent representation, where the representation of the input non-frontal LR face is forced to be similar to the latent representation of its frontal counterpart in the latent subspace. Then, we pass the latent representations, *i.e.*, the frontalized LR feature maps, through a subnetwork that is composed of deconvolutional and spatial transformer layers [Jaderberg et al., 2015], whose goal is to generate HR outputs. Inspired by previous works [Goodfellow et al., 2014; Denton et al., 2015; Yu and Porikli, 2016, 2017a; Xu et al., 2017], we choose to employ an adversarial network to make these HR outputs more closely resemble real human faces.

In order to train our network, we not only employ the traditional pixel-wise image appearance similarity and class-wise similarity constraints used in our previous works [Yu and Porikli, 2017b, 2016], but also develop a triplet loss to constrain the similarity of the latent representations between the input non-frontal faces and their ground-truth frontal LR ones. With the help of the proposed triplet loss, we are able to enforce that the representation of a side-view face to be close to its corresponding frontal LR face and far from other LR frontal faces in the latent subspace. In this

manner, the upsampled frontalized HR faces are not only similar to their HR frontal counterparts but also distinguishable from other hallucinated faces. Furthermore, we exploit a feature-wise similarity constraint, known as perceptual loss [Johnson et al., 2016], to make the hallucinated facial characteristics similar to the ground-truth, thus improving the visual quality.

Although deep neural networks have given rise to major advances in many computer vision tasks, they require very large datasets to train millions of parameters in their models. In our case, the existing large-scale face datasets [Huang et al., 2007; Liu et al., 2015] do not provide a sufficient number of frontal and non-frontal face image pairs for training our TANN. To obtain a large corpus of frontal and non-frontal face image pairs for the goal of training our deep neural network, we construct a set of out-of-plane rotated images from available frontal faces mapped onto a 3D face model. We first map randomly chosen frontal images to a 3D model, and then render different views of the 3D face, similar to the work [Masi et al., 2016]. This allows us to have high-quality HR frontal faces as our ground-truth images. It is important to note that this step is only to construct the *training dataset*, as we do not use any 3D models in our network (neither in training, nor in testing). In our experiments, we use non-frontal faces whose 3D models are unknown to demonstrate that TANN can hallucinate and frontalize different views of any unaligned LR face beyond the poses it is exposed to in training.

Overall, our contributions can be summarized as follows.

- We introduce a new transformative adversarial neural network to simultaneously hallucinate (by an upscaling factor of $8\times$) and frontalize tiny (16×16 pixels) unaligned face images with pose variations up to $\pm75^\circ$.
- We propose a new triplet loss to encode non-frontal LR faces into a latent subspace without distorting the encoding of frontal LR ones. With the help of the proposed triplet loss, we can force non-frontal LR faces to be close to their ground-truth frontal ones while keeping away from other faces in the latent subspace.
- We perform the training of our network in an end-to-end fashion by incorporating the reconstruction, perceptual, discriminative and triplet loss terms. In order to train our network, we also provide a dataset of corresponding frontal and non-frontal view face image pairs, which will be made available on-line to the vision community at large.
- We achieve superior hallucination results and outperforms the state-of-the-art by a large margin of **4.0** dB PSNR. Our method eliminates the need for facial landmarks or 3D face models as it is agnostic to the underlying in-plane and out-of-plane pose variations and spatial deformations. In the testing phase, our method can successfully process faces that are imaged at views not seen during training.

To the best of our knowledge, our method is the first attempt to provide a unified framework for super-resolution and frontalization of unaligned very low-resolution

face images, reducing significantly the artifacts introduced by either strategy, when considered individually.

9.4 Related Work

Our work mainly focuses on two aspects: face frontalization and hallucination. We briefly review noteworthy face frontalization and hallucination works below.

Face Frontalization: Generating a frontal face from a single non-frontal face image is very challenging due to self-occlusions and various pose variations, and has received significant attention in computer vision. Seminal works date back to the 3D Morphable Model (3DMM) [Blanz and Vetter, 1999], where a face is represented by the shape and texture bases in PCA subspace. After obtaining the the shape and texture coefficients of an input face image, Blanz and Vetter [1999] render novel views of an input face. Driven by 3DMM, Yang et al. [2011] estimate 3D surface from face appearance and then synthesize new expressions of the given face. However, these methods require the input face images to be nearly frontal in order to estimate the shape and appearance coefficients of input faces in PCA subspace. Dovgard and Basri [2004] exploit the facial symmetry to estimate 3D geometry of the given faces and render frontal faces. Similarly, Hassner et al. [2015] use facial symmetry to render out-of-view facial regions. Some methods [Asthana et al., 2011; Hassner, 2013; Taigman et al., 2014; Masi et al., 2016; Zhu et al., 2015] attempt to reconstruct frontal views by mapping a 2D face image onto a 3D reference surface mesh after registering and normalizing the face image. Since they need to detect facial landmarks in the input images and establish correspondences of landmark points to 3D or 2D reference models, they require images in sufficiently high resolutions. Based on the fact that frontal faces have the minimum rank of all different poses, Sagonas et al. [2015] propose a statistical face frontalization method, but the appearance of their frontalized faces may not be consistent with the input faces.

Deep learning based face frontalization methods have been proposed recently as well [Zhu et al., 2014; Yim et al., 2015; Zhu et al., 2015; Tran et al., 2017b; Cole et al., 2017; Huang et al., 2017b; Yin et al., 2017]. Zhu et al. [2014] present a deep neural network to frontalize HR faces by exploiting the symmetry and similarity of facial components. Their method does not require estimation of a 3D model, but it cannot maintain appearance similarity between the frontalized and input faces either. Yim et al. [2015] develop a multi-task deep neural network to rotate faces, but their method outputs blurry frontal faces due to the aggressive downsampling operations in the encoder. Similarly, Cole et al. [2017] learn to generate facial landmarks and textures from features extracted by a face recognition network. Since Cole *et al.*warp input faces to the mean face geometry by using facial landmarks, the resolutions of their inputs need to be sufficiently large. Very recently, Huang et al. [2017b] employ two deep neural networks, *i.e.*, global and local networks, to frontalize faces. However, their local network needs to extract HR facial components for identity preservation and to align HR facial components to pre-defined positions, and thus

their method is not suitable for very LR unaligned non-frontal face images. Yin et al. [2017] combine 3DMM and a generative adversarial network to frontalize faces with arbitrary poses. They also need to localize facial landmarks when mapping the input faces to the 3DMM. Thus their method requires sufficient resolutions for input images. Tran et al. [2017a] present a convolutional neural network (CNN) to regress 3DMM shape and texture parameters to speed up the optimization of 3DMM, but their method does not render frontalized faces which are similar to the input faces in terms of image intensity.

Face Hallucination: Face super-resolution (FSR), also known as face hallucination, aims at magnifying an LR image to its HR version and can be roughly grouped into three categories: holistic-based, part-based, and deep network based solutions.

Holistic-based methods attempt to super-resolve an entire HR face by using global face models, often learned by PCA. Wang and Tang [2005] establish a linear mapping between LR and HR face subspaces to super-resolve HR faces, while Liu et al. [2007] learn a global appearance model for upsampling LR inputs and employ a local nonparametric model to enhance the facial details. Kolouri and Rohde [2015] propose to morph an HR output from the aligned exemplar faces similar to LR inputs by the optimal transport and subspace learning techniques. Because holistic-based methods require LR inputs to be accurately aligned and to share the same pose and expression as HR references when learning global face models, they are very sensitive to misalignments and pose variations.

Instead of super-resolving entire faces, part-based methods upsample facial regions and thus can address various poses. They either use reference position patches, or employ facial components to restore the HR counterparts of LR inputs. For instance, Baker and Kanade [2002] reconstruct high-frequency details of aligned frontal face images by finding the best mapping between LR and HR patches. Similarly, Ma et al. [2010] employ position patches extracted from multiple aligned HR images to upsample aligned LR face images. Rather than reconstructing patches in the image domain, Yang et al. [2010] and Li et al. [2014] super-resolve HR image patches by employing sparse coding techniques to achieve better performance. Tappen and Liu [2012] apply SIFT flow [Liu et al., 2011] to align the facial parts of LR images and reconstruct HR facial details by warping the reference HR images, while Yang et al. [2013, 2017a] localize facial components in the LR images by a facial landmark detector and then reconstruct details from the similar HR reference components. Since these methods need to extract facial components in LR face images accurately, their performance degrades dramatically when the LR faces are tiny. We refer the readers to the paper [Wang et al., 2014] for a more comprehensive survey on face hallucination using traditional approaches.

As large-scale datasets become available, Zhou and Fan [2015] propose a convolutional neural network (CNN) to extract facial features and recover facial details from the extracted features. Yu and Porikli [2018] consolidate deconvolutional and convolutional layers for super-resolving LR face images, but they improve the visual quality by a post-processing technique, *i.e.*, an unsharp filter. The work presented in [Yu and Porikli, 2016] develops a discriminative generative network to super-

resolve aligned LR face images in an end-to-end fashion while Huang et al. [2017a] exploit wavelet coefficients learned by CNN to restore HR faces. In order to relax the requirement of face alignment, Yu and Porikli [2017a] embed multiple spatial transformer networks [Jaderberg et al., 2015] into the generative network of Yu and Porikli [2016]. Their follow-up work [Yu and Porikli, 2017b] employs a decoder-encoder-decoder structure to super-resolve noisy LR faces while suppressing image noise. Xu et al. [2017] employ the generative adversarial framework [Goodfellow et al., 2014] as well as a multi-class adversarial loss to upsample blurry and LR face and text images. Dahl et al. [2017] exploit the framework of PixelCNN [Van Den Oord et al., 2016], known as an autoregressive generative model, to hallucinate very low-resolution face images. Towards the same goal, Zhu et al. [2016b] use a cascade bi-network to upsample very low-resolution and unaligned faces, of which one is used to super-resolve low-frequency components of face images and the other is employed to hallucinate high-frequency facial details. Since these deep learning based methods do not take out-of-plane rotations of faces into account and are restricted to small pose variations, (*i.e.*within $\pm 30^{\circ}$), they may fail to super-resolve LR faces with large pose variations.

Due to the above limitations, simply cascading face hallucination and frontalization methods is not an acceptable solution for our problem.

9.5 Proposed Method: TANN

Our network has two components: (i) a transformative upsampling network, which transforms different poses to the frontal one and also super-resolves the frontalized LR feature maps; and (ii) a discriminative network, which forces the generated HR frontal faces to lie on the manifold of authentic HR face images. Figure 9.2 illustrates the overall architecture of TANN.

In the training phase, the entire network is trained in an end-to-end fashion to compensate for possible artifacts induced by any of the frontalization and hallucination tasks. As shown in Fig. 9.3(i), when we train the upsampling network separately, *i.e.*, generating frontalized LR faces as intermediate results, the transformer subnetwork may suffer from the loss of information contained in its feature maps because it is enforced to output 3 channel LR faces as its objective function rather than 32 channel feature maps. This may lead to accumulated errors and obvious deviations in the output of the upsampling subnetwork due to the incorrect input images for upsampling. Thus, feeding 32 feature maps directly to the upsampling network is a better choice.

9.5.1 Transformative Upsampling Network (TUN)

In Fig. 9.2, our transformative upsampling network is shown (red box). TUN is composed of two parts: a transformer subnetwork and an upsampling subnetwork. The transformer part (purple box) aims at encoding non-frontal LR faces into latent representations which are close to the latent representations of their corresponding frontal

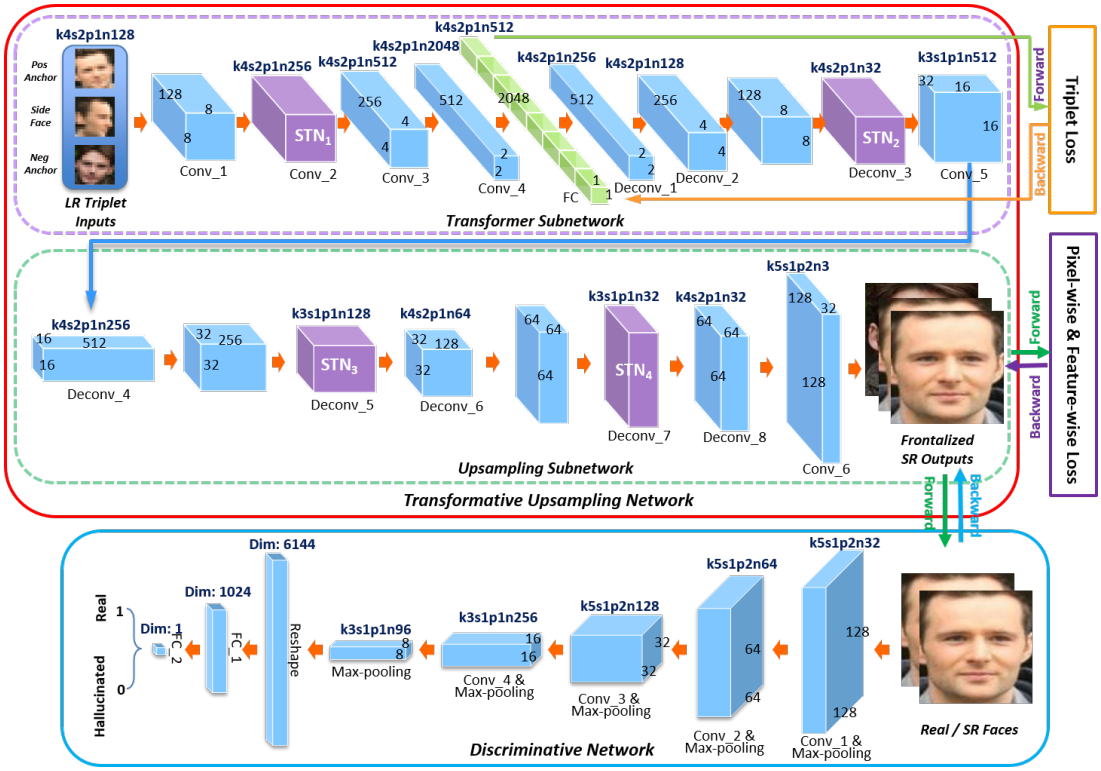


Figure 9.2: TANN consists of two parts: a transformative upsampling network (red box) and a discriminative network (blue box).

LR ones. By doing so, we can achieve the latent codes of frontalized LR faces. Our transformer subnetwork is constructed by convolutional layers, a fully-connected layer, deconvolutional layers and spatial transformer layers. Since the input LR faces undergo in-plane rotations, translations and scale changes, multiple spatial transformer networks (STN) [Jaderberg et al., 2015] are embedded as intermediate layers to compensate for such affine transformations. Moreover, because STNs learn 2D affine warps rather than out-of-plane rotations, they cannot recover self-occluded parts of faces. To solve this problem, our intuition is that we can project different views of a face into a subspace, where their encoded representations are enforced to lie close to the representations of their corresponding frontal one. Therefore, we incorporate a fully-connected layer to encode the feature maps of LR profile faces as well as design a triplet loss to force the similarity between the representations of LR profile and frontal ones.

To illustrate the effectiveness of the transformer subnetwork, we change the channel number of its output layer to 3, and use LR frontal faces as ground-truth images to train this subnetwork. As shown in Fig. 9.3(j) and Fig. 9.4(d), it can successfully generate an LR frontal face image. Note that, when training our TANN, we do not employ LR frontal faces as supervision to prevent the aforementioned drift issue.

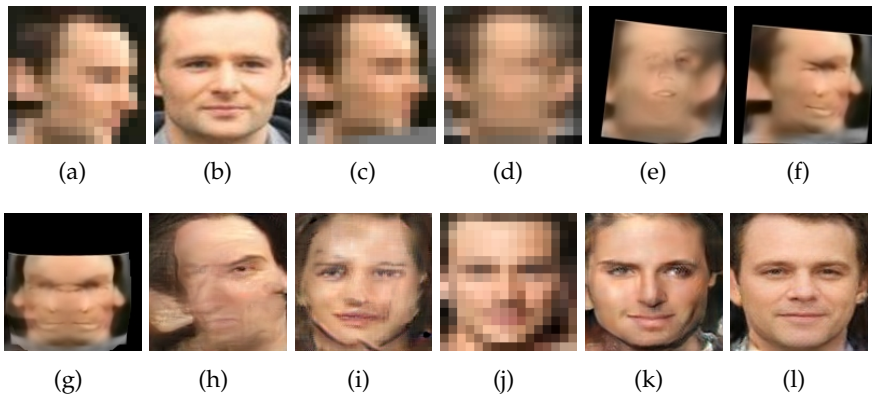


Figure 9.3: Artifacts caused by the state-of-the-art face frontalization and hallucination methods. (a) The input 16×16 LR image. (b) The original 128×128 HR frontal image. (c) The aligned upright version of (a) by STN_0. (d) Frontalized result of (c) using [Hassner et al., 2015]. Note that, we first upsample (c) by bicubic interpolation, then apply [Hassner et al., 2015], and downsample the frontalized result. (e) HR image after applying [Zhu et al., 2016b] to (d). (f) HR image after applying [Zhu et al., 2016b] to (c) directly. (g) The frontalized version of (f) by [Hassner et al., 2015]. (h) The result of applying [Yu and Porikli, 2017b] to (a). (i) The result of TANN without the transformer subnetwork, which is similar to the upsampling network [Yu and Porikli, 2017b], retrained with LR non-frontal and HR frontal faces. (j) The aligned and frontalized LR face by our transformer subnetwork. Note that, in our end-to-end trained TANN, the output of the transformer network is a set of feature maps not an image. (k) The hallucinated result of (j) by our upsampling subnetwork (here, we retrained the upsampling network). (l) Our final result.

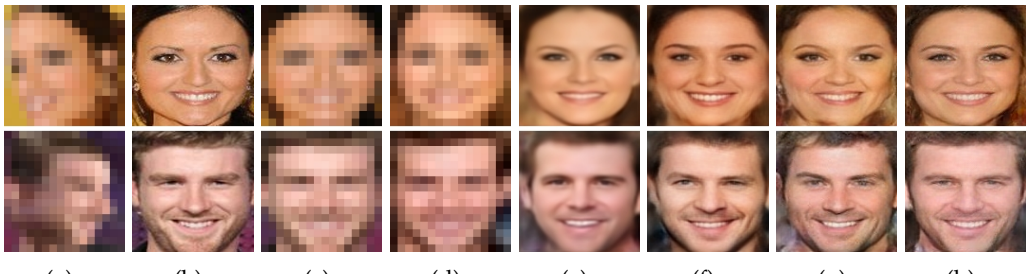


Figure 9.4: Illustrations of influence of different losses. (a) The input 16×16 LR images. (b) The original 128×128 HR frontal images. (c) The downsampled version of (b). (d) The frontalized LR faces by our transformer subnetwork. (e) The upsampling results only using pixel-wise loss. (f) The upsampling results using the pixel-wise and perceptual losses. (g) The upsampling results without using the triplet loss. (h) Our final results.

After obtaining the feature maps of LR frontal faces generated by the transformer subnetwork, we apply an upsampling subnetwork (green box in Fig. 9.2) to hallucinate the high-frequency facial details of frontal faces. Because the resolution of LR input images is very low, STNs in our transformer subnetwork may not align LR faces accurately. The LR feature maps generated by the transformer network may still contain misalignments. We employ the upsampling structure used in our previous works [Yu and Porikli, 2017a,b] for further alignment and super-resolution.

As shown in Fig. 9.3(h), simply applying the method of Yu and Porikli [2017b] to LR profile faces cannot provide high-quality HR frontal face images. This manifests that upsampling LR non-frontal faces with large pose variations is more difficult compared to LR frontal faces and also indicates the necessity of our transformer subnetwork. Since the mapping between common LR patterns and HR facial details can be easily learned from frontal faces, we frontalize LR inputs first and then hallucinate them.

9.5.2 Discriminative Network

As demonstrated in our previous works [Yu and Porikli, 2016, 2017a,b], only using Euclidean distance (pixel-wise ℓ_2 loss) between the upsampled faces and the ground-truth HR faces tends to generate over-smoothed results. Therefore, a class-specific discriminative objective is also incorporated into our TUN, aiming to force the hallucinated HR face images to lie on the same manifold of real frontal face images.

As shown in Fig. 9.2 (blue box), the discriminative network consists of convolutional layers, max-pooling layers, dropout layers, and fully-connected layers. It is designed to determine whether an image is sampled from real face images or the hallucinated ones. The discriminative loss, also known as adversarial loss, will be back-propagated to update the parameters of TUN as well. With the help of the adversarial loss, we can generate more realistic HR frontal faces. Figure 9.4 illustrates the impact of the adversarial loss on the final results.

9.5.3 Training Details of TANN

We construct LR profile and HR frontal ground-truth face image pairs $\{l_i, h_i\}$ for our training purpose, where h_i represents the aligned frontal HR face images (only eyes are aligned), and l_i is the synthesized LR side-view face images from h_i. For each HR frontal face h_i, we generate five different views, *i.e.*$\{0°, \pm 40°, \pm 75°\}$, to construct LR/HR training pairs. Using these five distinct poses is a trade-off between a sufficient coverage of pose variations and the reasonable size of the training dataset and also suggested in [Masi et al., 2016]. More details are provided in Sec. 9.6.

In training our TANN, we not only enforce the conventional pixel-wise intensity similarity, known as pixel-wise ℓ_2 loss, but also the feature-wise similarity, known as perceptual loss [Johnson et al., 2016], to obtain high-quality results. Similar to [Yu and Porikli, 2016, 2017a], the adversarial loss is also employed to attain visually appealing frontalized HR face images. As mentioned in Sec. 9.5.1, we also develop

a triplet loss to force the representations of LR profile faces to be similar to the representations of their frontal faces. In this manner, we can frontalize LR profile faces without degrading super-resolution of frontal ones.

Pixel-wise intensity similarity loss: We constrain the generated HR frontalized face $\hat{h}_i$ to be similar to its ground-truth frontal counterpart h_i in terms of image intensities. Thus we employ a pixel-wise ℓ_2 regression loss $\mathcal{L}_{pix}$ to impose the appearance similarity constraint, expressed as:

$$\begin{aligned}\mathcal{L}_{pix} &= \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\hat{h}_i - h_i\|_F^2 \\ &= \mathbb{E}_{(l_i,h_i)\sim p(l,h)}\|\mathcal{T}_t(l_i) - h_i\|_F^2,\end{aligned} \tag{9.1}$$

where t and $\mathcal{T}$ are the parameters and the output of TUN, $p(\hat{h},h)$ represents the joint distribution of the frontalized HR faces and their corresponding frontal HR ground-truths, and $p(l,h)$ indicates the joint distribution of the LR and HR face images in the training dataset.

Feature-wise similarity loss: As mention in [Yu and Porikli, 2016], pixel-wise ℓ_2 loss leads to over-smoothed super-resolved results. Here, we employ a feature-wise similarity loss, known as perceptual loss [Johnson et al., 2016], to constrain the super-resolved HR faces to share the same facial details as their ground-truth counterparts, thus attaining high-quality results with rich facial details. The perceptual loss $\mathcal{L}_{feat}$ measures Euclidean distance between the feature maps of HR frontalized and ground-truth faces extracted by a deep neural network, written as:

$$\begin{aligned}\mathcal{L}_{feat} &= \mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\|\Phi(\hat{h}_i) - \Phi(h_i)\|_F^2 \\ &= \mathbb{E}_{(l_i,h_i)\sim p(l,h)}\|\Phi(\mathcal{T}_t(l_i)) - \Phi(h_i)\|_F^2,\end{aligned} \tag{9.2}$$

where $\Phi(\cdot)$ denotes feature maps extracted by the ReLU32 layer in VGG-19 [Simonyan and Zisserman, 2014], which gives good empirical performance in our experiments.

Adversarial loss: In order to achieve visually appealing results, we infuse class-specific discriminative information into TUN by exploiting a discriminative network, similar to our previous works [Yu and Porikli, 2016, 2017a,b]. Our goal is to make the discriminative network fail to distinguish generated faces from real ones. In this manner, we enforce the super-resolved HR frontal faces to lie on the manifold of real frontal HR face images. Therefore, the discriminative network is used to categorize real HR frontal faces and generated ones, and thus its objective function is expressed as:

$$\begin{aligned}\mathcal{L}_{\mathcal{D}} &= -\mathbb{E}_{(\hat{h}_i,h_i)\sim p(\hat{h},h)}\left[\log \mathcal{D}_d(h_i) + \log(1 - \mathcal{D}_d(\hat{h}_i))\right] \\ &= -\mathbb{E}_{h_i\sim p(h)}\log \mathcal{D}_d(h_i) - \mathbb{E}_{\hat{h}_i\sim p(\hat{h})}\log(1 - \mathcal{D}_d(\hat{h}_i)) \\ &= -\mathbb{E}_{h_i\sim p(h)}\log \mathcal{D}_d(h_i) - \mathbb{E}_{l_i\sim p(l)}\log(1 - \mathcal{D}_d(\mathcal{T}(l_i))),\end{aligned} \tag{9.3}$$

where d represents the parameters of the discriminative network, $p(l)$, $p(h)$ and $p(\hat{h})$

indicate the distributions of the LR, HR ground-truth frontal and the generated faces respectively, and $\mathcal{D}_d(h_i)$ and $\mathcal{D}_d(\hat{h}_i)$ are the outputs of the discriminative network. To make the discriminative network distinguish hallucinated faces from real ones, we minimize the loss $\mathcal{L}_{\mathcal{D}}(d)$ and update the parameters d.

Meanwhile, our TUN aims to fool the discriminative network. Therefore, the adversarial loss for our TUN is:

$$\begin{aligned}\mathcal{L}_{\mathcal{T}} &= -\mathbb{E}_{\hat{h}_i \sim p(\hat{h})} \log(\mathcal{D}(\hat{h}_i)) \\ &= -\mathbb{E}_{l_i \sim p(l)} \log(\mathcal{D}(\mathcal{T}_t(l_i))).\end{aligned} \tag{9.4}$$

Here, we minimize the loss $\mathcal{L}_{\mathcal{T}}(t)$ to update the parameters t. These two adversarial losses in Eqn. 9.3 and Eqn. 9.4 are employed to update our TUN and discriminative network respectively in an alternating fashion.

Triplet loss: In order to frontalize side view LR faces, we present a triplet loss to constrain the encoded LR faces to be close to the latent representations of their corresponding frontal ones and far away from other frontal faces in the latent subspace. Therefore, our proposed triplet loss is expressed as:

$$\mathcal{L}_{tri} = \mathbb{E}_{(l_i^+, l_i^-, l_i) \sim p(\mathcal{S})} \frac{\left[\|\mathcal{F}(l_i) - \mathcal{F}(l_i^+)\|_F^2 - \|\mathcal{F}(l_i) - \mathcal{F}(l_i^-)\|_F^2\right]_+}{\|\mathcal{F}(l_i)\|_F^2}, \tag{9.5}$$

where $\mathcal{F}(\cdot)$ indicates the encoded latent representation by the fully-connected layer in our transformer subnetwork, (l_i^+, l_i^-, l_i) represents a triplet sample from the set of all possible triplets $\mathcal{S}$ in the training set. l_i is an LR profile face, l_i^+, dubbed positive anchor, is the corresponding frontal LR face of l_i, and l_i^-, dubbed negative anchor, is any other frontal LR face. One example of the triplets is shown in Fig. 9.2. In addition, $[x]_+$ denotes the operator $\max\{x, 0\}$.

Since our network aims at super-resolving LR faces rather than clustering faces, it should not distort the mapping between LR and HR frontal faces. Considering that positive and negative anchors are LR frontal faces, updating the gradients with respect to the representations of the positive and negative anchors will distort the mapping between LR and HR frontal faces. In other words, clustering triplets by adjusting the latent representations of positive and negative anchors would damage the end-to-end mapping between LR and HR frontal faces and thus leads to inferior super-resolution performance. Different from the triplet loss presented in [Schroff et al., 2015], we take positive and negative anchors as constant and only back-propagate gradients with respect to the latent codes of LR non-frontal faces. In this manner, we are able to upsample frontal faces without introducing distortions while forcing the encoded LR profile faces to be close to the representations of their frontal counterparts.

In our TANN, all the layers are differentiable and RMSprop [Hinton, 2012] is used to update the parameters t and d. We update the parameters d by minimizing the

adversarial loss $\mathcal{L}_{\mathcal{D}}$ as follows:

$$\begin{aligned} \Delta^{i+1} &= \gamma\Delta^{i} + (1-\gamma)(\frac{\partial\mathcal{L}_{\mathcal{D}}}{\partial d})^2, \\ d^{i+1} &= d^{i} - r\frac{\partial\mathcal{L}_{\mathcal{D}}}{\partial d}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}, \end{aligned} \tag{9.6}$$

where r and γ represent the learning rate and the decay rate respectively, i indicates the index of the iterations, Δ is an auxiliary variable, and ϵ is set to 10^{-8} to avoid division by zero. We employ multiple losses, *i.e.*, $\mathcal{L}_{pix}$, $\mathcal{L}_{feat}$, $\mathcal{L}_{\mathcal{T}}$ and $\mathcal{L}_{tri}$, to update our TUN and the object function is expressed as:

$$\mathcal{L}_{TUN} = \mathcal{L}_{pix} + \eta\mathcal{L}_{feat} + \lambda\mathcal{L}_{\mathcal{T}} + \mu\mathcal{L}_{tri}, \tag{9.7}$$

where η, λ and μ are the trade-off weights. Since we aim at super-resolving frontal HR faces rather than generating random faces, we put lower weights on the feature-wise, adversarial and triplet losses and set λ, η and μ to $10e^{-2}$, $10e^{-2}$ and $10e^{-4}$ respectively. Then, the parameters of TUN t are updated by the gradient descent as follows:

$$\begin{aligned} \Delta^{i+1} &= \gamma\Delta^{i} + (1-\gamma)(\frac{\partial\mathcal{L}_{TUN}}{\partial t})^2, \\ t^{i+1} &= t^{i} - r\frac{\partial\mathcal{L}_{TUN}}{\partial t}\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}. \end{aligned} \tag{9.8}$$

As the iteration progresses, the output faces will be more similar to real faces. Therefore, we gradually reduce the impact of the discriminative network by decreasing λ,

$$\lambda^{j} = \max\{\lambda\cdot 0.995^{j}, \lambda/2\}, \tag{9.9}$$

where j is the index of the epochs. Equation 9.9 not only increases the impact of the appearance similarity term but also preserves the class-specific discriminative information in the training phase.

9.5.4 Hallucinating Frontal HR from Non-frontal LR

The discriminative network is only employed in the training phase. In the testing phase, we feed an unaligned LR profile face image into the transformative upsampling network to obtain its upright and frontal HR version. Note that, only in the training stage, we need to feed the network with triplet samples due to employing the triplet loss. In the testing stage, our network is able to super-resolve and frontalize a single image. Since aligned HR frontal face images are employed as ground-truths, TUN will output aligned and frontalized HR faces directly. As a result, our method does not need to estimate the face orientations or align very low-resolution images beforehand, and provides an end-to-end and highly nonlinear mapping from an unaligned LR profile face image to its frontal HR version.

9.5.5 Implementation Details

The STN layers, as shown in Fig. 9.2, are built by convolutional and ReLU layers (Conv+ReLU), max-pooling layers with a stride 2 (MP2) and fully connected layers (FC). Since STN is mainly used for calibrating in-plane transformations, we employ the similarity transformation for alignment. Specifically, STN_1 and STN_2 share the same architecture and consist of Conv+ReLU (filter size: $20\times128\times3\times3$ with 1 pixel padding), MP2, Conv+ReLU ($20\times20\times3\times3$), FC+ReLU (from 400 to 20 dimensions), and FC (from 20 to 4 dimensions). STN_3 is composed of MP2, Conv+ReLU ($20\times256\times5\times5$), MP2, Conv+ReLU ($20\times20\times5\times5$), FC+ReLU (from 80 to 20 dimensions) and FC (from 20 to 4 dimensions). STN_4 is composed of MP2, Conv+ReLU ($128\times64\times5\times5$), MP2, Conv+ReLU ($20\times128\times5\times5$), MP2, Conv+ReLU ($20\times20\times3\times3$), FC+ReLU (from 120 to 20 dimensions) and FC (from 20 to 4 dimensions).

Similar to [Goodfellow et al., 2014; Radford et al., 2015], batch normalization [Ioffe and Szegedy, 2015] is employed after each convolution except the final output layer of TUN and dropout is applied to the feature maps in the discriminative network. In the experimental part, some algorithms may require alignment of LR inputs, *e.g.* [Ma et al., 2010]. Hence, we employ another network STN_0 to align the LR face images to the upright position, and STN_0 consists of Conv+ReLU ($128\times3\times3\times3$ with 1 pixel padding), MP2, Conv+ReLU ($20\times20\times3\times3$), MP2, FC+ReLU (from 180 to 20 dimensions), and FC (from 20 to 4 dimensions).

We also use a triplet pair $\{(l_i^+, l_i, l_i^-), (h_i, h_i, h_i^-)\}$ as a unit to construct our mini-batch in training, where h_i is the HR frontal face image corresponding to the LR profile face l_i and the LR frontal face l_i^+, and h_i^- is the HR frontal version of the LR frontal face l_i^-. The triplet pairs are not only designed to calculate the triplet loss but also compatible with the other losses. Therefore, our network can be trained in an end-to-end fashion.

The learning rate r is set to 0.001 and multiplied by 0.99 after each epoch, η is set to 0.01, and the decay rate is set to 0.01.

9.6 Synthesized Dataset

Training of a deep neural network requires a large number of samples to prevent models from overfitting to the training dataset. However, the publicly available large-scale face datasets [Huang et al., 2007; Liu et al., 2015] only provide faces in the wild but not frontal/non-frontal pairs. For the training purpose, we opt to generate a large set of synthesized LR non-frontal faces from HR frontal face images.

There are a number of alternative approaches available. For instance, Hassner et al. [2015] render 2D frontal faces from different side-view faces using a single 3D reference mesh. However, when the out-of-view face regions are large, these methods are prone to artifacts. Similarly, landmark detection algorithms may fail to localize facial landmarks accurately in large poses.

We adopt the idea of Masi et al. [2016] to generate different views from HR frontal ones. We use a single 3D face model to render HR out-of-plane rotated faces

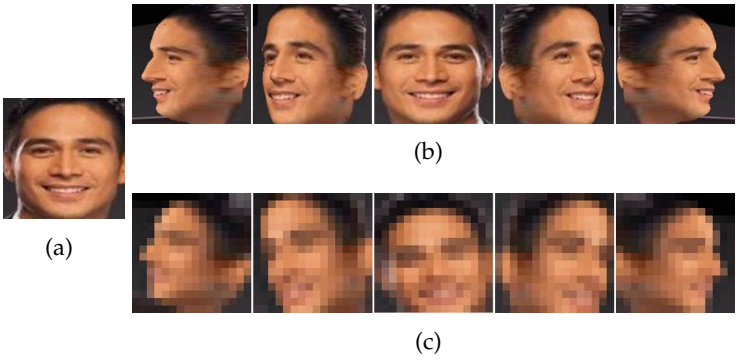


Figure 9.5: Illustration of the synthesized dataset. (a) Original frontal HR face image. (b) The generated views of (a). (c) Spatially transformed and downsampled version of (b).

while taking advantage of the mirror-symmetry for the positive and negative angles to produce five different views of faces, *i.e.*, $\{0°, \pm 40°, \pm 75°\}$. Specifically, we first randomly select 10K cropped frontal faces (within $\pm 5°$) from the CelebA [Liu et al., 2015], and resize them to 128×128 pixels. We use these images as our HR ground-truth faces h_i. Then we generate the non-frontal LR faces l_i by transforming and downsampling the reconstructed HR images down to 16×16 pixels. Therefore, we obtain 50K LR/HR face pairs for training and testing of our network. Figure 9.5 illustrates sample pairs $\{l_i, h_i\}$ generated from a single frontal face.

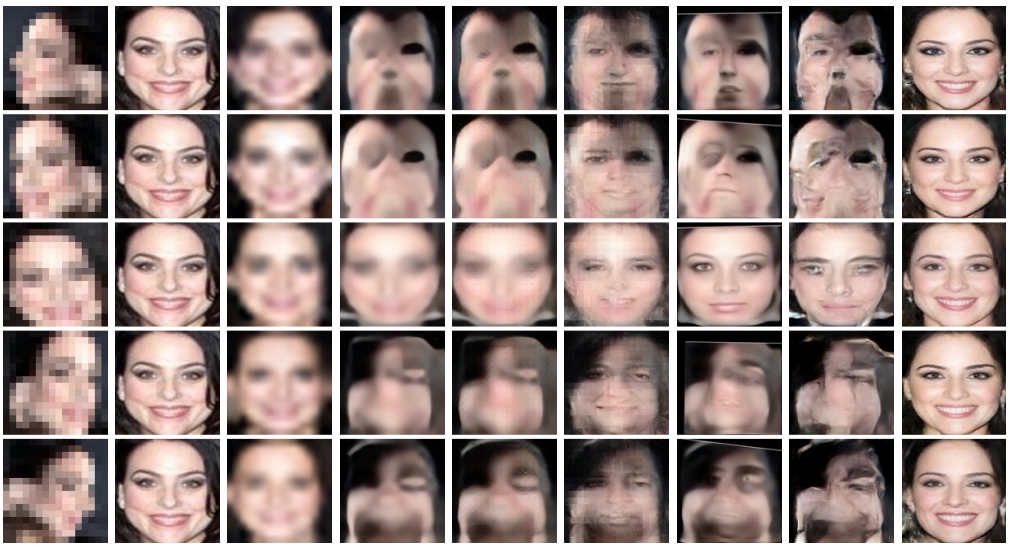
9.7 Experimental Evaluation

We compare our method with ten state-of-the-art methods qualitatively and quantitatively. As mentioned in Sec. 9.6, we assemble 50K LR/HR face pairs, and randomly choose 9K frontal face images for training (45K LR/HR pairs), and 1K faces for testing (5K LR/HR pairs). In training TANN, we randomly choose a side-view LR face, its corresponding frontal LR face and any other frontal LR face to construct an input triplet (l_i^+, l_i, l_i^-) as well as employ their corresponding HR ground-truth triplet (h_i, h_i, h_i^-) as supervision. In all cases, the training data and test data do **not** overlap. We use different ground-truth HR frontal faces in the training and testing phases.

9.7.1 Qualitative Comparisons with the SoA

Since Ma et al. [2010] require the input LR faces to be aligned uprightly, we train STN_0 to align the LR inputs to the upright position for a fair comparison. Note that, our method does not need any alignment or pose estimation in advance.

As illustrated in Fig. 9.6(c) and Fig. 9.7(c), different combinations of bicubic interpolation and the frontalization method [Hassner et al., 2015] cannot produce authentic frontal face details. Because of the low resolution of inputs, the method of Hassner et al. [2015] fails to detect facial landmarks and outputs erroneous frontalized faces



(a) (b) (c) (d) (e) (f) (g) (h) (i)

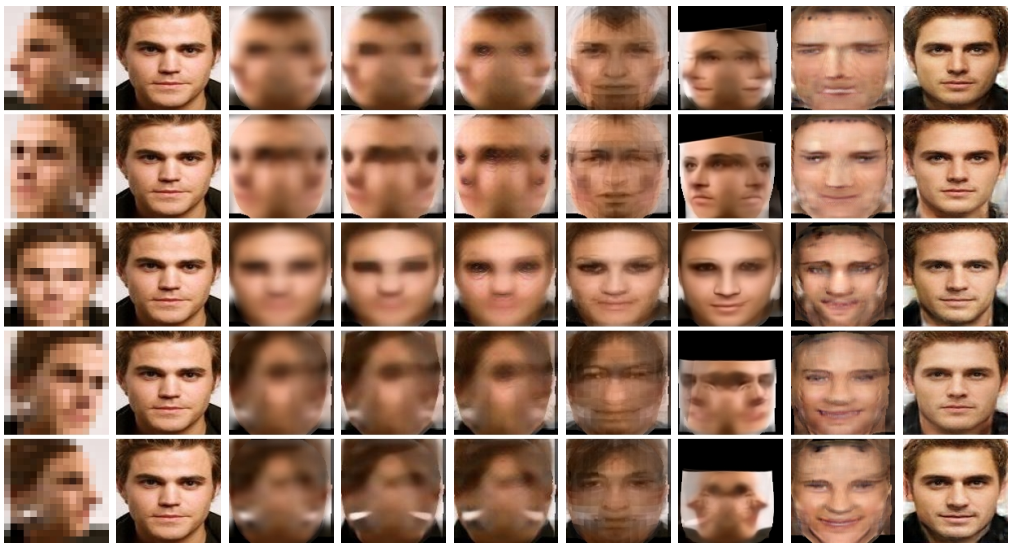
Figure 9.6: Results of the state-of-the-art methods for **frontalization followed by hallucination**. The input faces are first frontalized by [Hassner et al., 2015] and then hallucinated by different algorithms. Rows: +75°, +40°, 0°, -40°, and -75°. Columns: (a) Unaligned non-frontal LR inputs. (b) Original frontal HR images. (c) [Hassner et al., 2015] + bicubic interpolation. (d) [Hassner et al., 2015] + [Kim et al., 2016a]. (e) [Hassner et al., 2015] + [Ledig et al., 2017]. (f) [Hassner et al., 2015] + [Ma et al., 2010]. (g) [Hassner et al., 2015] + [Zhu et al., 2016b]. (h) [Hassner et al., 2015] + [Yu and Porikli, 2017b]. (i) Our method. Notice that, TANN does not need or use the method of Hassner et al. [2015].

while bicubic interpolation is handicapped to generate necessary high-frequency facial details.

Kim et al. [2016a] propose a very deep CNN based general purpose super-resolution (SR) method, known as VDSR. Since VDSR is trained on natural image patches and does not provide an upscaling factor of 8×, we retrain VDSR with face patches by an upscaling factor of 8×. As shown in Fig. 9.6(d) and Fig. 9.7(d), VDSR fails to produce facial details and thus contaminates the outputs of Hassner et al. [2015] with ghosting artifacts.

Ledig et al. [2017] present a generic super-resolution method, dubbed SRGAN. SRGAN employs the framework of generative adversarial networks [Goodfellow et al., 2014; Radford et al., 2015] to enhance the visual quality and is trained by using not only a pixel-wise ℓ_2 loss but also an adversarial loss. Although SRGAN provides an upscaling factor of 8×, it fails to capture the entire face structure and produces ringing artifacts to mimic high-frequency facial details, which also brings difficulty for frontalization, as shown in Fig. 9.6(e) and Fig. 9.7(e).

Ma et al. [2010] super-resolve LR inputs by exploiting position patches, and re-



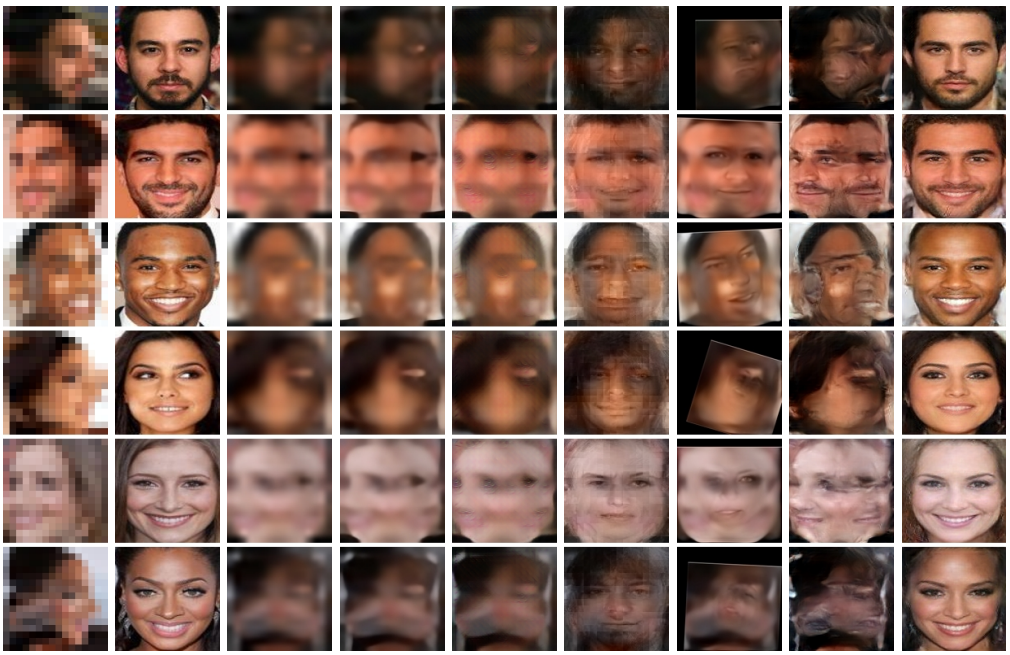
(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 9.7: Results of the state-of-the-art methods for **hallucination followed by frontalization** by [Hassner et al., 2015]. Columns: (a) Unaligned non-frontal LR inputs. (b) Original frontal HR images. (c) Bicubic interpolation + [Hassner et al., 2015]. (d) [Kim et al., 2016a] + [Hassner et al., 2015]. (e) [Ledig et al., 2017] + [Hassner et al., 2015]. (f) [Ma et al., 2010] + [Hassner et al., 2015]. (g) [Zhu et al., 2016b] + [Hassner et al., 2015]. (h) [Yu and Porikli, 2017b] + [Hassner et al., 2015]. (i) Our method.

quire the LR inputs to be precisely aligned with the exemplar training dataset. It spawns severe artifacts in the upsampled faces because of large pose variations that exist in the input LR images as visible in Fig. 9.7(f). Due to the faulty frontalization by Hassner et al. [2015], this method also produces distorted facial details, as shown in Fig. 9.6(f).

Zhu et al. [2016b] present a deep cascaded bi-network for face hallucination, called CBN, which first localizes facial landmarks and then aligns LR faces based on the localized landmarks. However, when the inputs undergo large pose variations, CBN cannot localize facial landmarks accurately, and thus causes severe artifacts as seen in Fig. 9.7(g). Figure 9.6(g) shows that CBN cannot hallucinate authentic HR faces from the incorrect frontalized LR faces either.

Yu and Porikli [2017b] propose a transformative discriminative autoencoder (T-DAE) as an extension to the method of Yu and Porikli [2016] to upsample unaligned and noisy LR face images. TDAE interweaves deconvolutional and STN layers to align and super-resolve LR faces while employing a discriminative network that forces the generative network to produce sharper results. However, TDAE can only hallucinate unaligned frontal faces rather than profile faces as demonstrated in Fig. 9.7(h) since it does not take out-of-plane rotations into account and the first



(a) (b) (c) (d) (e) (f) (g) (h) (i)

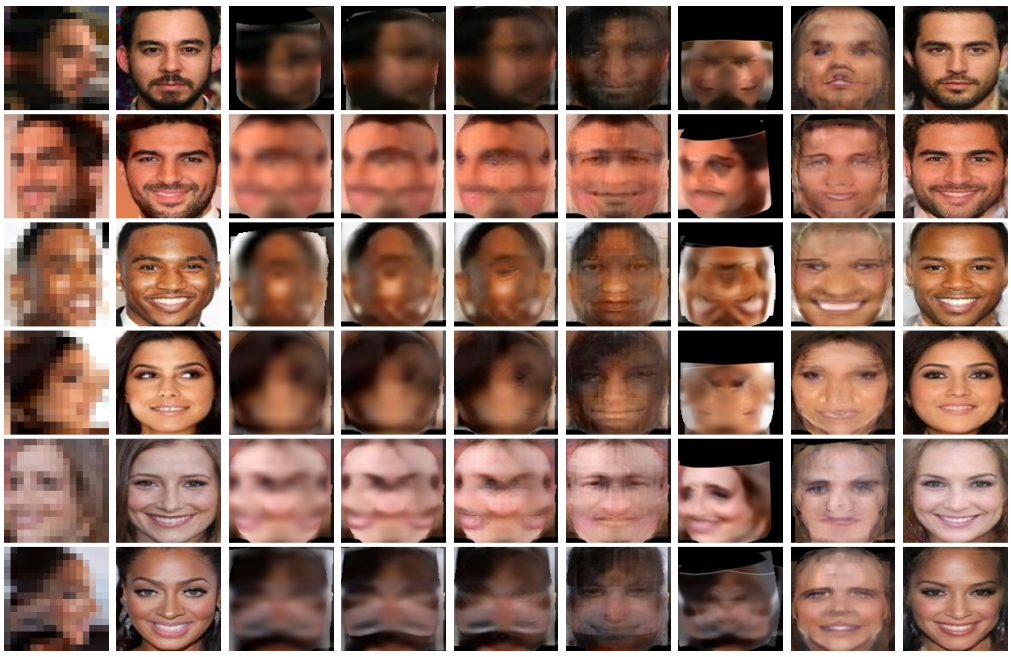
Figure 9.8: Results of the state-of-the-art methods for **frontalization followed by hallucination**. Columns: (a) Unaligned non-frontal LR inputs. (b) Original frontal HR images. (c) [Hassner et al., 2015] + bicubic interpolation. (d) [Hassner et al., 2015] + [Kim et al., 2016a]. (e) [Hassner et al., 2015] + [Ledig et al., 2017]. (f) [Hassner et al., 2015] + [Ma et al., 2010]. (g) [Hassner et al., 2015] + [Zhu et al., 2016b]. (h) [Hassner et al., 2015] + [Yu and Porikli, 2017b]. (i) Our method.

decoder and encoder in TDAE are used for noise reduction rather than frontalization. Figure. 9.6(h) shows that TDAE cannot produce realistic HR faces due to the deteriorated LR facial patterns caused by the incorrect frontalization.

Our method reconstructs authentic facial details as shown in Fig. 9.6(i) and Fig. 9.7(i). In the experiments, the face poses vary from -75° to +75°. Since our transformer subnetwork can frontalize and align LR input faces more accurately, our upsampling subnetwork achieves superior reconstruction performance from the frontalized and aligned LR features.

9.7.2 Quantitative Comparisons to the SoA

We measure the reconstruction performance of all methods on the entire test dataset by the average PSNR and the structural similarity (SSIM) scores. Furthermore, we also use the layer ReLU32 in the pretrained VGG-network to measure the differences between the ground-truth facial features and the super-resolved ones, named perceptual error (PE) score. A lower PE score indicates better super-resolution performance. Note that, when we hallucinate non-frontal faces, the hair and background regions



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 9.9: Results of the state-of-the-art methods for **hallucination followed by frontalization** by [Hassner et al., 2015]. Columns: (a) Unaligned non-frontal LR inputs. (b) Original frontal HR images. (c) Bicubic interpolation + [Hassner et al., 2015]. (d) [Kim et al., 2016a] + [Hassner et al., 2015]. (e) [Ledig et al., 2017] + [Hassner et al., 2015]. (f) [Ma et al., 2010] + [Hassner et al., 2015]. (g) [Zhu et al., 2016b] + [Hassner et al., 2015]. (h) [Yu and Porikli, 2017b] + [Hassner et al., 2015]. (i) Our method.

may not be symmetric or the same compared to the original HR face images. Thus, for a fair comparison for all methods, we compute the PSNR and SSIM on the face regions.

We report results for two possible scenarios. In the first case, we first apply the method of Hassner et al. [2015] to frontalize LR face images, and then super-resolve the frontalized LR images by the state-of-the-art SR/FSR methods (denoted as F+H). In the second case, we super-resolve LR face images first by the state-of-the-art SR/FSR methods and then frontalize the upsampled results by the method of Hassner et al. [2015] (denoted as H+F). We apply STN_0 to align LR inputs uprightly in both cases. Table. 9.1 shows that our method achieves the superior performance in comparison to the other methods, and outperforms the second best method over **4.0 dB** in PSNR.

Table 9.2 indicates the PSNR and SSIM scores for different out-of-plane rotation degrees in the F+H and the H+F cases. In Tab. 9.2, the first and second numbers denote PSNR and SSIM scores respectively. As indicated in Tab. 9.2, first frontalizing

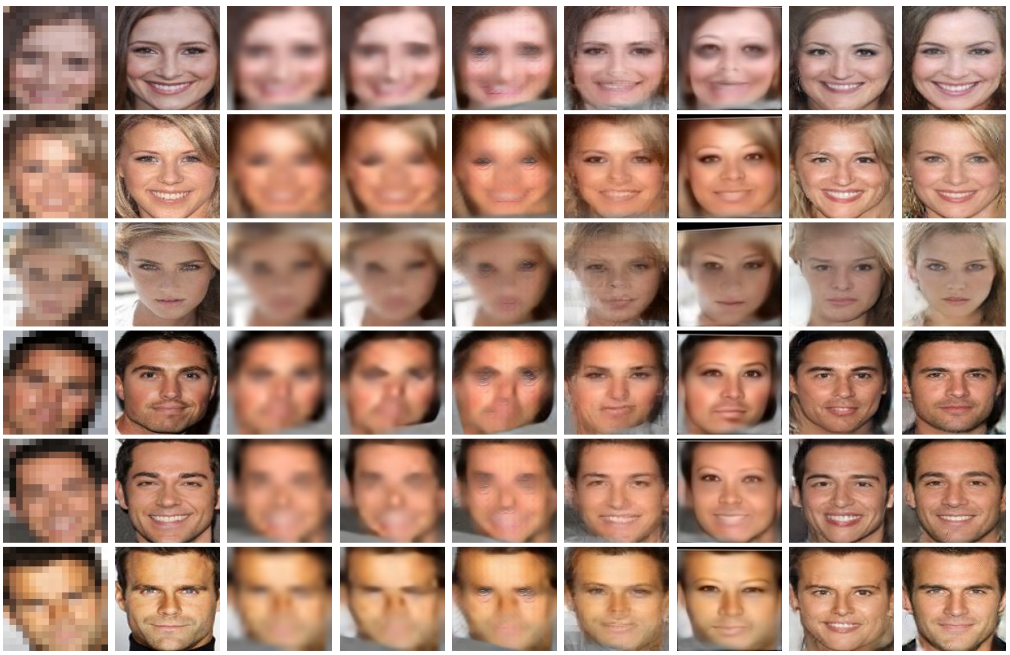


Figure 9.10: Results of the state-of-the-art face hallucination methods for frontal LR faces. Columns: (a) Unaligned non-frontal LR inputs. (b) Original frontal HR images. (c) Bicubic interpolation. (d) Results of Kim et al. [2016a]. (e) Results of Ledig et al. [2017]. (f) Results of Ma et al. [2010]. (g) Results of Zhu et al. [2016b]. (h) Results of Yu and Porikli [2017b]. (i) Our method.

and then upsampling faces can achieve slightly better results than first upsampling followed by frontalization. This also implies that it is easier to super-resolve frontal LR facial patterns than non-frontal ones. Because of the mirror symmetry operation in [Hassner et al., 2015], the PSNR and SSIM scores of the other methods in the positive degrees are lower than those in the negative degrees, as seen in Tab. 9.2. However, our method does not have this effect and produces consistent PSNR scores in both negative and positive degrees. Furthermore, as the rotation degree increases, our method does not degrade like the other methods. From 0° to $\pm 75^\circ$, our performance only decreases 1.95 dB while the performance of the second best method decreases 3.75 dB.

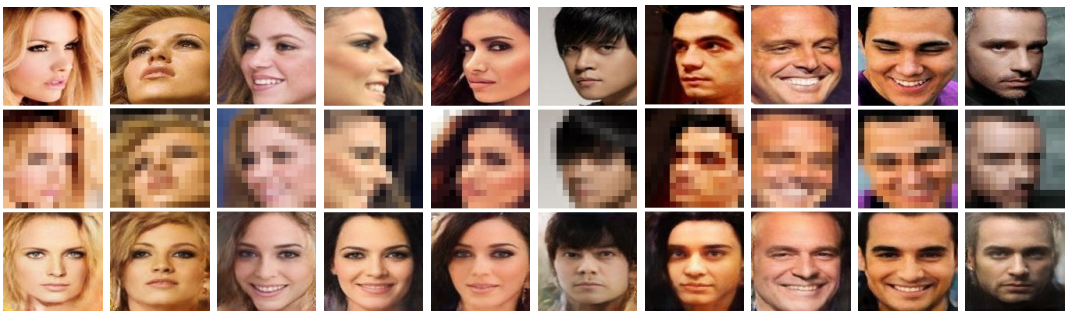
9.7.3 Comparisons with SoA on Face Retrieval

It is important to notice that we do not claim our method is designed for face recognition for two reasons: (i) we do not explicitly incorporate an identification objective in our formulation, and (ii) it might seem fruitless to attempt recognizing people in such tiny non-frontal images even for humans.

Yet, to our advantage, our method achieves significant improvement in face re-

Table 9.1: Quantitative evaluations on the entire test dataset.

H Method	F [Hassner et al., 2015]+H			H+F [Hassner et al., 2015]		
	PSNR	SSIM	PE	PSNR	SSIM	PE
Bicubic	20.99	0.80	3.56	20.41	0.79	3.76
VDSR	21.04	0.80	3.42	20.47	0.79	3.61
SRGAN	20.94	0.80	3.34	20.34	0.79	3.60
Ma *et al.*	21.60	0.82	2.99	21.15	0.80	3.28
CBN	20.61	0.79	3.94	19.40	0.77	4.74
TDAE	20.68	0.79	3.49	19.89	0.77	3.94
Ours	**25.69**	**0.87**	**1.97**	**25.69**	**0.87**	**1.97**



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 9.11: Results on LR face images beyond 3D model and training poses. Top row: real HR images. Middle row: unaligned LR images. Bottom row: our frontalized and hallucinated results.

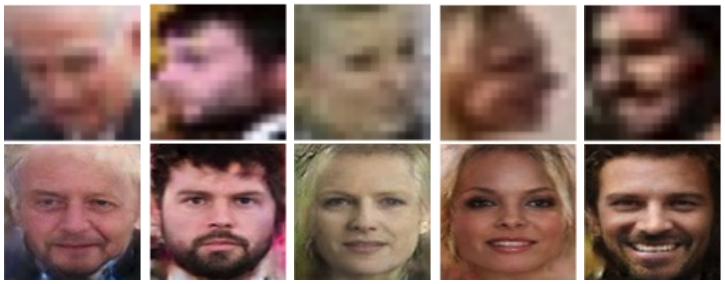
trieval performance as shown in Tab. 9.4. We use an off-the-shelf deep face recognition model [Parkhi et al., 2015] to evaluate the performance of all the methods. First, we randomly choose 100 frontal faces from the test data as our gallery. We generate their corresponding four LR non-frontal images, and employ six algorithms listed above to hallucinate the frontal HR faces on both F+H and H+F scenarios. Following the standard protocol in [Parkhi et al., 2015], we compute the accuracy score based on whether the correct person is included within the top-5 candidates (thus, the probability of random selection is 5%). Here, we notice that directly using off-the-shelf face recognition is inappropriate to measure the similarity between generated HR faces and real HR faces because there is still a domain gap between them. For instance, the features of real faces may be different from those of generated HR faces. In order to mitigate the domain gap, we train an autoencoder by using the same protocol of training TANN to transfer HR real faces to the domain where generated HR images lie in. In this way, we can significantly reduce the domain gap.

Table 9.2: Quantitative evaluations on different out-of-plane rotation degrees.

	H Methods	-75°	-40°	0°	40°	$+75^\circ$
F+H	Bicubic	20.63 / 0.80	21.43 / 0.81	24.52 / 0.83	19.51 / 0.78	18.87 / 0.77
	VDSR	20.69 / 0.80	21.47 / 0.81	24.59 / 0.84	19.54 / 0.78	18.90 / 0.77
	SRGAN	20.58 / 0.80	21.34 / 0.80	24.53 / 0.83	19.41 / 0.78	18.81 / 0.77
	Ma *et al.*	21.15 / 0.81	22.05 / 0.82	24.90 / 0.85	20.38 / 0.80	19.53 / 0.80
	CBN	20.34 / 0.79	21.14 / 0.80	24.14 / 0.83	19.08 / 0.77	18.36 / 0.76
	TDAE	20.44 / 0.79	20.69 / 0.79	23.13 / 0.82	19.74 / 0.78	19.43 / 0.78
H+F	Bicubic	20.25 / 0.79	20.68 / 0.80	23.46 / 0.83	19.05 / 0.77	18.62 / 0.77
	VDSR	20.41 / 0.80	20.83 / 0.80	23.43 / 0.83	19.04 / 0.77	18.66 / 0.77
	SRGAN	20.36 / 0.79	20.69 / 0.79	23.12 / 0.82	18.98 / 0.77	18.53 / 0.77
	Ma *et al.*	21.23 / 0.80	21.90 / 0.81	23.37 / 0.83	19.97 / 0.79	19.26 / 0.78
	CBN	18.64 / 0.75	19.23 / 0.76	22.13 / 0.81	18.84 / 0.76	18.16 / 0.75
	TDAE	19.35 / 0.77	19.97 / 0.77	22.62 / 0.80	19.36 / 0.77	18.13 / 0.76 x
	Ours$^-$	24.86 / 0.87	25.24 / 0.87	26.58 / 0.88	25.22 / 0.87	24.78 / 0.87
	Ours	**25.02 / 0.87**	**25.72 / 0.87**	**26.97 / 0.89**	**25.70 / 0.87**	**25.03 / 0.87**

Table 9.3: Quantitative evaluations on the frontal view

Method	Bicubic	VDSR	SRGAN	Ma *et al.*	CBN	TDAE	Ours
PSNR	25.64	25.78	25.58	26.45	25.37	26.39	**26.97**
SSIM	0.86	0.86	0.85	0.88	0.86	0.87	**0.89**



(a) (b) (c) (d) (e)

Figure 9.12: Results on real LR face images. Top row: real LR images. Bottom row: our frontalized and hallucinated results.

Table 9.4: Face retrieval results for different methods.

H Method	Accuracy	
	F+H	H+F
Bicubic	5.8%	6.6%
VDSR	7.0%	8.0%
SRGAN	6.0%	9.0%
Ma *et al.*	6.0%	9.0%
CBN	6.2%	8.2%
TDAE	7.2%	5.6%
Ours	**86.7%**	

As seen in Tab. 9.4, we improve the face retrieval accuracy with a large margin of 77.7%. This also implies that our method is able to preserve the appearance similarity rather than generating averaged HR faces when frontalizing and hallucinating LR faces.

9.7.4 Comparisons with SoA on Frontal Faces

Because we do not distinguish the views of LR faces deliberately before frontalization, the frontalization method [Hassner et al., 2015] is applied to all the views of LR faces. As shown in Fig. 9.6, using the face frontalization method [Hassner et al., 2015] distorts LR input faces due to the erroneous localization of facial components and its symmetrizing operations. Therefore, the super-resolution performance of frontal LR faces degrades dramatically.

For a fair comparison, we also include an evaluation for the frontal view case where the frontalization is not employed. As shown in Tab. 9.3, our method still outperforms all others in the frontal view case. Note that, our previous method T-

Table 9.5: Quantitative evaluations on the influence of different losses

	w/o $\mathcal{L}_{tri}$			**w/** $\mathcal{L}_{tri}$		
	$\mathcal{L}_{pix}$	$\mathcal{L}_{pix+feat}$	$\mathcal{L}_{pix+feat+\mathcal{T}}$	$\mathcal{L}_{pix}$	$\mathcal{L}_{pix+feat}$	$\mathcal{L}_{pix+feat+\mathcal{T}}$
PSNR	25.01	25.17	25.33	25.19	25.33	25.69
SSIM	0.87	0.87	0.87	0.88	0.87	0.87

DAE [Yu and Porikli, 2017b] intends to increase the depth of its decoder to achieve better super-resolution performance but is limited by the GPU memory. In contrast, our network employs an autoencoder, *i.e.*, our transformer subnetwork, before upsampling, and thus it does not require as much memory as TDAE yet achieves better performance. This also demonstrates that our transformer subnetwork can not only frontalize LR profile faces but also improve super-resolution performance.

9.7.5 Influence of Different Losses

Table 9.5 indicates the influences of different losses on the performance quantitatively. As indicated in Fig. 9.4(f) and Tab. 9.5, the feature-wise loss not only improves the visual quality but also increases the quantitative results. The adversarial loss makes the hallucinated faces sharper and more realistic, as shown in Fig. 9.4(f). As illustrated in Tab. 9.5, using adversarial loss is also able to force the super-resolved face images to be frontal and thus improves the super-resolution performance.

As demonstrated in Tab. 9.5, using our triplet loss improves the final results. Because our triplet loss forces the LR profile faces to be close to their frontal ones in the latent subspace, the upsampled HR frontalized faces are more similar to their frontal ground-truths. Furthermore, we also illustrate the quantitative results without using our triplet loss for different out-of-plane rotation degrees in Tab. 9.2, marked as Ours^{-}. This experiment confirms that the triplet loss does not degrade the performance of upsampling frontal faces but improves the SR performance of LR profile faces. In addition, our triplet loss is able to reduce the reconstruction loss of LR profile faces earlier in the transformer subnetwork rather than spreading the loss through the entire upsampling network. Thus, the upsampling subnetwork can focus on learning mappings between LR and HR facial patterns as suggested in [Yu and Porikli, 2016]. With the help of the triplet loss, we can even achieve better super-resolution performance on LR frontal faces, as indicated in Tab. 9.2.

9.7.6 Performance on Faces beyond 3D models

Although our method is trained on a dataset of LR non-frontal and HR frontal image pairs synthesized by using a single 3D face model, our method can be effectively generalized to faces beyond the 3D model and the poses used in the training stage. To demonstrate this, we randomly choose face images from CelebA excluding the frontal faces used for generating our training dataset. Then we spatially deform, *i.e.*,

2D transformation including rotations, translations and scale changes, and downsample these images to obtain LR face samples. The synthesized LR faces do not share 3D shapes or poses with the examples used in the training dataset, and thus these samples are much more challenging. As shown in Fig. 9.11, our network can hallucinate and frontalize such randomly chosen images, demonstrating it is not restricted to these five poses and certain models.

We also apply our network to real LR face images chosen from the WiderFace dataset [Yang et al., 2016], where LR faces are captured in the wild. Notice that the real LR faces are even blurrier than our training samples. Our super-resolved results are shown in Fig. 9.12. Since our network does not need to select one specific model for a particular angle, our method does not require estimation of the face pose angles explicitly. Instead, our method frontalizes and hallucinates LR profile faces in different angles by a single network.

9.8 Conclusion

We introduced a transformative adversarial network to upsample and frontalize very low-resolution unaligned face images simultaneously in an end-to-end fashion. Our network is able to learn how to frontalize and align LR faces while upsampling $8\times$. Benefiting from our proposed triplet loss, we are able to enforce LR profile faces to be close to their frontal counterparts in the latent subspace and thus achieve better frontalization performance. With the help of the intra-class discriminative information and the feature constraints, our network generates realistic facial details.

Chapter 10

Face Destylization

10.1 Foreword

In previous chapters, we mainly address the problem of upsampling a low-resolution face image to its high-resolution version. Besides, recovering realistic face images from stylized portraits can be also considered as "*hallucinating*" faces, named face destylization. However, our previous methods cannot be directly applied to the face destylization task, since our previous networks only take very low-resolution face images as inputs while stylized portraits have higher resolutions. In addition, our previous networks do not have the mechanism to transfer the features extracted from stylized images to the features of real faces. Inspired by the generative adversarial networks (GANs) as well as our network URDGN presented in chapter 2, we present a network to remove styles in face portraits regardless of different types of styles in this chapter.

This chapter has been published as a conference paper: Fatemeh Shiri, Xin Yu, Piotr Koniusz, Fatih Porikli: Face Destylization. In *Digital Image Computing: Techniques and Applications (DICTA)* 1-8, 2017.

10.2 Abstract

Numerous style transfer methods which produce artistic styles of portraits have been proposed to date. However, the inverse problem of converting the stylized portraits back into realistic faces is yet to be investigated thoroughly. Reverting an artistic portrait to its original photo-realistic face image has potential to facilitate human perception and identity analysis. In this paper, we propose a novel Face Destylization Neural Network (FDNN) to restore the latent photo-realistic faces from the stylized ones. We develop a style removal network composed of convolutional, fully-connected and deconvolutional layers. The convolutional layers are designed to extract facial components from stylized face images. Consecutively, the fully-connected layer transfers the extracted feature maps of stylized images into the corresponding feature maps of real faces and the deconvolutional layers generate real faces from the transferred feature maps. To enforce the destylized faces to be similar to authentic face images, we employ a discriminative network, which consists of convolution-

al and fully connected layers. We demonstrate the effectiveness of our network by conducting experiments on an extensive set of synthetic images. Furthermore, we illustrate our network can recover faces from stylized portraits and real paintings for which the stylized data was unavailable during the training phase.

10.3 Introduction

Applying artistic styles to existing photographs has attracted much attention in both academia and industry with several interesting applications. The inverse problem of reverting an artistic portrait back to its photo-realistic version is investigated in this paper. Revealing the latent real faces can provide essential information for human perception, computer analysis and photo-realistic multimedia content editing. Since facial details and expressions in stylized portraits often undergo severe distortions and become contaminated with artifacts such as profile edges and color changes e.g., as in Fig. 10.1(a) and Fig. 10.1(e), recovering a photo-realistic face image from its stylized version is very challenging.

The seminal work of Gatys et al. [2016b] stylizes the content of an arbitrary image according to a given reference artwork and achieves appealing style transfer results, hovewer, its iterative optimization procedure is computationally costly. Several methods based on feed-forward neural networks [Ulyanov et al., 2016a,b; Johnson et al., 2016; Dumoulin et al., 2016; Li et al., 2017a; Chen and Schmidt, 2016; Zhang and Dana, 2017; Huang and Belongie, 2017] accelerate the style transfer for specific styles.

For our inverse problem, the above style transfer methods fail to recover authentic face images as shown in Fig. 10.1(f) and Fig. 10.1(g). These approaches typically use Gram matrices to capture style-related contents. Since Gram matrices are designed to measure the correlations between feature maps of a style image and a target face, the spatial structure of an output image is not guaranteed to be similar to the target face. Therefore, existing style transfer methods which rely on Gram matrices are not sufficient for restoring photo-realistic portraits.

To capture local statistics of a style image, some approaches use a so-called patch-based Generative Adversarial Network (GAN) [Li and Wand, 2016b; Isola et al., 2016]. However, patch-based GANs do not take the global structure of faces into account thus a direct application of patch-GAN may not produce satisfactory results. We will show later that patch-based methods [Li and Wand, 2016b; Isola et al., 2016] fail to attain the consistency of face colors. For the inverse problem, the patch-based GAN methods result in even bigger inconsistencies.

We note that the state-of-the-art style transfer methods [Li and Wand, 2016b; Ulyanov et al., 2016a; Johnson et al., 2016] do not fully take into consideration how to extract facial features from different stylized images and then recover realistic face images. Our goal is to reveal the latent real face images from multiple style portraits (seen styles) and achieve destylization even when the styles are not available in the training dataset (unseen styles).

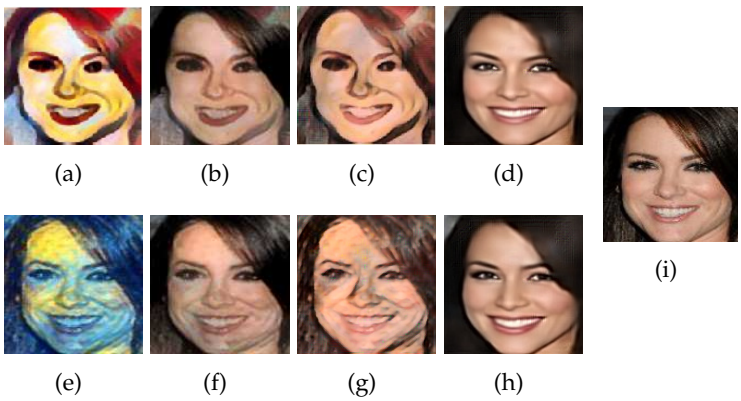


Figure 10.1: Comparison to the state-of-art methods. (a) and (e) 128×128 stylized face images in *Candy* style (which is seen and used for training) and in *Starry Night* style (which is unseen style), respectively. (b, f) Results obtained by applying the method of Gatys et al. [2016b] for the given stylized faces. (c, g) Results obtained by applying the method of Johnson et al. [2016]. (d, h) Our destylization results. (i) 128×128 ground-truth face image (used for evaluation purposes; not available to the algorithm for training).

To this end, we propose a novel destylization network that automatically maps the stylized faces to photo-realistic ones in an end-to-end fashion. Our network is composed of two components: a generative part, named *Style Removal Network (SRN)*, and a discriminative part. SRN constitutes convolutional, fully-connected and deconvolutional layers. The convolutional layers are exploited to extract facial components from stylized face images. As we aim to generate realistic face images, a fully-connected layer is developed to map the extracted feature maps of stylized faces to the feature maps of real faces. Then the mapped feature maps are projected to the image domain, thus forming face images. The discriminative network enforces the generated face images to lie in the same latent space as the realistic face images, similar to [Goodfellow et al., 2014; Denton et al., 2015; Yu and Porikli, 2016]. We train the entire network on a large-scale dataset of stylized and real face pairs. Our proposed framework can restore important facial details and attributes thanks to the style removal and discriminative subnetworks.

Furthermore, we observe that the filters of Convolutional Neural Network (CNN) learned during training (seen styles) are able to extract features from images containing unseen styles. Thus, the facial information of stylized portraits can be extracted and used to represent features of real faces. Therefore, our network can also restore the images of faces given an unseen style. In the experimental section, we demonstrate that our network is able to recover realistic faces from both seen and unseen styles e.g., synthesized and original portraits and paintings.

Below, we summarize our main contributions:

- We propose FDNN which is able to generate photo-realistic faces from stylized ones. The results resemble accurately the ground-truth faces in terms of facial

properties e.g., facial profiles and expressions.

- We develop a style removal sub-network to extract features from stylized input face images, then map these style features to real facial features and re-project them to the image domain for the purpose of generating authentic looking faces.
- We provide a dataset of pairs of the stylized and real face images used in our experiments to stimulate further research in destylization.

To the best of our knowledge, our framework is the first attempt to provide a unified approach for face destylization which can remove both seen and unseen styles (observed cf. unobserved styles during training).

10.4 Related Work

Next, we briefly review deep generative image models, deep style transfer methods, and image translation approaches.

10.4.1 Deep Generative Image Models

Recently, several frameworks have been proposed for image generation, such as variational auto-encoders [Kingma and Welling, 2013], auto-regressive models [Van Den Oord et al., 2016], and GANs [Goodfellow et al., 2014]. Among these models, GANs generate impressive results because they employ adversarial losses that force the generated images to be indistinguishable from their real counterparts. In order to improve the stability of the training procedure of GANs, various methods have been proposed [Huang et al., 2017c; Denton et al., 2015; Isola et al., 2016; Reed et al., 2016; Salimans et al., 2016; Arjovsky et al., 2017]. GANs are also employed by the style transfer [Li and Wand, 2016b] and cross-domain image generation [Bousmalis et al., 2017; Ioffe and Szegedy, 2015; Liu and Tuzel, 2016; Liu et al., 2017; Kim et al., 2017] approaches. Li and Wand [2016b] train a Markovian GAN for image style transfer such that a discriminative training is applied on Markovian neural patches to capture local style statistics. However, patch-based methods may fail to capture the global structure of objects.

10.4.2 Deep Style Transfer

Style transfer methods transfer the style of a specific artwork into a given photograph. They can be divided into two categories: *image optimization-based* and *feed forward* methods.

The optimization-based method [Gatys et al., 2016b] transfers the style by updating pixels of the image iteratively. It minimizes the distance between Gram matrices generated from feature maps of the style and synthesized image with respect to input noise. Gram matrices capture so-called feature co-occurrences and they are

popular in image recognition [Koniusz et al., 2017b,a; Koniusz and Cherian, 2016]. The approach [Yin, 2016] initializes the optimization algorithm with a content image instead of noise. Li and Wand [2016a] use Markov Random Field (MRF) in the deep feature space to enforce local patterns. The work [Gatys et al., 2016a] employs linear models to transfer styles and to preserve colors by matching color histograms. Gatys et al. [2017] detect and control spatial, color and scale factors during the stylization process. In [Risser et al., 2017], the loss function is improved by imposing a histogram-based loss. The above optimization-based methods require a time-consuming iterative optimization process, which limits their practical application.

In contrast, *feed-forward* approaches replace the original on-line iterative optimization procedure by off-line training to produce stylized images through a single forward pass [Ulyanov et al., 2016a; Johnson et al., 2016; Li and Wand, 2016b]. Johnson et al. [2016] train the generative network by perceptual loss functions. The architecture of their generator network follows the work [Radford et al., 2015]. However, they additionally use residual blocks and replace pooling layers by so-called fractionally strided convolutions. In a concurrent work, Ulyanov et al. [2016a] use a multi-resolution architecture for their generator network. Li and Wand [2016b] pre-compute a Markovian GAN which captures the feature statistics of patches. To achieve faster convergence, Ulyanov et al. [2016b, 2017] replace batch normalization with instance normalization in their generator. These feed-forward approaches [Ulyanov et al., 2016a; Johnson et al., 2016; Li and Wand, 2016b; Ulyanov et al., 2016b] are three orders of magnitude faster than optimization-based style transfer methods. However, these networks only transfer images for a predefined style and they need to be re-trained for each new style. Some recent approaches improve the style transfer from a single style to multiple styles [Chen and Schmidt, 2016; Dumoulin et al., 2016]. Dumoulin et al. [2016] propose to train a style transfer network for multiple styles by the use of a conditional instance normalization. Given feature activations of the content and style images, Chen and Schmidt [2016] replace the content features with the closest-matching style features patch-by-patch. A recent summary of state-of-the-art stylization methods can be found in the survey paper [Jing et al., 2017].

10.4.3 Image Transformation

Mapping images from one domain to another has a wide range of applications. The idea of image transformation comes from so-called image analogies [Hertzmann et al., 2001] which focuses on the non-parametric patch-based texture synthesis from a single input-output training image pair. Methods [Isola et al., 2016; Yu and Porikli, 2016; Sangkloy et al., 2017; Karacan et al., 2016; Denton et al., 2015; Radford et al., 2015; Salimans et al., 2016] employ neural networks to learn a parametric translating function from a large dataset of input-output pairs, such as super-resolution and colorization. Isola et al. [2016] propose the âĂIJpix2pixâĂİ framework to learn a mapping from input to output by a conditional GAN. Similar ideas have been applied to generating photographs from sketches [Sangkloy et al., 2017], semantic layout and scene attributes [Karacan et al., 2016].

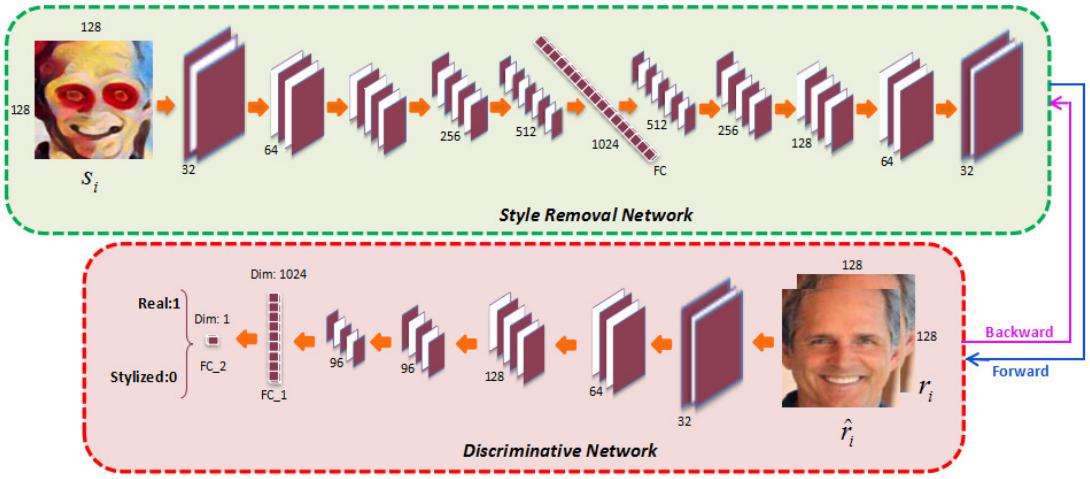


Figure 10.2: Face destylization neural network consists of two parts: a generative network (green frame) and a discriminative network (red frame).

Moreover, Isola et al. [2016] also use a convolutional patchGAN classifier for their discriminator network. The above patch-based method does not take the global structure of faces into account. Furthermore, their network employs the architecture "Unet" to transfer the source to the target domain and utilizes low-level features in the generative part that can result in distorted facial images. In contrast, our approach takes into account the global structure of faces and learns how to extract usuful features for face destylization.

10.5 Method

Our FDNN network has two components: (i) a Style Removal Network (SRN), which transforms stylized faces to the photo-realistic ones, and (ii) a discriminative network, which enforces the generated faces by SRN to be indistinguishable from the real faces. Figure 10.2 illustrates the overall architecture of our proposed network.

10.5.1 Style Removal Network

In Fig. 10.2, our SRN is enclosed by the green frame. SRN aims at removing various styles of portraits and generating realistic faces. Our SRN comprizes convolutional layers followed by batch normalization layers, a fully connected layer and deconvolutional layers followed by batch normalization layers. The convolutional layers are employed to extract facial features from stylized face images. Then, we incorporate a fully-connected layer to transfer the extracted feature maps of stylized images into the feature maps of real faces. In order to synthesize images of real faces, deconvolutional layers project these transferred feature maps to the image domain.

In order train SRN, we use stylized portraits as inputs and their corresponding

ground-truth images of real faces as desired supervising output signals. Since a dataset of portrait/real face pairs is not readily available, we opt to generate a large number of stylized faces in numerous styles from real face images. Figure 10.3(c) and Fig. 10.3(f) illustrate the effectiveness of SRN.

10.5.2 Discriminative Network

Using only Euclidean distance, i.e. ℓ_2 loss, between the destylized faces and the corresponding ground-truth real ones tends to generate over-smoothed results as shown in Fig. 10.3(c) and Fig. 10.3(f), and this phenomenon is also mentioned in [Yu and Porikli, 2016]. Therefore, a class-specific discriminative objective is also incorporated into our SRN, aiming to enforce the destylized face images to lie on the same latent space of the authentic face images.

As shown in the red frame of Fig. 10.2, the discriminative network is constructed by convolutional and fully connected layers. Its role is to determine whether an image is sampled from real face images or the destylized ones. With the help of the so-called discriminative adversarial loss, we can force generated destylized faces to be more similar to real ones. This is achieved by back-propagating the adversarial loss to update the parameters of SRN. Figure 10.3(d) and Fig. 10.3(g) illustrate the impact of the adversarial loss on the final results.

10.5.3 Training Details

Our FDNN is trained in an end-to-end manner. We use Stylized Face (SF) and Real Face (RF) ground-truth image pairs (s_i, r_i) as our training dataset, where r_i represents the real face images aligned by eyes only, and s_i is a synthesized SF image from r_i. For each real face r_i, we generate eight different SFs i.e., Edvard Munch's *Scream*, *Candy*, *Feathers*, *Starry Night* by Van Gogh, *la Muse* by Pablo Picasso, Wassily Kandinsky's *Composition VII*, *Mosaic* and Francis Picabia's *Udnie*, and obtain SF/RF training pairs. The stylized faces of *Scream*, *Candy* and *Feathers* are used in the training stage. As detailed in Sec. 10.6, we find that these distinct portraits provide a sufficient training data for our needs.

Our training strategy enforces the generated face $\hat{r}_i$ to be similar to its corresponding ground-truth r_i. Therefore, we employ a pixel-wise ℓ_2 loss between $\hat{r}_i$ and r_i, and we minimize the objective $Q(\mathcal{T})$ of SRN as follows:

$$\begin{aligned} \min_{\mathcal{T}} \quad Q(\mathcal{T}) &= \mathbb{E}_{(\hat{r}_i, r_i)\sim p(\hat{r}, r)} \|\hat{r}_i - r_i\|_F^2 \\ &= \mathbb{E}_{(s_i, r_i)\sim p(s, r)} \|G_{\mathcal{T}}(s_i) - r_i\|_F^2, \end{aligned} \tag{10.1}$$

where $\mathcal{T}$ indicates the parameters of SRN generator G, $p(s, r)$ represents the joint distribution of the SF and RF images in the training dataset and $p(\hat{r}, r)$ represents the joint distribution of destylized and the ground-truth faces.

(a) (b) (c) (d) (e) (f) (g)

Figure 10.3: Contribution of each FDNN part. (a) Ground-truth real face images. (b) Input portrait of *Feathers* from training styles and (e) input portrait of *la Muse* from unseen styles (from test dataset; not available in the training stage). (c, f) Destylization results without adversarial loss. (d, g) Our final results.

To achieve high-quality results, we force SRN to fool the discriminative supervising network that employs a binary classifier which task is to distinguish whether incoming image samples contain real or generated faces. Similar to the idea of [Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015], our goal is to make the discriminative network fail to distinguish generated faces from real ones. Hereby, we maximize the adversarial loss of the discriminative network $F(\mathcal{L})$ as follows:

$$\begin{aligned}\max_{\mathcal{L}} \quad F(\mathcal{L}) &= \mathbb{E}\left[\log D_{\mathcal{L}}(r_i) + \log(1 - D_{\mathcal{L}}(\hat{r}_i))\right] \\ &= \mathbb{E}_{r_i \sim p(r)}[\log D_{\mathcal{L}}(r_i)] + \mathbb{E}_{\hat{r}_i \sim p(\hat{r}))}[\log(1 - D_{\mathcal{L}}(\hat{r}_i))],\end{aligned} \tag{10.2}$$

where $\mathcal{L}$ represents the parameters of the discriminative network D, $p(r)$ and $p(\hat{r})$ indicate the distributions corresponding to the real and the generated faces, respectively, and $D_{\mathcal{L}}(r_i)$ and $D_{\mathcal{L}}(\hat{r}_i)$ are the outputs of network D. Since the loss F is back-propagated to update not only the parameters $\mathcal{L}$ but also $\mathcal{T}$, we also minimize the objective function $Q_f(\mathcal{T})$ of SRN:

$$\min_{\mathcal{T}} \quad Q_f(\mathcal{T}) = \mathbb{E}_{(s_i, r_i) \sim p(s,r)} \|G_{\mathcal{T}}(s_i) - r_i\|_F^2 + \lambda \mathbb{E}_{s_i \sim p(s))}[\log D_{\mathcal{L}}(G_{\mathcal{T}}(s_i))], \tag{10.3}$$

where scalar λ is a trade-off between supervising the generator by the ground-truth data vs. the discriminator supervision, respectively.

Since each layer in our FDNN is differentiable, we employ the Root Mean Square Propagation (RMSprop) [Hinton, 2012] to update $\mathcal{T}$ and $\mathcal{L}$. In order to maximize the adversarial loss F, the stochastic gradient ascent is used to update $\mathcal{L}$:

$$\begin{aligned}\Delta^{i+1} &= \beta \Delta^i + (1 - \beta)(\frac{\partial F}{\partial \mathcal{L}})^2, \\ \mathcal{L}^{i+1} &= \mathcal{L}^i + \alpha \frac{\partial F}{\partial \mathcal{L}} \frac{1}{\sqrt{\Delta^{i+1} + \epsilon}},\end{aligned} \tag{10.4}$$

where α and β represent the learning and the decay rate respectively, i is the iteration

index, Δ is an auxiliary variable, and ϵ is set to 10^{-8} to avoid division by zero. For SRN, both losses Q and F are used to update $\mathcal{T}$ by the stochastic gradient descent:

$$\begin{aligned} \Delta^{i+1} &= \beta\Delta^{i} + (1-\beta)(\frac{\partial Q_f}{\partial \mathcal{T}})^2, \\ \mathcal{T}^{i+1} &= \mathcal{T}^{i} - \alpha(\frac{\partial Q_f}{\partial \mathcal{T}})\frac{1}{\sqrt{\Delta^{i+1}+\epsilon}}, \end{aligned} \tag{10.5}$$

We set $\lambda = 0.01$ to limit supervision of the generator by the discriminator and allow appearance-based learning from the ground-truth image pairs. As the iterations progress, the output faces will resemble the real faces more. Therefore, we gradually reduce the impact of the discriminative network by decreasing λ,

$$\lambda^n = \max\{\lambda \cdot 0.995^n, \lambda/2\}, \tag{10.6}$$

where n is the index of the epochs. Eqn. 10.6 not only increases the impact of the appearance similarity term but also preserves the class-specific discriminative information in the training phase.

10.5.4 Implementation Details

Similar to [Goodfellow et al., 2014; Radford et al., 2015], we employ batch normalization after the convolutional and deconvolutional layers of SRN except for the last deconvolutional layers. We also use leaky rectified linear units (leakyReLU) with a negative slope 0.2 as non-linear activation functions. For training, the learning rate α is set to 0.001 and multiplied by 0.99 after each epoch, and the decay rate is set to 0.01. The discriminative network is only employed in the training phase. In the testing phase, we feed a stylized face image into the SRN to obtain its realistic version.

10.6 Synthesized Dataset

Training of a deep neural network requires a large number of samples to prevent models from overfitting to the training data. The publicly available large-scale face datasets [Huang et al., 2007; Liu et al., 2015] only provide faces in the wild but not pairs of real images of faces and their stylizations. Therefore, we opt to generate a large number of stylized faces from the corresponding real face images in eight distinct styles: *Starry Night, la Muse, Composition VII, Scream, Candy, Feathers, Mosaic* and *Udnie*. To generate such a dataset, there are a number of alternative feed-forward approaches available [Ulyanov et al., 2016a,b; Johnson et al., 2016]. We choose the recent feed-forward style transfer model [Johnson et al., 2016].

We firstly select at random 10K images of cropped real faces (within $\pm 30^\circ$ orientation) from the CelebA dataset [Liu et al., 2015] for training and 1K images for testing, and then resize them to 128×128 pixels. We use 10K training images as our real ground-truth faces r_i. To generate three different portraits of each face, we retrain

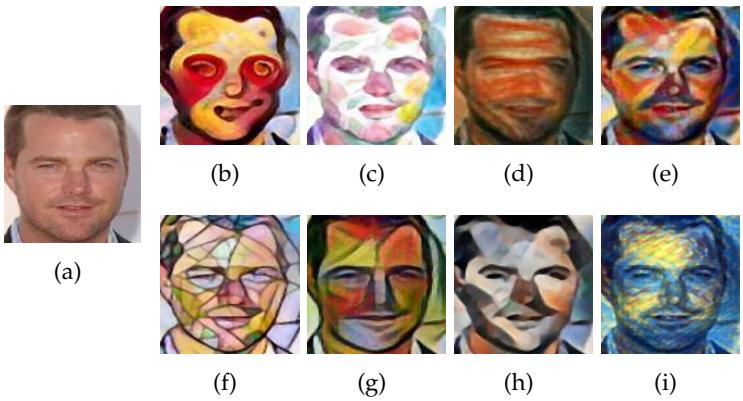


Figure 10.4: Illustration of the synthesized dataset. (a) Original real face image. (b)-(d) The synthesized stylized faces of (a) form *Candy, Feathers* and *Scream* which have been used for training our network. (e)-(i) The synthesized stylized faces of (a) form *Composition VII, Mosaic, la Muse, Udnie* and *Starry* styles which have not been used for training.

the style transfer model [Johnson et al., 2016] for *Scream, Candy* and *Feathers* styles separately. Finally, we obtain 30K SF/RF pairs for training our network. We also use 1K test real faces to generate 8K SF/RF face pairs from eight different styles (each test face corresponds to eight distinct styles) for testing our network. Figure 10.4 shows the stylized samples that are generated from a single real image containing a face (Fig. 10.4(a)).

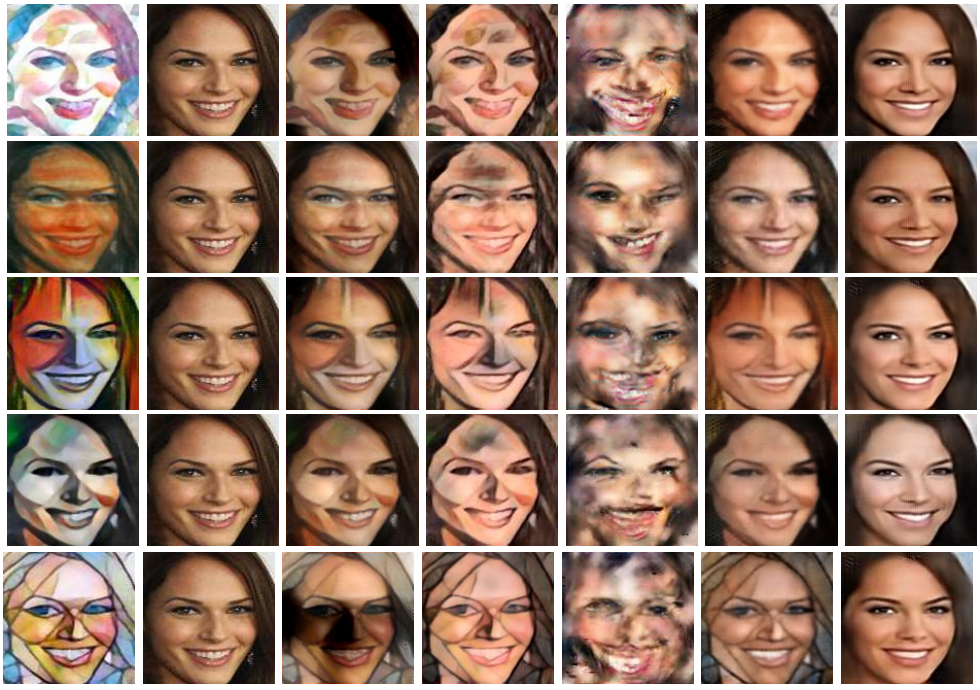
10.7 Experiments

We compare our method qualitatively and quantitatively against four different state-of-the-art methods. As explained in Sec. 10.6, we gather 30K SF/RF face pairs from three styles as a training dataset and 8K SF/RF pairs faces generated from different eight styles for testing. In all the cases, the ground-truth real faces and the corresponding stylized faces do not overlap in the training and testing datasets. Since our method is feed-forward and no optimization is required at test time. Our method cost 10 ms for a 128-by-128 image.

10.7.1 Qualitative Evaluation

Comparison to the state of the art. Firstly, we note that the test stylized face images were not used for training of our model. The resolution of stylized and destylized output faces in this study is 128×128 pixels. We compare our approach against four different approaches as detailed below.

We compare our work against the method of Gatys et al. [2016b], an image-optimization based style transfer method free of the training stage. To generate real faces, this network aims to preserve the contents of a portrait and the corresponding



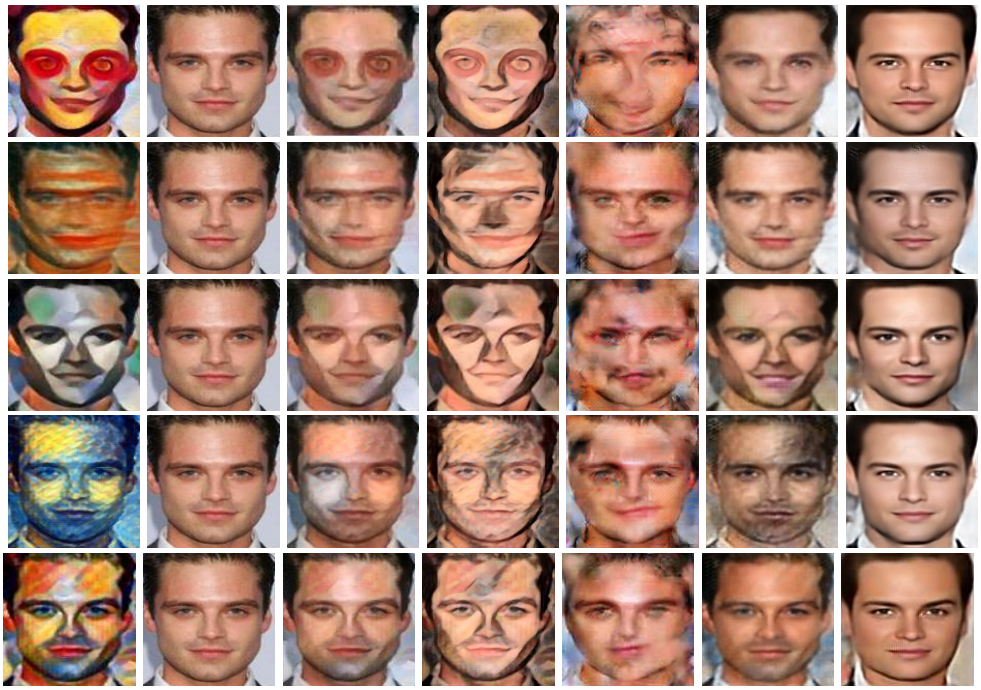
(a) (b) (c) (d) (e) (f) (g)

Figure 10.5: Results of the state-of-the-art methods for face destylization. (a) Input portraits of *Feathers, Scream* from seen styles as well as *la Muse, Udnie* and *Mosaic* from unseen styles (from test dataset; not available to the algorithm during training) (b) Ground-truth images of real faces. (c) Results of Gatys et al. [2016b]. (d) Results of Johnson et al. [2016]. (e) Results of Li and Wand [2016b] (MGAN). (f) Results of Isola et al. [2016] (pix2pix). (g) Our results.

photo-realistic face. The network fails to produce appealing results as shown in Fig. 10.5(c) and Fig. 10.6(c). This method captures the correlations in feature maps of style and synthesized images by Gram matrices and discards the spatial arrangement at the pixel level.

We also use a feed-forward approach [Johnson et al., 2016] for destylization. Due to the Gram matrix, this method also produces distorted facial details. As shown in the first row of Fig. 10.5(d), the edges of the face were blurred and the color of the face is not consistent. From the first row of Fig. 10.6(d), one can see that the style overlapping with the eyes was not fully removed. Thus, their network fails to restore authentic looking eyes.

Li and Wand [2016b] propose a patch-based style transfer method, known as Markovian GAN. We use their network for destylization and apply their standard protocols. As such a method is trained with stylized face patches, it cannot capture the global structure of facial images. As seen in Fig. 10.5(e) and Fig. 10.6(e), the facial color consistency cannot be preserved either. In contrast, our method produces highly-consistent facial colors and captures the global structure of faces well.



(a) (b) (c) (d) (e) (f) (g)

Figure 10.6: Result of the state-of-the-art methods for face destylization. (a) Input portraits of *Candy* and *Scream* from seen styles as well as *la Muse, starry Night* and *Mosaic* from unseen styles (from test dataset; not available to the algorithm during training) (b) Ground-truth images of real faces. (c) Results of Gatys et al. [2016b]. (d) Results of Johnson et al. [2016]. (e) Results of Li and Wand [2016b] (MGAN). (f) Results of Isola et al. [2016] (pix2pix). (g) Our results.

Table 10.1: Comparison of physical (PSNR) and perceptual (SSIM) quality measures for the entire test dataset.

Method	Seen Styles		Unseen Styles	
	PSNR	SSIM	PSNR	SSIM
Gatys [Gatys et al., 2016b]	22.6792	0.8656	20.2320	0.8493
Johnson [Johnson et al., 2016]	22.8481	0.8745	21.2184	0.8632
MGAN [Li and Wand, 2016b]	19.5254	0.8548	17.2645	0.8270
pix2pix [Isola et al., 2016]	22.9893	0.8871	21.6316	0.8860
Ours	**23.2086**	**0.9087**	**22.4430**	**0.9015**

Isola et al. [2016] present a general image-to-image translation method, known as pix2pix. It employs the architecture "Unet" for the generator network. A convolutional patch based neural network is trained to discriminate between image patches

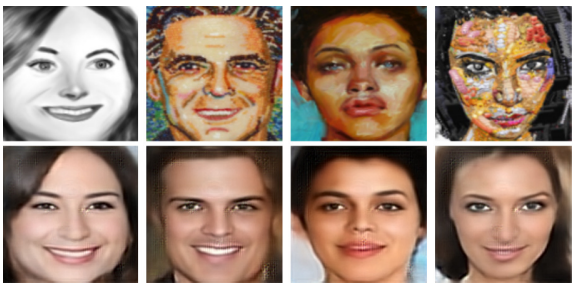


Figure 10.7: Results for the original paintings. Top row: the original portraits from DevianArt. Bottom row: our destylization results.

extracted from real and generated faces. In addition, the low-level features from the bottom layers of Unet also participate in generating faces. These low-level features corrupt the destylized images and result in poor removal of styles in the images e.g., for unseen styles. As shown in Fig. 10.5(f) and Fig. 10.6(f), while pix2pix can produce acceptable results for seen styles, it fails to remove previously unseen styles. As shown in the fourth row of Fig. 10.6(f), obvious artifacts appear in the generated face of an unseen style.

Our destylized results exhibit higher fidelity with respect to the real faces, better consistency in colors and can even preserve the identity of the subject, as shown in Fig. 10.5(g) and Fig. 10.6(g).

10.7.2 Quantitative Evaluation

Face Reconstruction. In Tab. 10.1, we report the reconstruction performance measured on the entire test dataset for each approach. We use the average Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) [Wang et al., 2004] scores for which higher scores indicate better results.

We report performance of destylization algorithms for two scenarios: seen and unseen styles. For the seen styles, results of the state-of-the-art style transfer methods are shown in the first and second rows of Fig. 10.5 and Fig. 10.6. For the destylization of portraits of unseen styles, we demonstrate results in the third, fourth and fifth rows of Fig. 10.5 and Fig. 10.6.

Table 10.1 shows that our results achieve better PSNR and SSIM than the state-of-the-art methods on seen styles and unseen styles. This performance also coincides with the visual results.

Consistency Analysis. Intuitively, the destylized faces from the different styles of the same person should look similar. Examples generated from multiple styles are shown in Fig. 10.5(g) and Fig. 10.6(g). In this experiment, we demonstrate that our method not only recovers realistic faces with high fidelity but also generates faces looking close to each other given multiple styles of the same person on input. This indicates that SRN can indeed extract facial features from portraits despite different styles and transfer these features to recover underlying faces.

To evaluate the consistency of generated faces from different portraits of the same

Table 10.2: Comparison of consistency between destylized faces from various seen and unseen styles.

	Seen Styles	Unseen Styles
Gatys [Gatys et al., 2016b]	82%	83%
Johnson [Johnson et al., 2016]	73%	72.5%
MGAN [Li and Wand, 2016b]	2%	1%
pix2pix [Isola et al., 2016]	93.33%	85.1%
Ours	**98%**	**90.8%**

person, we adapt the off-the-shelf deep face recognition approach [Parkhi et al., 2015]. First, we randomly choose 100 RF and 800 corresponding SF faces from eight different styles in the test dataset for our gallery (three seen styles and five unseen styles). Then, we employ Gatys [Gatys et al., 2016b], Johnson [Johnson et al., 2016], MGAN [Li and Wand, 2016b], pix2pix [Isola et al., 2016] and our FDNN to recover real faces from eight various stylized faces. For each method, we set 100 destylized faces from the *Candy* style as a query dataset and set the other 700 destylized faces from the other seven styles as a search dataset. Following the standard protocol, we compute the Face Recognition Rate (FRR) which quantifies if the correct person is retrieved within the top-5 candidates (the probability of successful retrieval by chance is 0.71%). We also use the same procedure for other styles. Table 10.2 shows the average FRR of each method for seen and unseen styles. Our method yields high consistency score for both seen and unseen styles. This indicates the effectiveness of our FDNN in producing realistic faces of high-fidelity.

10.7.3 Performance on Original Paintings

Despite our method is trained on a synthetic dataset, it can efficiently generalize to real paintings/portraits. To demostrate this, we randomly choose some paintings with faces from DevianArt. We crop images of these faces and then align them to the CelebA face dataset in an off-line pre-processing step. Our method successfully reconstructs plausible facial details from real paintings as shown in Fig. 10.7. This highlights that our method is not restricted to synthesized stylized faces.

10.7.4 Limitations

Our proposed network requires that the eyes of stylized faces to be aligned beforehand to a template. Without such an alignment, FDNN may generate artifacts. However, we plan to automatically align the stylized facial images in our future work. As illustrated in Fig. 10.8(a), destylization is performed on an unaligned stylized face. As a consequence, our network cannot localize facial features correctly and produces

(a) Unaligned (b) SF (c) Ours

(d) Upright pose (e) SF (f) Ours

Figure 10.8: Failures. (a) An unaligned ground-truth face. (e) Stylized face of (a). (c) Our result. (d) An upright pose. (e) Stylized face of (d). (c) Our result.

erroneous feature maps. In addition, our method may produce artifacts for portraits suffering from large pose variations, such as profile views of faces etc. Since there are not enough side-view images of faces in the training dataset, this results in artifacts. As shown in Fig. 10.8(f), the network fails to generate satisfying results for an upright pose. Exploring how to address large pose variations will be our future work.

10.8 Conclusion

We present a face destylization method that extracts features of a stylized portrait and then exploits them to generate its corresponding photo-realistic face. Our network learns a mapping from stylized facial feature maps to realistic facial feature maps. Our network can successfully extract facial features from different styles and thus is able to destylize unseen style portraits as well.

Chapter 11

Identity-preserving Face Recovery from Portraits

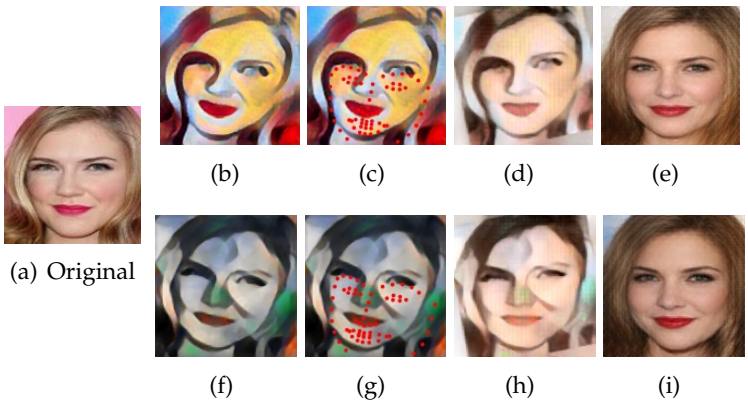
11.1 Foreword

In chapter 10, we employ a generative adversarial network to recover realistic face images from stylized portraits. In particular, a discriminative network is used to enforce the generated faces to be authentic and an ℓ_2 loss to constrain the appearance similarity between the recovered faces and the ground-truth real faces. However, there is no guarantee to preserve the identity information in the destylized faces, *e.g.*, facial characteristics. In addition, our previous face destylization mehtod can only tackle aligned portraits. State-of-the-art face alignment methods may fail to align stylized portraits since facial details have been distorted by different image styles. In this chapter, we aim to recover realistic faces from unaligned stylized portraits while preserving the identity information in the portraits. This work is also motivated by our network TDN presented in chapter 4, where Spatial Transformer Networks (STNs) are employed to align low-resolution faces during upsampling.

This chapter has been published as a conference paper: Fatemeh Shiri, Xin Yu, Fatih Porikli, Richard Hartley, Piotr Koniusz: Identity-Preserving Face Recovery from Portraits. In *IEEE Winter Conference on Application of Computer Vision (WACV)*, 102-111, 2018.

11.2 Abstract

Recovering the latent photorealistic faces from their artistic portraits aids human perception and facial analysis. However,a recovery process that can preserve identity is challenging because the fine details of real faces can be distorted or lost in stylized images. In this paper, we present a new Identity-preserving Face Recovery from Portraits (IFRP) to recover latent photorealistic faces from unaligned stylized portraits. Our IFRP method consists of two components: Style Removal Network (SRN) and Discriminative Network (DN). The SRN is designed to transfer feature maps of stylized images to the feature maps of the corresponding photorealistic faces. By embedding spatial transformer networks into the SRN, our method can compensate



(a) Original (b) (c) (d) (e) (f) (g) (h) (i)

Figure 11.1: Comparisons to the state-of-art method. (a) Ground-truth face image (from test dataset; not available in the training dataset). (b) Unaligned stylized portraits of (a) from *Candy* style (seen/used style in training). (f) Unaligned stylized portraits of (a) from *Udnie* style (unseen style in training). (c, g) Detected landmarks by [Zhang et al., 2014]. (d, h) Results obtained by [Johnson et al., 2016]. (e, i) Our results.

for misalignments of stylized faces automatically and output aligned realistic face images. The role of the DN is to enforce recovered faces to be similar to authentic faces. To ensure the identity preservation, we promote the recovered and ground-truth faces to share similar visual features via a distance measure which compares features of recovered and ground-truth faces extracted from a pre-trained VGG network. We evaluate our method on a large-scale synthesized dataset of real and stylized face pairs and attain state of the art results. In addition, our method can recover photorealistic faces from previously unseen stylized portraits, original paintings and human-drawn sketches.

11.3 Introduction

A variety of style transfer methods have been proposed to generate portraits in different artistic styles from photorealistic images. However, the recovery of photorealistic faces from artistic portraits has not been fully investigated yet. In general, stylized face images contain various facial expressions, facial component distortions and misalignments. Therefore, landmark detectors often fail to localize facial landmarks accurately as shown in Fig. 11.1(c) and Fig. 11.1(g). Thus, restoring identity-consistent photorealistic face images from unaligned stylized ones is challenging.

While recovering photorealistic images from portraits is still uncommon in the literature, image stylization methods have been widely studied. Recently, Gatys et al. [2017] achieve promising results by transferring different styles of artworks to images via the semantic contents space. Since this method generates the stylized images by iteratively updating the feature maps of CNNs, it requires costly computations. In order to reduce the computational complexity, several feed-forward CNN based

methods have been proposed [Ulyanov et al., 2016a,b; Johnson et al., 2016; Dumoulin et al., 2016; Li et al., 2017a; Chen and Schmidt, 2016; Zhang and Dana, 2017; Huang and Belongie, 2017]. However, these methods can use only a single style fixed during the training phase. Such methods are insufficient for generating photorealistic face images, as shown in Fig. 11.1(d) and Fig. 11.1(h), because they only capture the correlations of feature maps by the use of Gram matrices and discard spatial relations [Koniusz et al., 2017b,a; Koniusz and Cherian, 2016].

In order to capture spatially localized statistics of a style image, several patch-based methods [Li and Wand, 2016b; Isola et al., 2016] have been developed. However, such methods cannot capture the global structure of faces either, thus failing to generate authentic face images. For instance, patch-based methods [Li and Wand, 2016b; Isola et al., 2016] fail to attain consistency of face colors, as shown in Fig. 11.6(e). Furthermore, the state-of-the-art style transfer methods [Gatys et al., 2017; Li and Wand, 2016b; Ulyanov et al., 2016a; Johnson et al., 2016] transfer the desired styles to the given images without considering the task of identity preservation. Hence, previous methods cannot generate real faces while preserving identity.

In this paper, we develop a novel end-to-end trainable identity-preserving approach to face recovery that automatically maps the unaligned stylized portraits to aligned photorealistic face images. Our network employs two subnetworks: a generative subnetwork, dubbed Style Removal Network (SRN), and a Discriminative Network (DN). The SRN consists of an autoencoder (a downsampling encoder and an upsampling decoder) and Spatial Transfer Networks (STNs) [Jaderberg et al., 2015]. The encoder extracts facial components from unaligned stylized face images and transfer the extracted feature maps to the domain of photorealistic images. Subsequently, our decoder forms face images. STN layers are used by the encoder and decoder to align stylized faces. The discriminative network, inspired by [Goodfellow et al., 2014; Denton et al., 2015; Yu and Porikli, 2016, 2017a], forces SRN to generate destylized faces to be similar to authentic ground-truth faces.

Moreover, as we aim to preserve the facial identity information, we constrain the recovered faces to have the same CNN feature representations as the ground-truth real faces. For this purpose, we employ pixel-level Euclidean and identity-preserving loss functions to guarantee the appearance- and identity-wise similarity to the ground-truth data. We also use an adversarial loss to achieve high-quality visual results.

To train our network, we require pairs of Stylized Face (SF) and ground-truth Real Face (RF) images. Therefore, we synthesize a large-scale dataset of SF/RF pairs. We observe that our CNN filters learned on images of seen styles (used for training) can extract meaningful features from images in unseen styles. Thus, the facial information of unseen stylized portraits can be extracted and used to generate photorealistic faces, as shown in the experimental section.

The main contributions of our work are fourfold:

- We propose an IFRP approach that can recover photorealistic faces from unaligned stylized portraits. Our method generates facial identities and expres-

sions that match the ground-truth face images well.

- We use STNs as intermediate layers to compensate for misalignments of input portraits. Thus, our method does not require the use of facial landmarks or 3D face models (typically used for face alignment).
- We fuse an identity-preserving loss, a pixel-wise similarity loss and an adversarial loss to remove seen/unseen styles from portraits and recover the underlying identity.
- As large-scale datasets of stylized and photorealistic face pairs are not available, we synthesize a large dataset of pairs of stylized and photorealistic faces, which will be available on-line.

To the best of our knowledge, our method is the first attempt to provide a unified approach to the automated style removal of unaligned stylized portraits.

11.4 Related Work

In this section, we briefly review neural generative models and deep style transfer methods for image generation.

11.4.1 Neural Generative Models

There exist many generative models for the problem of image generation [Van Den Oord et al., 2016; Kingma and Welling, 2013; Goodfellow et al., 2014; Denton et al., 2015; Zhang et al., 2017a; Shiri et al., 2017]. Among them, GANs are conceptually closely related to our problem as they employ an adversarial loss that forces the generated images to be as photorealistic as the ground-truth images.

Several methods adopt an adversarial training to learn a parametric translating function from a large-scale dataset of input-output pairs, such as super-resolution [Ledig et al., 2017; Yu and Porikli, 2017a; Huang et al., 2017b; Yu and Porikli, 2017b, 2016] and inpainting [Pathak et al., 2016]. These approaches often use the ℓ_2 or ℓ_1 norm and adversarial losses to compare the generated image to the corresponding ground truth image. Although these methods produce impressive photorealistic images, they fail to preserve identities of subjects.

Conditional GANs have been used for the task of generating photographs from sketches [Sangkloy et al., 2017], and from semantic layout and scene attributes [Karacan et al., 2016]. Li and Wand [2016b] train a Markovian GAN for the style transfer; a discriminative training is applied on Markovian neural patches to capture local style statistics. Isola et al. [2016] develop "pix2pix" framework which uses so-called "Unet" architecture and the patch-GAN to transfer low-level features from the input to the output domain. For faces, this approach produces visual artefacts and fails to capture the global structure of faces.

Patch-based methods fail to capture the global structure of faces and, as a result, they generate poor destylization results. In contrast, we propose an identity-preserving loss to faithfully recover the most prominent details of faces.

Moreover, there exist several methods to synthesize sketches from photographs (and vice versa) [Nejati and Sim, 2011; Yuen and Man, 2007; Tang and Wang, 2003; Sharma and Jacobs, 2011]. While sketch-to-face synthesis is a related problem, our unified framework can work with various more complex styles.

11.4.2 Deep Style Transfer

Style transfer is a technique which can render a given content image (input) by incorporating a specific painting style while preserving the contents of input. We distinguish *image optimization-based* and *feed-forward* style transfer methods. The seminal optimization-based work [Gatys et al., 2016b] transfers the style of an artistic image to a given photograph. It uses an iterative optimization to generate a target image which is randomly initialized (Gaussian distribution). During the optimization step, the statistics of the neural activations of the target, the content and style images are matched.

The idea of Gatys et al. [2016b] inspires many follow-up studies. Yin [2016] presents a content-aware style transfer method which initializes the optimization algorithm with a content image instead of a random noise. Li and Wand [2016a] propose a patch-based style transfer method by combining Markov Random Field (MRF) and CNN techniques. The work [Gatys et al., 2016a] proposes to transfer the style by using linear models. It preserves colors of content images by matching color histograms.

Gatys et al. [2017] decompose styles into perceptual factors and then manipulate them for the style transfer. Selim et al. [2016] modify the content loss through a gain map for the head portrait painting transfer. Risser et al. [2017] use histogram-based losses in their objective and build on the algorithm of Gatys et al. [2016b]. Although the above optimization-based methods further improve the quality of style transfer, they are computationally expensive due to the iterative optimization procedure, thus limiting their practical use.

To address the poor computational speed, feed-forward methods replace the original on-line iterative optimization step with training a feed-forward neural network off-line and generating stylized images on-line [Ulyanov et al., 2016a; Johnson et al., 2016; Li and Wand, 2016b].

Johnson et al. [2016] train a generative network for a fast style transfer using perceptual loss functions. The architecture of their generator network follows the work [Radford et al., 2015] and also uses residual blocks. Another concurrent work [Ulyanov et al., 2016a], named Texture Network, employs a multi-resolution architecture in the generator network. Ulyanov et al. [2016b, 2017] replace the spatial batch normalization with the instance normalization to achieve a faster convergence. Wang et al. [2017] enhance the granularity of the feed-forward style transfer with multimodal CNN which performs stylization hierarchically via multiple losses deployed across

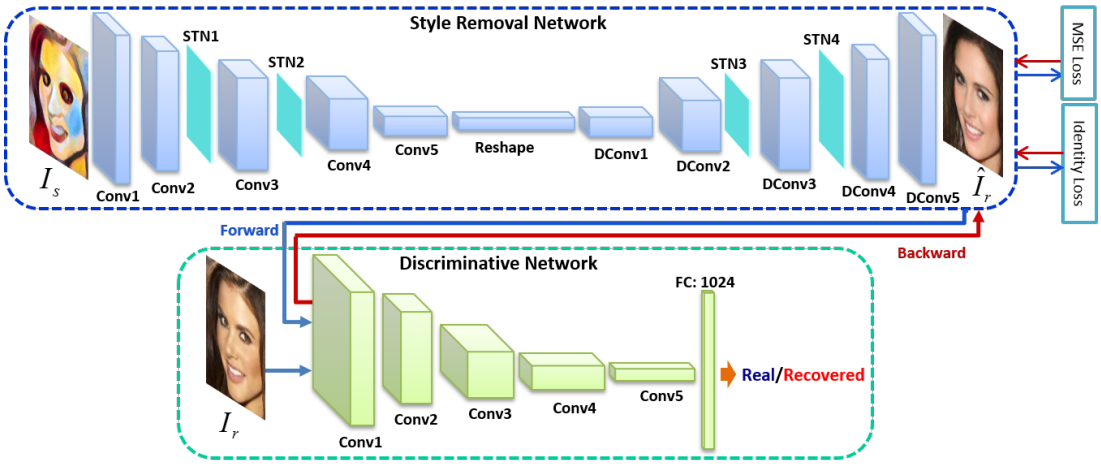


Figure 11.2: The Architecture of our identity-preserving face destylization framework consists of two parts: a style removal network (blue frame) and a discriminative network (green frame).

multiple scales.

Those feed-forward methods perform stylization about 1000 times faster than the optimization-based methods. However, they cannot adapt to arbitrary styles that are not used for training. For synthesizing an image from a new style, the entire network needs retraining. To deal with such a restriction, a number of recent approaches encode multiple styles within a single feed-forward network [Dumoulin et al., 2016; Chen and Schmidt, 2016; Chen et al., 2017; Li et al., 2017a].

Dumoulin et al. [2016] use conditional instance normalization that learns normalization parameters for each style. Given feature activations of the content and style images, Chen and Schmidt [2016] replace content features with the closest-matching style features patch-by-patch. Chen et al. [2017] present a network that learns a set of new filters for every new style. Li et al. [2017a] also adapt a single feed-forward network via a texture controller module which forces the network towards synthesizing the desired style only. We note that the existing feed-forward approaches have to compromise between the generalization [Li et al., 2017a; Huang and Belongie, 2017; Zhang and Dana, 2017] and quality [Ulyanov et al., 2017, 2016b; Gupta et al., 2017].

11.5 Proposed Method

We aim to infer a photorealistic and identity-preserving face $\widehat{I}_r$ from an unaligned stylized face I_s. For this purpose, we design our IFRP framework which contains a Style Removal Network (SRN) and a Discriminative Network (DN). We encourage our SRN to recover faces that come from the latent space of real faces. The DN is trained to distinguish recovered faces from real ones. The general architecture of our IFRP framework is depicted in Fig. 11.2.

11.5.1 Style Removal Network

Since the goal of face recovery is to generate a photorealistic destylized image, a generative network should be able to remove various styles of portraits without losing the identity-preserving information. To this end, we propose our SRN which comprises an autoencoder (a downsampling encoder and an upsampling decoder) and the STN layers. Figure 11.2 shows the architecture of our SRN (enclosed by the blue frame).

The autoencoder learns a deterministic mapping from a portrait space into a latent space with the use of encoder, and a mapping from the latent space to the real face space with the use of decoder. In this manner, the encoder extracts the high-level features of the unaligned stylized faces and projects them into the feature maps of the real face domain while the decoder synthesizes photorealistic faces from the extracted information.

Considering that the input stylized faces are often misaligned, tilted or rotated *etc*, we incorporate four STN layers [Jaderberg et al., 2015] to perform face alignments in a data-driven fashion. The STN layer can estimate the motion parameters of face images and warp them to a canonical view. Figure 11.3 illustrates that a successful alignment can be performed by combining STN layers with out network.

11.5.2 Discriminative Network

Using only a pixel-wise distance between the recovered faces and their ground-truth real counterparts leads to over-smoothed results, as shown in Fig. 11.3(c). To obtain appealing visual results, we introduce a discriminator, which forces recovered faces to reside in the same latent space as real faces. Our proposed DN is composed of convolutional layers and fully connected layers, as illustrated in Fig. 11.2 (the green frame). The discriminative loss, also known as the adversarial loss, penalizes the discrepancy between the distributions of recovered and real faces. This loss is also used to update the parameters of the SRN unit (we alternate over updates of the parameters of SRN and DN). Figure 11.3(d) shows the impact of the adversarial loss on the final results.

11.5.3 Identity Preservation

By using the adversarial loss, our SRN is able to generate high-frequency facial contents. However, the results often lack details of identities such as the beard or wrinkles, as illustrated in Fig. 11.3(d). A possible way to address this issue is to constrain the recovered faces to share as many features as possible with the ground-truth faces.

We construct an identity-preserving loss motivated by the idea of [Gatys et al., 2016b; Johnson et al., 2016]. Specifically, we define an Euclidean distance between the feature representations of the recovered and the ground truth image, respectively. The feature maps are obtained from the ReLU activations of the VGG-19 network [Simonyan and Zisserman, 2014]. Since the VGG network is pre-trained on a very large

(a) (b) (c) (d) (e)

Figure 11.3: Contribution of each component of our IFRP network. (a) Input unaligned portraits from unseen styles. (b) Ground-truth face images. (c) Recovered faces with the ℓ_2 loss. (d) Recovered faces without the identity-preserving loss. (e) Our final results.

image dataset, it can capture visually meaningful facial features. Hence, we can preserve the identity information by encouraging the feature similarity between the generated and ground-truth faces. We combine the pixel-wise loss, the adversarial loss and the identity-preserving loss together as our final loss function to train our network. Figure 11.3(e) illustrates that, with the help of the identity-preserving loss, our IFRP network can reconstruct satisfying identity-preserving results.

11.5.4 Training Details

To train our IFRP network in an end-to-end fashion, we require a large number of SF/RF training image pairs. For each RF, we synthesize different unaligned SF images from various artistic styles to obtain SF/RF $(\boldsymbol{I}_s, \boldsymbol{I}_r)$ training pairs. As described in Section 11.6, we only use stylized faces from three distinct styles in the training stage.

Our goal is to train a feed-forward network SRN to produce an aligned photorealistic face from any given unaligned portrait. To achieve this, we force the recovered face $\widehat{\boldsymbol{I}}_r$ to be similar to its ground-truth counterpart $\boldsymbol{I}_r$. Denote $\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s)$ as the output of our SRN. Since the STN layers are interwoven with the layers of our autoencoder, we optimize the parameters of the autoencoder and the STN layers simultaneously. The pixel-wise loss function $\mathcal{L}_{MSE}$ between $\widehat{\boldsymbol{I}}_r$ and $\boldsymbol{I}_r$ is expressed as:

$$\mathcal{L}_{MSE}(\boldsymbol{\Theta}) = \mathbb{E}_{(\boldsymbol{I}_s,\boldsymbol{I}_r)\sim p(\boldsymbol{I}_s,\boldsymbol{I}_r)} \|\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s) - \boldsymbol{I}_r\|_F^2, \tag{11.1}$$

where $p(\boldsymbol{I}_s, \boldsymbol{I}_r)$ represents the joint distribution of the SF and RF images in the training dataset, and $\boldsymbol{\Theta}$ denotes the parameters of the SRN unit.

To obtain convincing identity-preserving results, we propose an identity-preserving loss to be the Euclidean distance between the features of recovered face $\widehat{\boldsymbol{I}}_r = \mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s)$

and ground-truth face $\boldsymbol{I}_r$. The identity-preserving loss $\mathcal{L}_{id}$ is written as follows:

$$\mathcal{L}_{id}(\boldsymbol{\Theta}) = \mathbb{E}_{(\boldsymbol{I}_s,\boldsymbol{I}_r)\sim p(\boldsymbol{I}_s,\boldsymbol{I}_r)} \|\boldsymbol{\psi}(\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s)) - \boldsymbol{\psi}(\boldsymbol{I}_r)\|_F^2, \tag{11.2}$$

where $\boldsymbol{\psi}(.)$ denotes the extracted feature maps from the layer ReLU3-2 of the VGG-19 model with respect to some input image.

Motivated by the idea of [Goodfellow et al., 2014; Denton et al., 2015; Radford et al., 2015], we aim to make the discriminative network $\mathcal{D}_{\boldsymbol{\Phi}}$ fail to distinguish recovered faces from real ones. Therefore, the parameters of the discriminator $\boldsymbol{\Phi}$ are updated by minimizing $\mathcal{L}_{dis}$, expressed as:

$$\mathcal{L}_{dis}(\boldsymbol{\Phi}) = -\mathbb{E}_{\boldsymbol{I}_r\sim p(\boldsymbol{I}_r)}[\log \mathcal{D}_{\boldsymbol{\Phi}}(\boldsymbol{I}_r)] - \mathbb{E}_{\widehat{\boldsymbol{I}}_r\sim p(\widehat{\boldsymbol{I}}_r)}[\log(1-\mathcal{D}_{\boldsymbol{\Phi}}(\widehat{\boldsymbol{I}}_r))], \tag{11.3}$$

where $p(\boldsymbol{I}_r)$ and $p(\widehat{\boldsymbol{I}}_r)$ indicate the distributions of real and recovered faces respectively, and $\mathcal{D}_{\boldsymbol{\Phi}}(\boldsymbol{I}_r)$ and $\mathcal{D}_{\boldsymbol{\Phi}}(\widehat{\boldsymbol{I}}_r)$ are the outputs of $\mathcal{D}_{\boldsymbol{\Phi}}$. The $\mathcal{L}_{dis}$ loss is also back-propagated with respect to the parameters $\boldsymbol{\Theta}$ of the SRN unit.

Our SNR loss is a weighted sum of three terms: the pixel-wise loss, the adversarial loss, and the identity-preserving loss. The parameters $\boldsymbol{\Theta}$ are obtained by minimizing the objective function of the SRN loss as follows:

$$\begin{aligned}\mathcal{L}_{SNR}(\boldsymbol{\Theta}) =& \mathbb{E}_{(\boldsymbol{I}_s,\boldsymbol{I}_r)\sim p(\boldsymbol{I}_s,\boldsymbol{I}_r)} \|\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s) - \boldsymbol{I}_r\|_F^2 \\ &+\lambda\, \mathbb{E}_{\boldsymbol{I}_s\sim p(\boldsymbol{I}_s))}[\log \mathcal{D}_{\boldsymbol{\Phi}}(\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s))] \\ &+\eta\, \mathbb{E}_{(\boldsymbol{I}_s,\boldsymbol{I}_r)\sim p(\boldsymbol{I}_s,\boldsymbol{I}_r)} \|\boldsymbol{\psi}(\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_s)) - \boldsymbol{\psi}(\boldsymbol{I}_r)\|_F^2\end{aligned} \tag{11.4}$$

where λ and η are trade-off parameters for the discriminator and the identity-preserving losses respectively, and $p(\boldsymbol{I}_s)$ is the distribution of stylized faces.

Since both $\mathcal{G}_{\boldsymbol{\Theta}}(\cdot)$ and $\mathcal{D}_{\boldsymbol{\Phi}}(\cdot)$ are differentiable functions, the error can be back-propagated w.r.t. $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$ by the use of the Stochastic Gradient Descent (SGD) combined with Root Mean Square Propagation (RMSprop) [Hinton, 2012], which helps our algorithm to converge faster.

11.5.5 Implementation Details

The batch normalization procedure is applied after our convolutional and deconvolutional layers except for the last deconvolutional layer, similar to the models described in [Goodfellow et al., 2014; Radford et al., 2015]. We also use leaky rectifier with piece-wise linear units (leakyReLU [Maas et al.]) and the negative slope equal 0.2 as the non-linear activation function. Our network is trained with a mini-batch size of 64. In all our experiments, the parameters λ and η are set to 10^{-2} and 10^{-3}. We also set the learning rate to 10^{-3} and the decay rate to 10^{-2}.

As the iterations progress, the images of output faces will be more similar to the ground-truth. Hence, we gradually reduce the effect of the discriminative network by decreasing λ. Thus, $\lambda^n = \max\{\lambda \cdot 0.995^n, \lambda/2\}$, where n is the epoch index. The strategy of decreasing λ not only enriches the effect of the pixel-level similar-

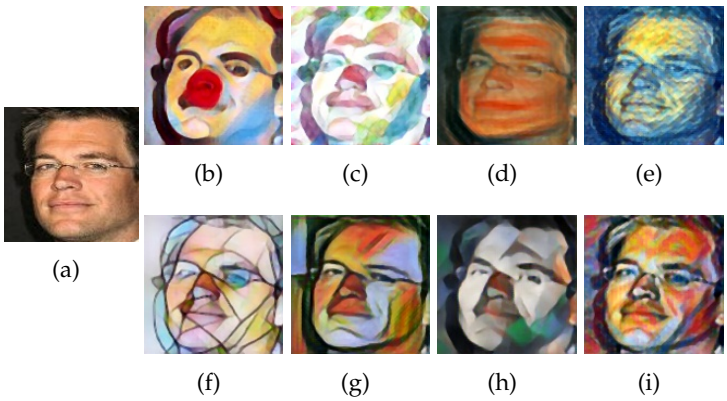


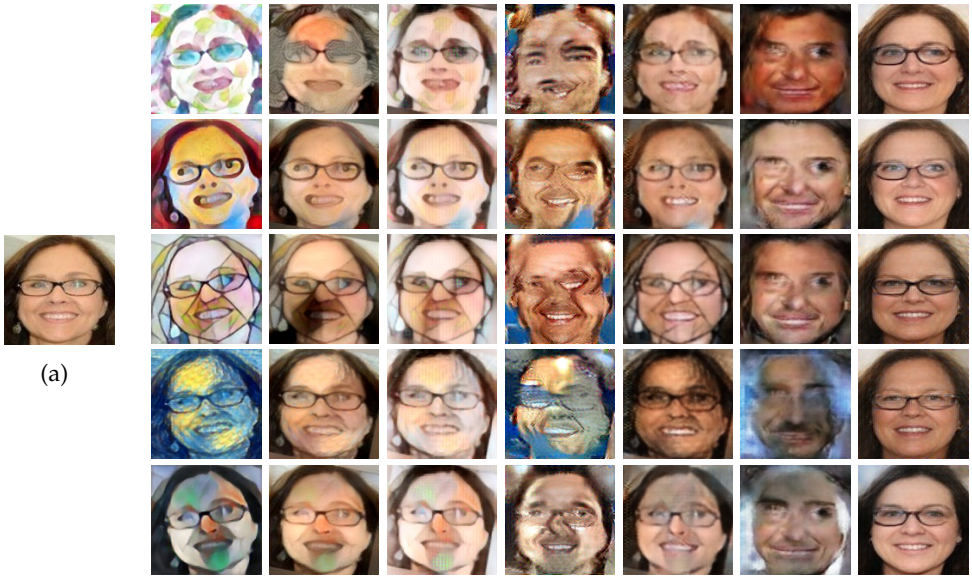
Figure 11.4: Samples of the synthesized dataset. (a) The ground-truth aligned real face image. (b)-(d) The synthesized portraits form *Candy*, *Feathers* and *Scream* which have been used for training our network. (e)-(i) The synthesized portraits form *Starry*, *Mosaic*, *la Muse*, *Udnie* and *Composition VII* styles which have not been used for training.

ity but also keeps the discriminative information in the SRN during training. We also decrease η to reduce the impact of the identity-preserving constraint after each iteration: $\eta^n = \max\{\eta \cdot 0.995^n, \eta/2\}$.

As our method is feed-forward and no optimization is required at the test time, it takes 10 ms to destylize a 128×128 image. We plan to release the dataset and the code.

11.6 Synthesized Dataset and Preprocessing

To train our IFRP network and avoid overfitting, a large number of SF/RF image pairs are required. To generate a dataset of such pairs, we employ the CelebA dataset [Liu et al., 2015]. We first randomly choose 10K aligned real faces from the CelebA dataset for training and 1K images for testing. We use these images as our RF ground-truth faces $\boldsymbol{I}_r$ which are aligned by eyes. The original size of the images is 178×218 pixels. We crop the central part of each image and resize it to 128×128 pixels. Second, we apply affine transformations to the aligned real faces to generate in-plane unaligned faces. To synthesize our training dataset, we retrain the "fast style transfer" network [Johnson et al., 2016] for three different artworks *Scream*, *Candy* and *Feathers* separately. Note that recovering photorealistic faces from *Candy*, *Feathers* and *Scream* styles is more challenging compared to other styles, because facial details are distorted and over-smoothed during the stylization process, as shown in Fig 11.4. Finally, we obtain 30K SF/RF training pairs. We also use 1K unaligned real faces to generate 8K SF images from eight diverse styles (*Starry Night*, *la Muse*, *Composition VII*, *Scream*, *Candy*, *Feathers*, *Mosaic* and *Udnie*) as our testing dataset. There is no overlap between the training and testing datasets.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 11.5: Comparisons of the state-of-the-art methods. (a) The ground-truth real face. (b) Input portraits (from the test dataset) including the seen styles *Feathers* and *Candy* as well as the unseen styles *Mosaic, Starry* and *Udnie*. (c) The method of Gatys et al. [2016b]. (d) The method of Johnson et al. [2016]. (e) The method of Li and Wand [2016b] (MGAN). (f) The method of Isola et al. [2016] (pix2pix). (g) The method of Zhu et al. [2017] (CycleGAN). (h) Our method.

11.7 Experiments

Below, we compare our approach qualitatively and quantitatively to the state-of-the-art methods. To the best of our knowledge, there are no methods which are designed to recover photorealistic faces from portraits. To conduct a fair comparison, we retrain the approaches [Gatys et al., 2016b; Johnson et al., 2016; Li and Wand, 2016b; Isola et al., 2016; Zhu et al., 2017] on our training dataset for the task of destylization.

11.7.1 Qualitative Evaluation

We visually compare our approach against five methods detailed below. To let them achieve their best performance, we align SF images in the test dataset (via STN network).

The method of Gatys et al. [2016b] is an image-optimization based style transfer method which does not have any training stage. This method captures the correlation between feature maps of the portrait and the synthesized face (Gram matrices) in different layers of a CNN. Therefore, spatial structures of face images cannot be preserved. As shown in Fig. 11.5(c) and Fig. 11.6(c), the network fails to produce realistic results and the artistic styles have not been fully removed.

We retrain the approach proposed by Johnson et al. [2016] for destylization. Due to the use of the Gram matrix, their network also generates distorted facial details and produces unnatural effects. As shown in Fig. 11.5(d) and Fig. 11.6(d), the facial details are blurred and the skin colors are not homogeneous. As shown in the first row of Fig. 11.6(d), we observe that the styles of the eyes were not removed from outputs.

MGAN [Li and Wand, 2016b] is a patch-based style transfer method. We retrain this network for the purpose of the face recovery. As this method is trained on RF/SF patches, it cannot capture the global structure of entire faces. As seen in Fig. 11.5(e) and Fig. 11.6(e), this method produces distorted results and the facial colors are inconsistent. In contrast, our method successfully captures the global structure of faces and generates highly-consistent facial colors.

Isola et al. [2016] train a "U-net" generator augmented with a PatchGAN discriminator in an adversarial framework, known as "pix2pix". Since the patch-based discriminator is trained to classify whether an image patch is sampled from real faces or not, this network does not take the global structure of faces into account. In addition, the U-net concatenates low-level features from the bottom layers of the encoder with the features in the decoder to generate face images. Because the low-level features of input images are passed to the outputs, this network fails to eliminate the artistic styles in the face images. As shown in Fig. 11.5(f) and Fig. 11.6(f), although pix2pix can generate acceptable results for the seen styles, it fails to remove the unseen styles and produces obvious artifacts.

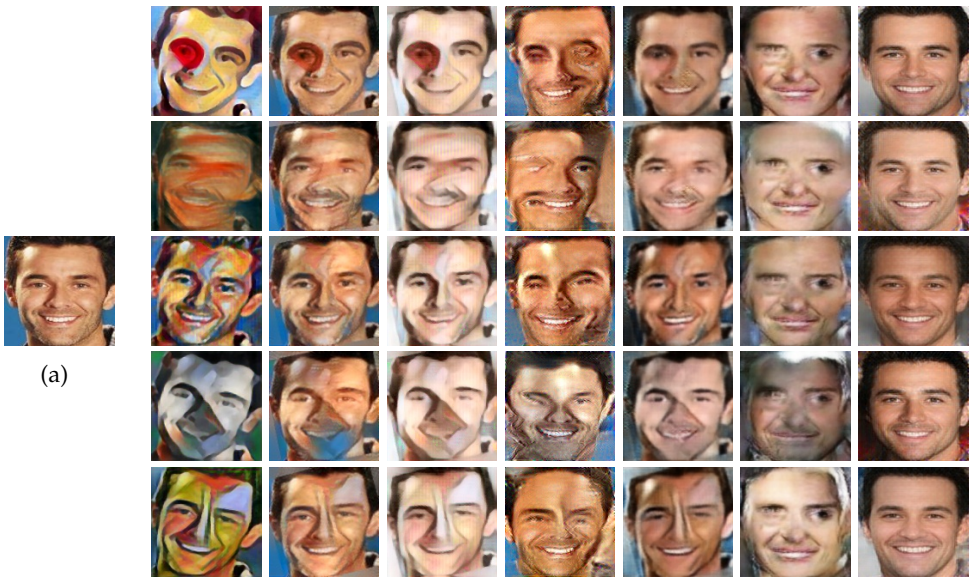
CycleGAN [Zhu et al., 2017] is an image-to-image translation method that uses unpaired datasets. This network provides a mapping between two different domains by the use of a cycle-consistency loss. Since CycleGAN also employs a patch-based discriminator, this network cannot capture the global structure of faces. As this network uses unpaired face datasets *i.e.*, unpaired RF and SF images, the low-level features of the stylized faces and real faces are uncorrelated. Thus, CycleGAN is not suitable for transferring stylized portraits to photorealistic ones. As shown in Fig. 11.5(g) and Fig. 11.6(g), this method produces distorted results and does not preserve the identities with respect to the input images.

In contrast, our results demonstrate higher fidelity and better consistency with respect to the real faces, such as facial expressions and skin colors. Our network can preserve identity information of a subject for both seen and unseen styles, as shown in Fig. 11.5(h) and Fig. 11.6(h).

11.7.2 Quantitative Evaluation

Pixel-wise Recovery Analysis:

To evaluate the pixel-wise recovery performance, we use the average Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) [Wang et al., 2004] scores on seen and unseen styles of our test dataset. The pixel-wise recovery results for each method are summarized in Tab. 11.1 (higher scores indicate better results). The PSNR and SSIM scores confirm that our IFRP approach outperforms other state-of-the-art



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 11.6: (a) The ground-truth real face. (b) Input portraits (from the test dataset) including the seen styles *Candy* and*Scream* as well as the unseen styles *Composition VII, Udnie* and *la Muse* from unseen styles. (c) The method of Gatys et al. [2016b]. (d) The method of Johnson et al. [2016]. (e) The method of Li and Wand [2016b] (MGAN). (f) The method of Isola et al. [2016] (pix2pix). (g) The method of Zhu et al. [2017] (CycleGAN). (h) Our method.

methods on both seen (the first and second rows) and unseen (the third, fourth and fifth rows) styles. Figure 11.5 and Fig. 11.6 verify the performance visually. Moreover, we also apply different methods on sketches from the CUFSF dataset as an unseen style without fine-tuning or re-training our network.

In order to demonstrate the contributions of each loss function to the quantitative results, we also show the results for when only the ℓ_2 loss is used, as indicated by SRN in Tab. 11.1, and for both the ℓ_2 and discriminative losses, as indicated by SRN+DN in Tab. 11.1. The ℓ_2 loss considers the intensity similarity only, thus it produces over-smooth faces. The discriminative loss further forces the generated faces to be realistic, thus it improves the final results qualitatively and quantitatively. Benefiting from our combined loss, our network not only achieves highest quantitative results but also generates photorealistic face images.

Face Retrieval Analysis:

In this section, we demonstrate that the faces recovered by our method are highly consistent with their ground-truth counterparts. To this end, we run a face recognition algorithm [Parkhi et al., 2015] on our test dataset for both seen and unseen styles. For each investigated method, we set 1K recovered faces from one style as a query dataset and then set 1K of ground-truth faces as a search dataset. We apply the method [Parkhi et al., 2015] to quantify whether the correct person is retrieved

Table 11.1: Comparisons of PSNR and SSIM on the entire test dataset.

Method	Seen Styles		Unseen Styles		Unseen Sketches	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Gatys [Gatys et al., 2016b]	23.88	0.84	23.25	0.83	23.33	0.82
Johnson [Johnson et al., 2016]	19.65	0.82	19.81	0.81	19.77	0.82
MGAN [Li and Wand, 2016b]	20.87	0.79	20.21	0.66	21.01	0.71
pix2pix [Isola et al., 2016]	25.28	0.89	23.10	0.85	23.88	0.86
CycleGAN [Zhu et al., 2017]	19.58	0.78	18.99	0.77	19.60	0.77
SRN	25.12	0.89	24.09	0.88	24.13	0.89
SRN + DN	25.25	0.90	24.25	0.89	24.56	0.90
IFRP	**27.08**	**0.93**	**24.83**	**0.91**	**24.89**	**0.92**

within the top-5 matched images. Then an average retrieval score is obtained. We repeat this procedure for every style and then obtain the average Face Retrieval Ratio (FRR) by averaging all scores from the seen and unseen styles, respectively. As indicated in Tab. 11.2, our IFRP network outperforms the other methods across all the styles. Even for the unseen styles, our method can still retain most identity features, making the destylized results similar to the ground-truth faces. Moreover, we also run an experiment on hand-drawn sketches of the CUFSF dataset used as an unseen style. The FRR scores are better compared to results on other styles as facial components are easier to extract from sketches/their contours. Despite our method is not dedicated to face retrieval, we compare it to the approach [Zhang et al., 2011]. To challenge our method, we did not re-train our network on sketches (we used other styles). Thus, we recovered faces from sketches (CUFSF dataset) and performed face identification that yielded ∼91% Verification Rate (VR) FAR=0.1%. This outperforms photo-synthesizing method [Zhang et al., 2011] (43.66% VR at FAR=0.1%) which uses sketches for training.

Consistency Analysis w.r.t. Styles:

As shown in Fig. 11.5(h) and Fig. 11.6(h), our network recovers the photorealistic faces from various stylized portraits of the same person. Note that recovered faces resemble each other. It indicates that our network is robust to different styles.

In order to demonstrate the robustness of our network to different styles quantitatively, we study the consistency of faces recovered from different styles. Here, we choose 1K faces destylized from one style. For each destylized face we search its top-5 most similar faces in another group of destylized faces. If the same person is retrieved within the top-5 candidates, we record it as a hit. Then an average hit number of one style is obtained. We repeat the same procedure for all the other 7 styles, and then calculate the average hit number, denoted as Face Consistency Ratio (FCR). Note that the probability of one hit by chance is 0.5%. Table 11.2 shows the average FCR scores on the test dataset for each method. The FCR scores indicate

Table 11.2: Comparisons of FRR and FCR on the entire test dataset.

Method	FRR			FCR
	Seen Styles	Unseen Styles	Unseen Sketch	
Gatys	64.67%	60.28%	68.36%	72.89%
Johnson	50.54%	38.87%	40.27%	44.99%
MGAN	6.97%	12.52%	17.99%	38.24%
pix2pix	75.13%	59.98%	61.63%	87.73%
CycleGAN	1.07%	0.68%	0.70%	13.32%
IFRP	**86.93%**	**74.52%**	**91.05%**	**92.06%**

Figure 11.7: Results for the original unaligned paintings. Top row: the original portraits from art galleries. Bottom row: our results.

that our IFRP method produces the most consistent destylized faces across different styles. This also implies that our SRN can extract facial features irrespective of image styles.

11.7.3 Destylizing Original Paintings and Sketches

We demonstrate that our method is not restricted to recovery of faces from computer-generated stylized portraits but it can also deal with real paintings and sketches. To confirm this, we randomly choose a few of paintings from art galleries such as Archibald [arc, 2017] and hand-drawn sketches from FERET dataset [Phillips et al., 1998]. Next, we crop face regions from them as our real test images. Figure 11.7 and Fig. 11.8 show that our method can efficiently recover photorealistic faces. This indicates that our method is not limited to the synthesized data and does not require an alignment procedure beforehand.

11.7.4 Limitations

We note that in the CelebA dataset, numbers of images of children, old people and young adults are unbalanced *e.g.*, there are more images of young adults than children and old people. This makes our synthesized dataset unbalanced. Hence, facial fea-

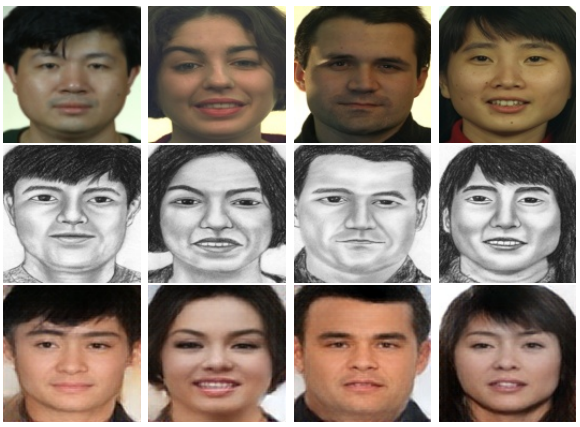


Figure 11.8: Recovering photo-realistic faces from hand-drawn sketches from the FERET dataset. Top row: ground-truth faces. Middle row: sketches. Bottom row: our results.

Figure 11.9: Limitations. Top row: ground-truth faces. Middle row: unaligned stylized faces. Bottom row: our results.

tures of children and old people is are not fully represented in our dataset. Therefore, our network may be prone to recover images with facial features of young adults for children and old people, as seen in Fig. 11.9. In addition, because the color information has been distorted in the stylized paintings, it is very challenging to recover the skin and hair color that is consistent with the ground-truth without introducing additional cues. In future, we intend to embed semantic information into our network and then generate more consistent face images in terms of the skin and hair color.

11.8 Conclusion

We introduce a novel neural network for face recovery. It extracts features from a given unaligned stylized portrait and then recovers a photorealistic face from these features. The SRN successfully learns a mapping from unaligned stylized faces to

aligned photorealistic faces. Moreover, our identity-preserving loss further encourages our network to generate identity trustworthy faces. This makes our algorithm readily available for tasks such as face recognition. We also show that our approach can recover latent faces of portraits in unseen styles, real paintings and sketches.

Chapter 12

Recovering Faces from Portraits with Auxiliary Facial Attributes

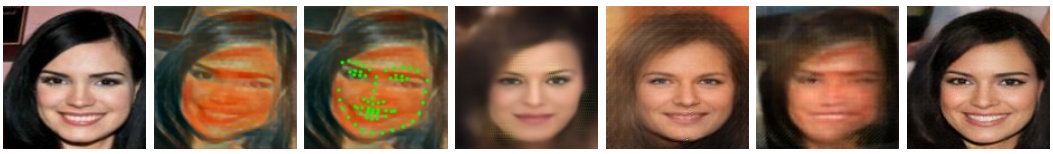
12.1 Foreword

In chapter 11, we aim at recovering realistic faces from stylized faces while preserveing indentity information. Since there are some facial details have been distorted in the stylized portraits, such as skin and hair colors, it is difficult to hallucinate those missing details as well as make them consistent with the ground-truth ones. Therefore, we exploit the high-level semantic information to facilitate face destylization, inspired by our work presented in chapter 8. Note that, different from chapter 8, where skin and hair colors can be deduced directly from input images, in this work we use facial attributes including color information to restore realistic face images since the original colors of the portraits, *e.g.*, hair colors, may be totally altered. In this manner, we can significantly reduce the ambiguity of the mapping between the stylized portraits and the recovered faces and thus achieve authentically photorealistic faces much closer to the ground-truth ones.

This chapter has been accpted as a conference paper: Fatemeh Shiri, Xin Yu, Richard Hartley, Fatih Porikli, Piotr Koniusz: Recovering Faces from Portraits with Auxiliary Facial Attributes. In *IEEE Winter Conference on Application of Computer Vision (WACV), 2019*.

12.2 Abstract

Recovering a photorealistic face from an artistic portrait is a challenging task since crucial facial details are often distorted or completely lost in artistic compositions. To handle this loss and contamination of information, here we propose an Attribute-guided Face Recovery from Portraits (AFRP) method that utilizes a Face Recovery Network (FRN) and a Discriminative Network (DN). FRN consists of an autoencoder with residual block-embedded skip-connections and incorporates facial attribute vectors into the feature maps of input portraits at the bottleneck of the autoencoder. DN has multiple convolutional and fully-connected layers, and it is conditioned to enforce FRN to generate authentic face images with corresponding facial attributes that



(a) (b) (c) (d) (e) (f) (g)

Figure 12.1: Comparisons to the state-of-the-art methods. (a) Ground-truth face image (from test dataset; not used in the training). (b) Unaligned stylized portraits of (a) from *Scream* style (unseen style in training), respectively. (c) Detected landmarks by the approach of Zhang et al. [2014]. (d) Results obtained by the approach of Shiri et al. [2017]. (e) Results obtained by the approach of Shiri et al. [2018]. (f) Results obtained by the approach of Isola et al. [2016] (pix2pix). (g) Our results.

are specified by the input attribute vectors. Levering on the spatial transformer networks, FRN automatically compensates for misalignments of portraits and generates aligned face images. For the preservation of the identity information, our method imposes the recovered and ground-truth faces to share similar visual features. Specifically, DN determines whether the recovered image looks like a real face as well as the facial attributes extracted from the recovered image are consistent with the given attributes. Our method can recover high-quality photorealistic faces from unaligned portraits while preserving the identity of the face images as well as it can reconstruct a photorealistic face image with a desired set of attributes. It can also recover photorealistic faces from unseen stylized portraits, artistic paintings, and hand-drawn sketches. On large-scale synthesized and sketch datasets, we demonstrate that our face recovery method achieves state-of-the-art results.

12.3 Introduction

Numerous style transfer methods have been proposed to transfer arbitrary artwork styles into content images. Unlike prior research on image stylization, we address a challenging inverse problem called photorealistic face recovery from stylized portraits which aims at recovering a photorealistic face image from a given stylized portrait. The recovery of the latent photorealistic face from its artistic portrait can provide critical information for facial analysis and the digital entertainment industry. Facial details in stylized portraits contain artistic effects and distortions such as profile edges and texture changes as shown in Fig. 12.1(b). These artistic effects result in a partial loss of facial details and identity-related information. Moreover, stylized face images may contain various facial expressions, facial distortions and misalignments. Off-the-shelf facial landmark detectors often fail to localize facial landmarks correctly as shown in Fig. 12.1(c). Therefore, restoring high-quality photorealistic faces from unaligned stylized artistic portraits is a challenging problem yet has numerous useful applications.

Motivated by such challenges, recovery of photorealistic images from portraits has recently received some attention [Shiri et al., 2017, 2018; Isola et al., 2016; Zhu et al., 2017]. The existing methods [Shiri et al., 2017, 2018; Isola et al., 2016; Zhu et al., 2017] take portrait images as inputs and then utilize a simple autoencoder to generate a photorealistic face image. These methods do not utilize the valuable semantic information available during the face recovery process. Despite of being trained on large-scale datasets, they fail to provide consistent mappings between Stylized Portraits (SP) and ground-truth Real Faces (RF). Thus, they cannot preserve or enforce desired facial attributes in the recovered images. As shown in Fig. 12.1(d), Fig. 12.1(e) and Fig. 12.1(f), the facial details recovered by the state-of-the-art methods [Shiri et al., 2017, 2018; Isola et al., 2016] are semantically and perceptually inconsistent with the ground-truth images. Inaccuracies range from an unnatural blur to attribute mismatches which include (but not limited to) the examples of *Black Hair* and *Open Mouth*.

Unlike previous works, we propose to utilize facial attributes as high-level semantic information to boost the visual performance of the recovered face images. We note that simply embedding the binary facial attribute vector as an additional input channel to the network results in visible distortions (see Fig. 12.4(e)).We observe that only low-frequency facial components are visible in the stylized input faces as a residual image (the difference between the RF image and the recovered face image) contains the missing high-frequency details. Therefore, in order to recover the high-frequency facial details, we propose to incorporate the auxiliary facial attribute information into the residual features.

Based on our observations above, we present a novel Face Recovery Network (FRN) that can embed facial attributes into the process of face recovery. Our FRN employs an autoencoder with residual block-embedded skip connections to incorporate visual features obtained from portraits as well as semantic cues provided by facial attributes. FRN progressively upsamples the concatenated feature maps through its deconvolutional layers. Moreover, we employ a discriminative network that examines whether a recovered face image resembles an authentic face image and whether the attributes extracted from the recovered face are consistent with the input attributes. As a result, our discriminative network can guide the generative network to incorporate the semantic information into the recovery process. As shown in Fig. 12.1(g), our network can learn more consistent mappings between SP and RF facial patterns and preserve low-frequency details unchanged. This allows us to generate realistic face images which include details of the ground truth faces (e.g. *Black Hair, Smiling, Straight Hair, Wearing lip stick, pink cheeks*), as shown in Fig. 12.1(g). Although the attributes are normalized between 0 and 1 during training, they can be further scaled up to manipulate the final results during the testing stage according to the users' needs.

In order to train our network, we require a large number of pairs of Stylized Portraits (SP) and Real Face (RF). For this purpose, we need to synthesize a large-scale training dataset. However, the choices of styles are numerous and thus we cannot generate all possible stylized faces for training. Thus, we need to select dis-

tinctive styles for training. To this end, we use a style-distance metric to measure the distinctiveness of styles. Since Gram matrices can capture the style information in images [Gatys et al., 2016b], we measure the similarity of styles by the Log-Euclidean distance of Gram matrices [Jayasumana et al., 2013]. Specifically, we first measure the distance between Gram matrices of stylized images and the average Gram matrix of real faces, and then select the most distinctive styles, *i.e.*largest distance, for training. Furthermore, we note that our CNN filters learned from the data of seen styles (used for the training phase) can also extract features from images belonging to unseen styles. Thus, the facial information of unseen stylized portraits can be extracted and used to generate realistic faces, as later demonstrated in our experiments. The main contributions of our work can be summarized as follows:

- We design a novel framework to automatically remove styles from unaligned stylized portraits. Our framework encodes stylized images with facial attributes and then recovers realistic faces from the encoded feature maps.
- We propose an autoencoder with residual block-embedded skip-connections to extract residual feature maps from SP inputs and combine the extracted feature maps with facial attributes. In this fashion, we fuse visual and semantic information to attain high-quality visual performance.
- By manipulating input attribute vectors, our network can also generate the realistic faces towards desired attributes.
- We propose a style-distance metric to measure the most distinct styles for the training purpose. Thus, our network achieves better generalization for other unseen styles.

To the best of our knowledge, our method is the first attempt to utilize facial attribute information into realistic face recovery from stylized faces. In what follows, we demonstrate that such an approach reduces the ambiguity in the recovered images.

12.4 Related Work

Below we review neural style transfer methods and deep generative models for image generation which are closely related to our task.

12.4.1 Deep Generative Models

Recently, Generative Adversarial Networks (GANs) [Goodfellow et al., 2014] have led to a significant improvement in image generation tasks, where a generator network attempts to fool a discriminator network that distinguishes real images from generated ones.

Preliminary GANs learn the distribution of the training data in an unconditional setting. Although these methods produce impressive photorealistic images, they

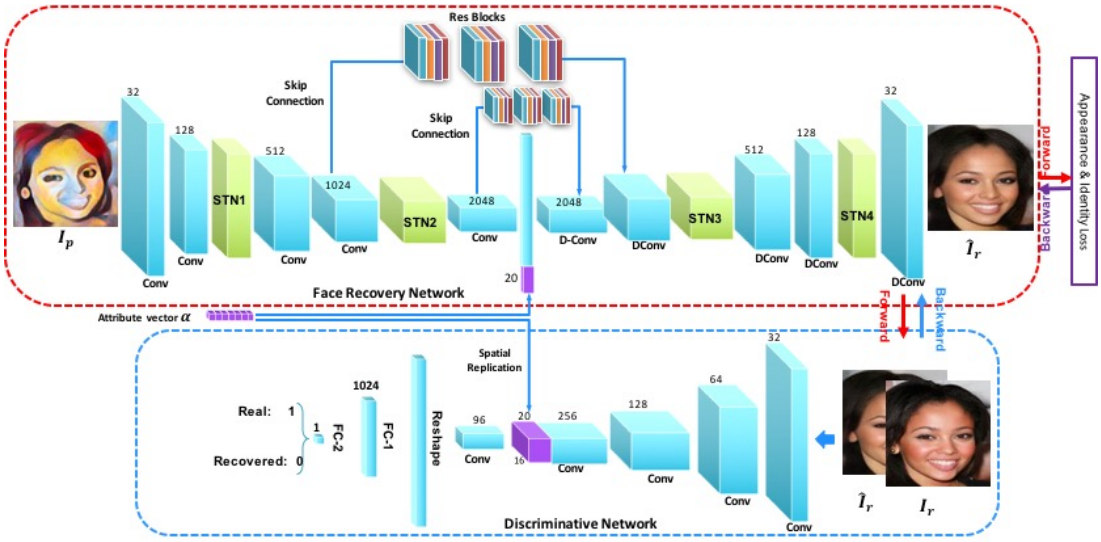


Figure 12.2: The architecture of our attribute-embedded face recovery framework consists of two parts: a generative network (red frame) and a discriminative network (blue frame).

cannot distinguish identities of subjects. Recently, conditional GANs [Isola et al., 2016] are introduced to learn conditional generative models which generate images conditioned on certain input variables. This makes conditional GANs benefit many applications such as super-resolution [Yu and Porikli, 2017a,b; Ledig et al., 2017], image generation [Van Den Oord et al., 2016; Kingma and Welling, 2013; Denton et al., 2015; Zhang et al., 2017a; Shiri et al., 2018], image inpainting [Yeh et al., 2016; Pathak et al., 2016], general purpose image-to-image translation [Isola et al., 2016], image manipulation [Zhu et al., 2016a], and style transfer [Ulyanov et al., 2016a]. In particular, Li and Wand [2016b] train a Markovian GAN for the style transfer; a discriminative training is applied on Markovian neural patches to capture local style statistics. Isola et al. [2016] develop "pix2pix" framework which uses the patch-GAN to transfer low-level features from the input to the output domain. When they are employed to destylized portraits, these patch-based approaches produce visual artifacts and fails to capture the global structure of the faces.

Moreover, there exist several methods which synthesize sketches from photographs (and vice versa) [Nejati and Sim, 2011; Yuen and Man, 2007; Tang and Wang, 2003; Sharma and Jacobs, 2011; Sangkloy et al., 2017]. When compared to sketch-to-face synthesis, viewed as a specific case of face recovery, our unified framework is able to process various more complex styles to recover photo-realistic faces.

Recently, Yan et al. [2016] use a conditional CNN to generate faces based on attributes. Perarnau et al. [2016] develop an invertible conditional GAN to generate new faces by editing facial attributes of input images, while Shen and Liu [2016] manipulate attributes of an input image via its residual image. As their methods are dedicated to generating new face images rather than recovering faces from portraits,

they do not deal with the identity preservation and the quality of the reconstructed faces varies. In contrast, our method utilizes the attribute information to reduce the uncertainty of the face recovery process. We focus on a faithful recovery of real faces underlying artistic input portraits.

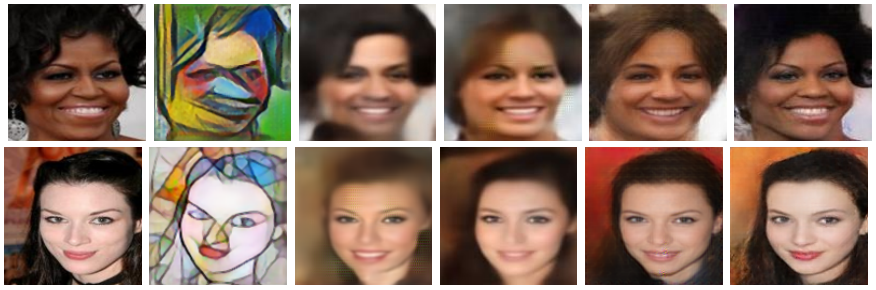
12.4.2 Neural Style Transfer

Style transfer methods aim to synthesize an image that preserves visual contents of the input image and carry characteristics of a chosen style. The seminal work of Gatys et al. [2015] shows that the correlation between feature maps, (*i.e.*, Gram matrix formed on features extracted by a trained deep neural network), has the ability to capture visual styles. Since then, many follow-up works synthesized stylized images by minimizing Gram-related objectives, such as iterative optimization [Gatys et al., 2016b, 2017; Li and Wand, 2016a; Risser et al., 2017] and feed-forward networks [Ulyanov et al., 2016a; Johnson et al., 2016; Li and Wand, 2016b]. Iterative optimization methods are computationally inefficient due to the optimization step required at the testing stage. In contrast, feed-forward methods learn the transformation network which performs stylization in feed-forward manner.

Johnson et al. [2016] train a generative network for a fast style transfer using perceptual loss functions. The architecture of their generator network follows the work [Radford et al., 2015] and also uses residual blocks. Another concurrent work [Ulyanov et al., 2016a], named Texture Network, employs a multi-resolution architecture in the generator network. Ulyanov et al. [2017] replace the spatial batch normalization with the instance normalization to achieve a faster convergence. Wang et al. [2017] enhance the granularity of the feed-forward style transfer with a multimodal CNN which performs stylization hierarchically via multiple losses deployed across multiple scales.

These feed-forward methods are limited by their requirement of training one network per style due to the lack of generalization in network design. To deal with such a restriction, a number of recent approaches encode multiple styles within a single feed-forward network [Dumoulin et al., 2016; Chen et al., 2017; Li et al., 2017a,b]. Dumoulin et al. [2016] use conditional instance normalization to learn necessary normalization parameters for each style. Given feature activations of the content and style images, the approach of Chen and Schmidt [2016] replaces content features with the closest-matching style features patch-by-patch. To achieve an arbitrary style transfer, Chen et al. [2017] propose to swap the content feature with the closest style feature locally. Li et al. [2017a] adapt a single feed-forward network via a texture controller module which forces the network towards synthesizing the desired style only.

As observed in the previous work [Shiri et al., 2018], direct use of neural style transfer to the task of face recovery is suboptimal. Even though recent works [Shiri et al., 2017, 2018] are designed to destylize portrait images, they tend to distort facial details and cannot recover facial traits (*e.g.*, hair color, lipstick, open/closed lips) which match the ground-truth well. Since some facial traits, such as hair colors,



(a) (b) (c) (d) (e) (f)

Figure 12.3: Contribution of each loss function in AFRP network. (a) Ground-truth face images. (b) Input unaligned portraits from unseen styles. (c) Recovered faces without utilizing DN and identity-preserving loss. (d) Recovered faces with the ℓ_2 loss and discriminative loss. (e) Recovered faces with the ℓ_2 loss, discriminative loss and identity-preserving loss. (f) Our final results by embedding facial attributes.

are difficult to be inferred, only employing the pixel-wise ℓ_2 norm and perceptual losses does not yield correct facial attributes. Thus, state-of-the-art face destylization methods produce ambiguous results.

12.5 Proposed Method

Below we present an attribute-guided framework for face recovery that takes SP images and facial attribute vectors as inputs and outputs photorealistic images of faces.

12.5.1 Network Architecture

The entire network consists of two parts: a Face Recover Network (FRN) and a Discriminative Network (DN). FRN is composed of an autoencoder as well as skip connections with residual blocks. FRN extracts residual feature maps from input portraits and concatenates the corresponding 20-dimensional attribute vector with the extracted residual feature vector at the bottleneck of the autoencoder and then upsamples it. In this manner, we fuse visual and semantic information to attain high-quality visual performance. The role of DN is to enforce the input attributes and the recovered face images to be similar to their real counterparts. The attribute vector is replicated and then concatenated with the extracted feature maps of the convolutional layer of DN. The entire architecture of our network is illustrated in Fig. 12.2.

12.5.1.1 Face Recover Network

This module employs a deep fully convolutional autoencoder for face restoration from portraits (as shown in the red frame of Fig. 12.2). The convolutional layers of

the encoder capture the feature maps of input portraits and deconvolutional layers of the decoder upsample the feature maps to recover the facial details. Previous works [Shiri et al., 2017, 2018; Isola et al., 2016; Zhu et al., 2017] take stylized portraits as inputs to recover the underlying faces. However, they do not make use of valuable semantic information during face recovery. Unlike the previous works, our FRN incorporates low-level visual and high-level semantic information (*i.e.*facial attributes) for face recovery to reduce the ambiguity of mappings between SP and RF images. Specifically, at the bottleneck of the autoencoder, the attribute vector is concatenated with the residual feature vector as indicated by the purple blocks in Fig. 12.2. Simply embedding a semantic vector into SP inputs may increase the ambiguity. As shown in Fig. 12.4(e), if we encode input portraits with attributes instead of residual feature maps, the mapping between the recovered faces and the ground-truth suffers distortions, *i.e.*the identity has been changed.

We also symmetrically link top convolutional and deconvolutional layers via skip-layer connections [Long et al., 2015]. These skip connections pass higher-resolution visual details of portraits from convolutional to deconvolutional layers, which lead to a better restoration performance. Moreover, each skip-connection comprises three residual blocks. Due to usage of the residual blocks, our network can remove the styles of input portraits while increasing accuracy as shown in Fig. 12.4(g) while the network without the skip-connections tends to output a blurry face image as shown in Fig. 12.4(c).

Note that input portraits are misaligned (*i.e.*, in-plane rotations, translations). Similar to the work [Shiri et al., 2018], we use multiple spatial transformer networks (STNs) [Jaderberg et al., 2015] in FRN, as shown in the green blocks in Fig. 12.2. These intermediate STN layers compensate for misalignments of the input portraits. Thus, our method does not require the use of facial landmarks or 3D face models (often used for face alignment).

To constrain the appearance similarity between the recovered faces and their RF ground-truth counterparts, we exploit a pixel-wise ℓ_2 loss and an identity-preserving loss [Shiri et al., 2018]. The pixel-wise ℓ_2 loss enforces intensity-based similarity between images of recovered faces and their ground-truth images. The autoencoder supervised by the ℓ_2 loss tends to output over-smoothed results as shown in Fig. 12.3(c). For the identity-preserving loss, we use FaceNet [Schroff et al., 2015] to extract features from images (see Sec. 12.5.2 for more details), and then we compare Euclidean distance between features of two images. In this way, we encourage the feature similarity between the recovered faces and their ground-truth counterparts. Without the identity-preserving loss, the network produces random artifacts that resemble facial details, such as wrinkles, as shown in Fig. 12.3(d).

12.5.1.2 Discriminative Network

In order to force the FRN to encode facial attribute information, we employ a conditional discriminative network. In particular, the discriminative network is designed to distinguish whether the attributes of face images recovered by FRN match the

(a) (b) (c) (d) (e) (f) (g)

Figure 12.4: Ablation study of our network architecture. (a) RF ground-truth image. (b) Unaligned input portrait. (c) Result without using skip connections/residual blocks in the autoencoder. (d) Result without using residual blocks in the autoencoder. (e) Result when the attribute vector is concatenated with the SF input directly. (f) Result without using the attribute embedding. A standard discriminative network is used, similar to the discriminative network in [Shiri et al., 2018]. (g) Our final result.

desired attributes. Moreover, DN promotes the recovered images to be similar to RF images. Since our FRN network may learn to ignore attribute vectors, *e.g.*, the weights corresponding to the semantic information are all zeros, we design a discriminator network that enforces semantic attribute information into the generative process. As shown in the first row of Fig. 12.4(f), the recovered hair color in the image is brown even if the ground truth hair color is black. This implies that the attribute information is not exploited by the network. Therefore, we design the discriminative network which promotes attribute embedding into the learning process.

As shown in the blue frame of Fig. 12.2, DN consists of convolutional and fully connected layers. The real and recovered faces are fed into the network. The attribute information is fed into the middle layer of the network as conditional information. As CNN filters in the first layers extract low-level features and filters in the higher layers extract semantically-meaningful image patterns [Zeiler and Fergus, 2014], in our experiment concatenating features maps with the attribute vectors on the fourth convolutional layer in DN yields better empirical results. When there is a mismatch between the extracted features and the input attributes, the discriminative network will pass the errors to the FRN network during backpropagation. With the help of the discriminative network, the attribute information can be embedded into the FRN network. As shown in Fig. 12.4(g), our final result matches the ground-truth facial expression, age and gender.

12.5.2 Training Procedure

To train our AFRP network in an end-to-end fashion, we construct SP, RF and attribute vector triplets $(\boldsymbol{I}_p, \boldsymbol{I}_r, \boldsymbol{a})$ as our training dataset, where $\boldsymbol{I}_r$ is the aligned real face image, and $\boldsymbol{I}_p$ is the corresponding synthesized unaligned portrait image. For each RF, we synthesize different unaligned SP images from various artistic styles to obtain SP/RF training pairs. As described in Sec. 12.6, we only use stylized portraits from three distinct styles for training. We use SP image $\boldsymbol{I}_p$ and its ground-truth at-

tribute label vector $\boldsymbol{a}$ as inputs and the corresponding RF ground-truth image $\boldsymbol{I}_r$ as a target in the training stage.

We train our FRN network using a pixel-wise ℓ_2 loss, a feature-wise loss and an adversarial loss to enforce the generated face $\boldsymbol{I}_r$ to resemble its corresponding ground-truth. In addition, we employ a binary cross-entropy loss to update our discriminative network. Since the STN layers are interwoven with the layers of our autoencoder, we optimize the parameters of the autoencoder and the STN layers simultaneously. Below we explain each loss individually.

Pixel-wise Intensity Similarity Loss: We aim to train a feed-forward network to produce an aligned photorealistic face from any given unaligned portrait. To achieve this, we feed our FRN with $\boldsymbol{I}_p$ images and their corresponding attributes $\boldsymbol{a}$ as inputs and then force the recovered face $\widehat{\boldsymbol{I}}_r$ to be similar to its ground-truth counterpart $\boldsymbol{I}_r$ in the intensity-wise sense. Hence, we minimize the objective function $\mathcal{L}_{pix}$:

$$\mathcal{L}_{pix}(\boldsymbol{\Theta}) = \mathbb{E}\|\widehat{\boldsymbol{I}}_r - \boldsymbol{I}_r\|_F^2 = \mathbb{E}_{(\boldsymbol{I}_p,\boldsymbol{I}_r,\boldsymbol{a})\sim p(\boldsymbol{I}_p,\boldsymbol{I}_r,\boldsymbol{a})}\|\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p,\boldsymbol{a}) - \boldsymbol{I}_r\|_F^2, \tag{12.1}$$

where $\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p,\boldsymbol{a})$ and $\boldsymbol{\Theta}$ represent the output and parameters of our FRN, respectively. We denote $p(\boldsymbol{I}_p,\boldsymbol{I}_r,\boldsymbol{a})$ as the joint distribution of the SP and RF images and the corresponding attributes in the training dataset.

Identity-preserving Loss: To obtain faithful identity-preserving results, we extract feature maps from the ReLU activations of the FaceNet. Then we compute the Euclidean distance between the features of recovered face $\widehat{\boldsymbol{I}}_r = \mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p,\boldsymbol{a})$ and ground-truth face $\boldsymbol{I}_r$. As the FaceNet network is pre-trained on a very large image dataset, it has the ability to capture visually meaningful facial features. Hence, we can preserve the identity information by encouraging the feature similarity between the generated and ground-truth faces. The identity-preserving loss $\mathcal{L}_{id}$ is written as follows:

$$\mathcal{L}_{id}(\boldsymbol{\Theta}) = \mathbb{E}\|\boldsymbol{\psi}(\widehat{\boldsymbol{I}}_r) - \boldsymbol{\psi}(\boldsymbol{I}_r)\|_F^2 = \mathbb{E}_{(\boldsymbol{I}_p,\boldsymbol{I}_r,\boldsymbol{a})\sim p(\boldsymbol{I}_s,\boldsymbol{I}_r,\boldsymbol{a})}\|\boldsymbol{\psi}(\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p,\boldsymbol{a})) - \boldsymbol{\psi}(\boldsymbol{I}_r)\|_F^2, \tag{12.2}$$

where $\boldsymbol{\psi}(.)$ denotes the feature maps extracted from the layer ReLU3-2 of the FaceNet.

Discriminative Loss: Similar to [Yan et al., 2016; Zhang et al., 2017b], our goal is to make the discriminative network to tell if recovered faces contain the desired attributes or not but fail to distinguish recovered faces from real ones. In the meanwhile, FRN should make the discriminative network $\mathcal{D}_{\boldsymbol{\Phi}}$ fail to distinguish recovered faces from real ones and the attributes of generated faces should match the input attributes. Hence, in order to train the discriminative network, we take real FR face images $\boldsymbol{I}_r$ and their corresponding ground-truth attributes $\boldsymbol{a}$ as positive sample pairs $(\boldsymbol{I}_r,\boldsymbol{a})$. Negative samples are constructed from recovered faces $\widehat{\boldsymbol{I}}_r$ and their ground-truth attributes $\boldsymbol{a}$ as well as real FR faces and mismatched (fake) attributes $\tilde{\boldsymbol{a}}$. Therefore, the negative sample pairs consist of both $(\widehat{\boldsymbol{I}}_r,\boldsymbol{a})$ and $(\boldsymbol{I}_r,\tilde{\boldsymbol{a}})$. The parameters of the discriminator $\boldsymbol{\Phi}$ are updated by minimizing $\mathcal{L}_{dis}$, expressed as:

$$\begin{aligned}\mathcal{L}_{dis}(\boldsymbol{\Phi}) = &-\mathbb{E}_{(\boldsymbol{I}_r,\boldsymbol{a})\sim p(\boldsymbol{I}_r,\boldsymbol{a})}[\log \mathcal{D}_{\boldsymbol{\Phi}}(\boldsymbol{I}_r,\boldsymbol{a})] - \mathbb{E}_{(\widehat{\boldsymbol{I}}_r,\boldsymbol{a})\sim p(\widehat{\boldsymbol{I}}_r,\boldsymbol{a})}[\log(1-\mathcal{D}_{\boldsymbol{\Phi}}(\widehat{\boldsymbol{I}}_r,\boldsymbol{a}))] \\ &-\mathbb{E}_{(\boldsymbol{I}_r,\tilde{\boldsymbol{a}})\sim p(\boldsymbol{I}_r,\tilde{\boldsymbol{a}})}[log(1-\mathcal{D}_{\boldsymbol{\Phi}}(\boldsymbol{I}_r,\tilde{\boldsymbol{a}}))],\end{aligned} \tag{12.3}$$

where $p(\boldsymbol{I}_r, \boldsymbol{a})$, $p(\widehat{\boldsymbol{I}}_r, \boldsymbol{a})$ and $p(\boldsymbol{I}_r, \tilde{\boldsymbol{a}})$ indicate the distributions of real and recovered faces and the corresponding attributes respectively, and $p(\boldsymbol{I}_r, \tilde{\boldsymbol{a}})$ represents the distributions of the recovered faces and the corresponding mismatched (fake) attributes. $\mathcal{D}_{\boldsymbol{\Phi}}(\boldsymbol{I}_r, \boldsymbol{a})$, $\mathcal{D}_{\boldsymbol{\Phi}}(\widehat{\boldsymbol{I}}_r, \boldsymbol{a})$ and $\mathcal{D}_{\boldsymbol{\Phi}}(\boldsymbol{I}_r, \tilde{\boldsymbol{a}})$ are the outputs of $\mathcal{D}_{\boldsymbol{\Phi}}$. We first update the parameters of the discriminative network, and $\mathcal{L}_{dis}$ loss is also back-propagated to FRN.

Our FNR loss is a weighted sum of three terms: the pixel-wise loss, the discriminative loss, and the identity-preserving loss. The parameters $\boldsymbol{\Theta}$ are obtained by minimizing the objective function of the FRN loss as follows:

$$\begin{aligned}\mathcal{L}_{FNR}(\boldsymbol{\Theta}) =& \mathbb{E}_{(\boldsymbol{I}_p, \boldsymbol{I}_r, \boldsymbol{a}) \sim p(\boldsymbol{I}_p, \boldsymbol{I}_r, \boldsymbol{a})} \|\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p) - \boldsymbol{I}_r\|_F^2 \\ &+\lambda\ \mathbb{E}_{\boldsymbol{I}_p \sim p(\boldsymbol{I}_p, \boldsymbol{a}))} [\log \mathcal{D}_{\boldsymbol{\Phi}}(\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p, \boldsymbol{a}), \boldsymbol{a})] \\ &+\eta\ \mathbb{E}_{(\boldsymbol{I}_p, \boldsymbol{I}_r, \boldsymbol{a}) \sim p(\boldsymbol{I}_p, \boldsymbol{I}_r, \boldsymbol{a})} \|\boldsymbol{\psi}(\mathcal{G}_{\boldsymbol{\Theta}}(\boldsymbol{I}_p, \boldsymbol{a})) - \boldsymbol{\psi}(\boldsymbol{I}_r)\|_F^2,\end{aligned} \tag{12.4}$$

where λ determines a trade-off between the appearance and the attribute similarity, and η determines a trade-off between the image intensity and the feature similarity.

As $\mathcal{G}_{\boldsymbol{\Theta}}(\cdot)$ and $\mathcal{D}_{\boldsymbol{\Phi}}(\cdot)$ are differentiable, we apply back-propagation with respect to $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$, and optimize via the Stochastic Gradient Descent (SGD) combined with Root Mean Square Propagation (RMSprop).

12.5.3 Implementation Details

The discriminative network DN is only required in the training phase. In the testing phase, we take SP portraits and their corresponding attribute vectors as inputs and feed them to FRN. The outputs of FRN are the recovered photo-realistic face images. Although the attributes used for training are normalized between 0 and 1, they can be scaled up and down, *e.g.*above 1 or below 0, to manipulate the final results according to the users' demand.

We employ convolutional layers with kernels of size 4×4 and stride 2 in the encoder and deconvolutional layers with kernels of size 4×4 and stride 2 in the decoder. The feature maps in our encoder are passed to the decoder by skip connections. Our network is trained with a mini-batch size of 64 with the learning rate set to 10^{-3} and the decay rate set to 10^{-2}. For STNs, we also use the same architectures as in [Shiri et al., 2018] to align feature maps. In all our experiments, the parameters λ and η are set to 10^{-2} and 10^{-3}, respectively, gradually reducing λ by a factor 0.995 to emphasize the importance of the appearance similarity. However, to guarantee the attributes to be embedded in the training phase, we cease decreasing λ when it is lower than 0.005. As our method is feed-forward and no optimization is required at the test time, it takes 8 ms to destylize a 128×128 image.

12.6 Dataset and Preprocessing

To train our AFRP network and avoid overfitting, a large number of SP/RF image pairs are required. We use the CelebA dataset [Liu et al., 2015] to generate our

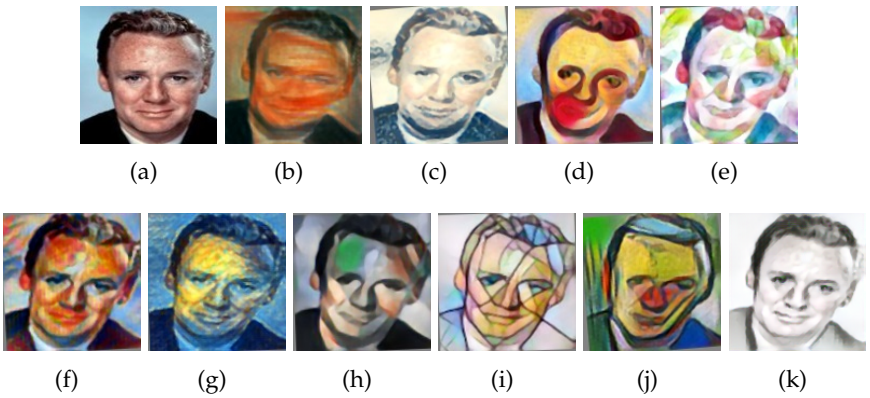


Figure 12.5: Samples of our synthesized dataset. (a) The ground-truth aligned real face image. (b)-(k) The synthesized unaligned portraits form *Wave*, *Scream*, *Candy*, *Feathers*, *Composition VII*, *Starry night*, *Udnie*, *Mosaic*,*la Muse* and *Sketch* styles which have been used for training and testing our network.

training data. First, we randomly select 110K real faces from the CelebA dataset for training and 2K images for testing. Second, we crop the central part of each image and resize it to 128×128 pixels as our RF ground-truth face images $\boldsymbol{I}_r$. Then, we transform RF images including rotation and translation to obtain unaligned faces. Besides, we only use three distinct styles for synthesizing our training dataset and the selection criterion for the styles will be explained in Sec. 12.6.1. Finally, we obtain 330K SP/RF pairs and their corresponding attributes for training. We also use 2K unaligned real faces to synthesize 20K SP images from 10 diverse styles as our testing dataset. Those 10 different sytles used for training and testing are shown in Fig. 12.5. Furthermore, we also add sketches as an unseen style to our test dataset. There is no overlap between the training and testing datasets.

We choose 20 dominant attributes (*Bald*, *Bangs*, *Big nose*, *Black Hair*, *Blond Hair*, *Brown Hair*, *Eyeglasses*, *Gray Hair*, *Heavy Makeup*, *Male*, *Mouth Open*, *Mustache*, *Narrow Eyes*, *No Beard*, *Pale Skin*, *Smiling*, *Straight Hair*, *Wavy Hair*, *Wearing Lipstick* and *Young*) from 40 attributes in CelebA and the ground truth attributes are binary 0/1 values.

12.6.1 Style Distance Metric

It is not practical to generate a large number of possible styles for training, and thus we propose a style distance metric to select the most difficult styles for the face recovery process. To this end, we compute Gram matrices for various styles from feature maps of pre-trained VGG-network [Simonyan and Zisserman, 2014]. Then, we measure the similarity of styles based on the Log-Euclidean metric [Jayasumana et al., 2013] between Gram matrices of style images and the average Gram matrix of all real faces in our training dataset. Here, we choose *Candy*, *Wave* and *Mosaic* styles for training as their Gram matrices have larger distances to the average Gram matrix of real faces among all the available styles.

Table 12.1: Impact of tuning attributes on the classification results.

Attributes	GT Attr. Acc.	Increased Attr. Acc.	Decreased Attr. Acc.
Young	95%	100%	0.5%
Male	100%	100%	1%
Beard	79%	100%	15%

12.7 Experiments

We compare our approach qualitatively and quantitatively to the state-of-the-art methods [Johnson et al., 2016; Shiri et al., 2017; Isola et al., 2016; Zhu et al., 2017; Shiri et al., 2018]. To conduct a fair comparison, we retrain these approaches on our training dataset for the task of photorealistic face recovery from stylized portraits.

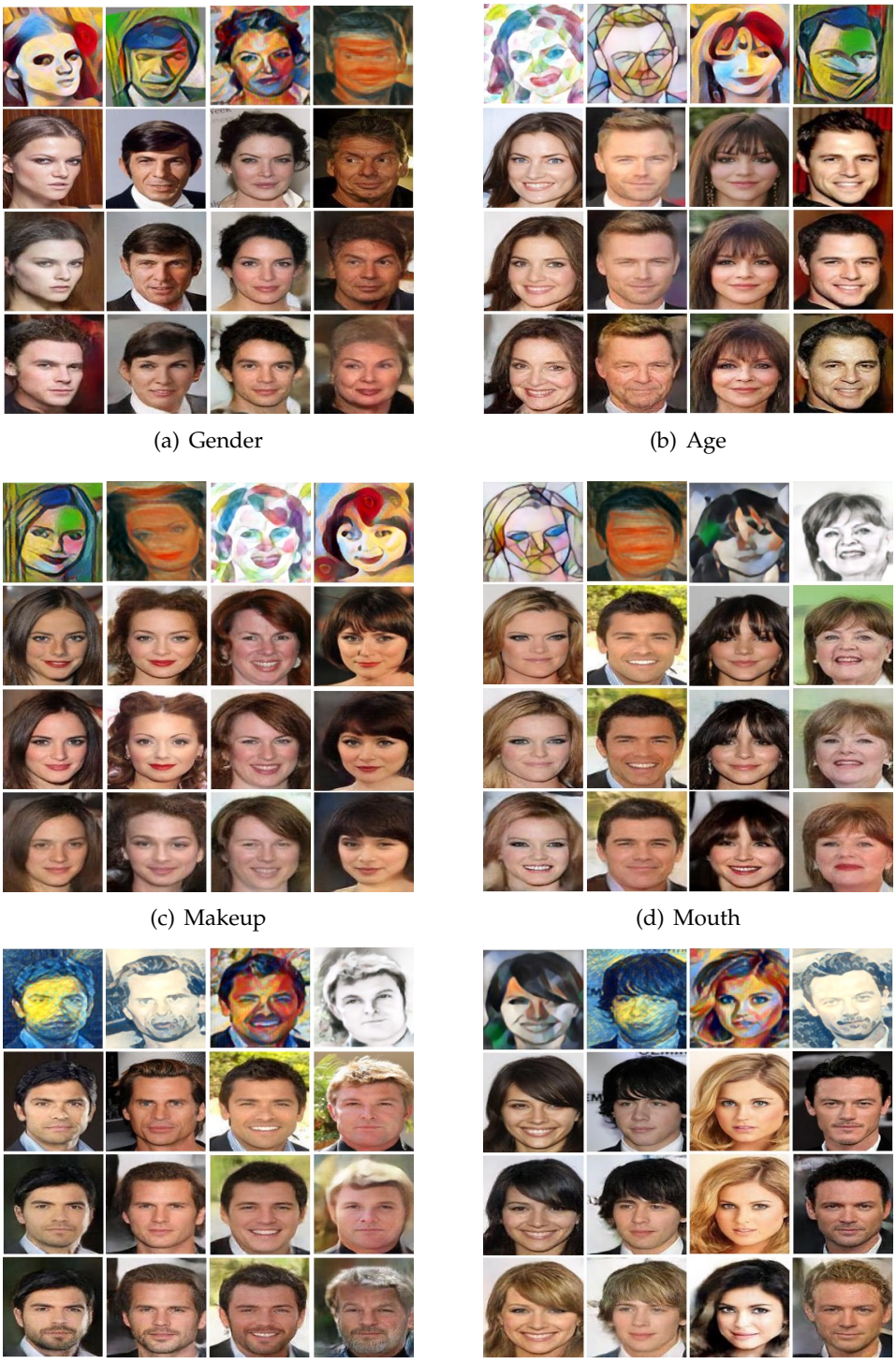
12.7.1 Attribute Manipulation in Face Recovery

Given an SP portrait, previous methods based on deep neural networks [Shiri et al., 2017, 2018; Isola et al., 2016; Zhu et al., 2017; Johnson et al., 2016] produce an arbitrary photorealistic face image. Those methods cannot output desired attributes in the final results. In contrast, our method generates authentic face images which share similar attributes to the ground-truths. Furthermore, by manipulating the attribute vectors, our method can also post-edit the recovered results. As shown in Fig. 12.6(f), by changing the hair color attribute, we can restore a face image of the same person with different hair colors. Our method can manipulate the age of the recovered faces, *i.e.*, adding more wrinkles and age spots by changing the *Young* attribute, as seen in Fig. 12.6(b). In addition, our network can remove the eye-lines and lipstick in Fig. 12.6(c), open or close mouths in Fig. 12.6(d), add beard in Fig. 12.6(e), as well as change the hair color in Fig. 12.6(f).

Moreover, to test whether the attribute information has been successfully embedded in our network, we choose three different attributes, *i.e.Young*, *Male* and *Beard*, and we train an attribute classifier for each attribute. By increasing and decreasing the corresponding attribute values, the true positive accuracy is changed accordingly, as illustrated in Tab. 12.1. This indicates that the attribute information has been successfully embedded in our network. Therefore, we significantly increase the flexibility of our method and successfully inject semantic information into the recovery process.

12.7.2 Qualitative Evaluation

For qualitative evaluations, we provide sample results in Fig. 12.7. Note that Shiri et al. [2017]; Isola et al. [2016]; Johnson et al. [2016] and Zhu et al. [2017] require input SP faces to be aligned before recovery. For a fair comparison, we employ an

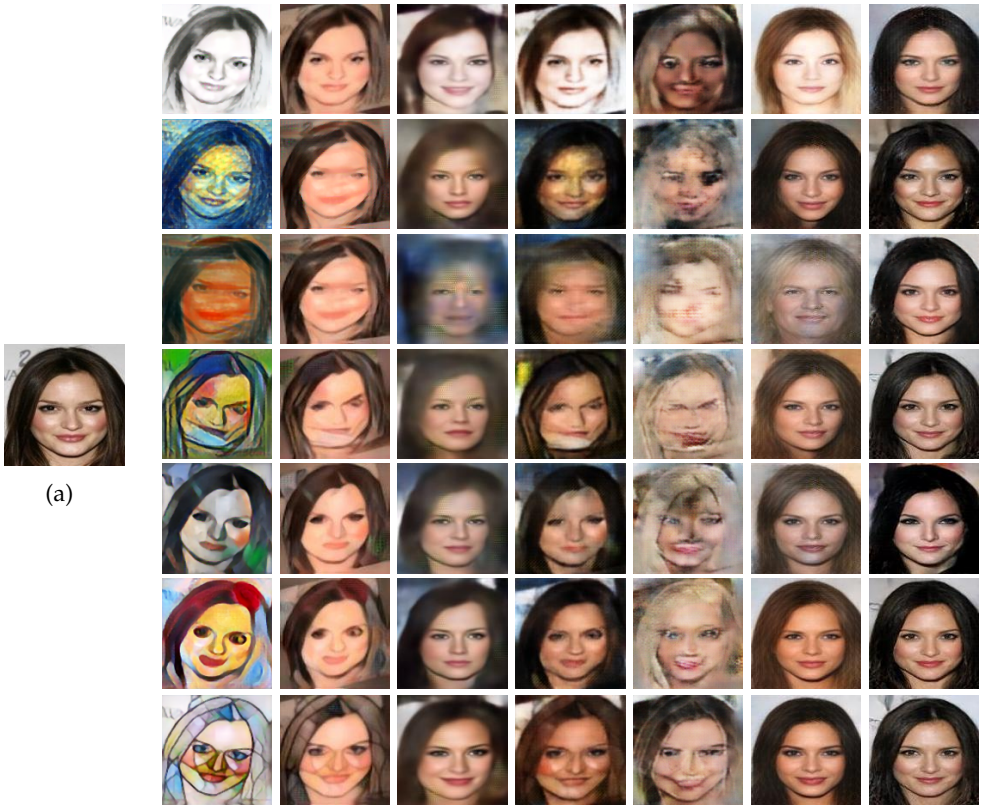


(a) Gender (b) Age

(c) Makeup (d) Mouth

(e) Beard (f) Hair Color

Figure 12.6: Our method lets us fine-tune the recovered results by manipulating the attributes. First row: Unaligned input portraits. Second row: RF ground-truth faces. Third row: Our results with ground-truth attributes. Fourth row: Our results by adjusting attributes. (a) Changing gender. (b) Adding age. (c) Removing makeup. (d) Opening/ closing mouth. (e) Adding beard. (d) Changing hair color.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 12.7: Comparisons to the state-of-the-art methods. (a) The original RF image. (b) Input portraits (from the test dataset) including the unseen styles *Sketch*, *Starry*, *Scream*, *La Muse* and *Udnie* as well as the seen styles *Candy* and *Mosaic*. (c) Results of Johnson et al. [2016]. (d) Results of Shiri et al. [2017] (e) Results of Isola et al. [2016] (pix2pix). (f) Results of Zhu et al. [2017] (CycleGAN). (g) Results of Shiri et al. [2018]. (h) Our results.

STN to align all the SP images. Our method and the method of Shiri et al. [2018] automatically generate upright real face images. The aligned upright RF ground-truth images are shown for comparison. We visually compare our approach against five methods detailed below.

Johnson et al. [2016] propose a feed-forward style transfer method. We retrain this approach for destylization. This method captures the correlation between feature maps of the portrait and the synthesized face (Gram matrices) in different layers of a CNN, but fails to preserve spatial structures of face images. Thus, their network generates distorted facial details and produces unnatural artefacts. As shown in Fig. 12.7(c), the facial details are blurred and the artistic styles have not been fully removed.

Shiri et al. [2017] introduce a face destylization method which only uses a pixel-wise loss in their generative network and a standard discriminator to enhance facial

details. Even though it is trained on a large-scale dataset, It fails to generate authentic facial details due to the existence of various styles. As seen in Fig. 12.7(d), this method produces distorted results and the facial colors are inconsistent. It cannot recover faces from unaligned portraits or large pose portraits either.

Isola et al. [2016] train a "U-net" generator augmented with a PatchGAN discriminator in an adversarial framework, known as "pix2pix". Since the patch-based discriminator is trained to classify whether an image patch is sampled from real faces or not, this network does not take the global structure of faces into account. As shown in Fig. 12.7(e), although pix2pix can generate acceptable results for the seen styles, it fails to remove the unseen styles and produces obvious artifacts.

CycleGAN [Zhu et al., 2017] is an image-to-image translation method that uses unpaired datasets. This network provides a mapping between two different domains by the use of a cycle-consistency loss. Since CycleGAN also employs a patch-based discriminator, it cannot capture the global structure of faces either. As CycleGAN uses unpaired face datasets, the low-level features of the stylized faces and real faces do not match correctly. As shown in Fig. 12.7(f), this method produces distorted results and does not preserve the identities with respect to the input images.

Shiri et al. [2018] exploit an identity-preserving loss to reveal the photorealistic faces from unaligned stylized faces while keeping the identity of the face image. They also employ a simple autoencoder and standard discriminative network to recover the real faces, but their discriminative network is only used to force the generative network to produce sharper results without imposing attribute information. As visible in Fig. 12.7(g), their method suffers mismatched hair colors. As shown in the third rows of Fig. 12.7(g), their method also recovers male facial details.

In contrast, our results demonstrate higher fidelity and better consistency with respect to the ground-truth face images as shown in Fig. 12.7(h). We evaluate on portraits from seen/unseen styles and sketches, and our method produces high-quality realistic faces which also match the semantic composition of ground-truth images. In addition, our network recovers the photorealistic faces from various stylized portraits of the same person as shown in Fig. 12.7. Note that the recovered faces resemble each other. This demonstrates the robustness of our network with respect to different styles. Also, thanks to our proposed style-distance metric, we can select more difficult styles to train our network, which also facilitates the generalization ability of our network.

12.7.3 Quantitative Evaluation

Face Reconstruction Analysis. To evaluate the reconstruction performance, we measure the average Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) [Wang et al., 2004] scores on the entire test dataset. Table 12.2 indicates that our method achieves superior quantitative performance in comparison to other methods on both seen and unseen styles. As indicated in Tab. 12.2, we show the quantitative results of solely using FRN, marked as FRN. Also, the results of using both FRN and a standard DN, indicated by FRN+SDN, is demonstrated in Tab. 12.2.

Table 12.2: Comparisons of PSNR and SSIM on the entire test dataset.

Method	Seen Styles		Unseen Styles		Unseen Sketches	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
[Johnson et al., 2016]	17.85	0.76	18.07	0.75	18.11	0.76
[Shiri et al., 2017]	19.22	0.81	19.09	0.80	19.01	0.80
[Isola et al., 2016]	18.45	0.78	18.12	0.77	18.07	0.78
[Zhu et al., 2017]	18.35	0.75	18.29	0.75	18.08	0.76
[Shiri et al., 2018]	19.35	0.80	19.31	0.79	19.045	0.80
FRN	18.42	0.76	18.51	0.75	18.49	0.75
FRN + SDN	19.66	0.81	19.58	0.80	19.62	0.80
AFRP	**20.01**	**0.84**	**19.99**	**0.83**	**19.98**	**0.83**

The standard DN only forces FRN to generate realistic faces, and thus it improves the results qualitatively and quantitatively. Since FRN augmented with attributes may learn a trivial solution, where all attribute vectors will be neglected, using a standard DN cannot force FRN to embed such attribute information. On the contrary, our conditional DN is able to distinguish whether the attributes match the input faces or not, thus forcing FRN to embed attribute information in the process of face recovery. In this manner, the ambiguity is significantly reduced and the network achieves better performance.

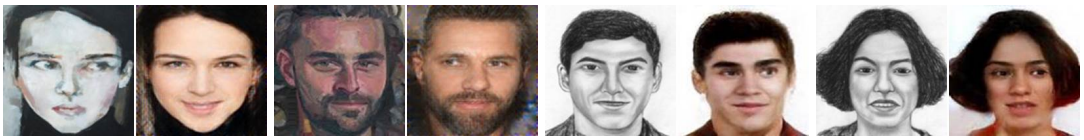
Face Retrieval Analysis. To demonstrate that the faces recovered by our method are highly consistent with their ground-truth counterparts, we run a face recognition algorithm [Parkhi et al., 2015] on our test dataset for both seen and unseen styles. For each investigated method, we consider 2K recovered faces from one style as query images and then use their ground-truth real faces as a gallery dataset. We run the method of Parkhi et al. [2015] to check whether the correct person is retrieved within the top-5 matched images and then an average retrieval score is obtained. We repeat this procedure for each style and then obtain the average Face Retrieval Ratio (FRR) by averaging all scores from the seen and unseen styles, respectively. As indicated in Tab. 12.3, our AFRP network outperforms the other methods across all the styles. Even for the unseen styles, our method can still generate realistic facial details in high fidelity to the ground-truths.

12.7.4 Destylizing Original Paintings and Sketches

Figure 12.8 illustrates that our method is not limited to computer-generated stylized portraits and it can also efficiently recover photorealistic faces from original paintings and sketches. We choose real paintings from art galleries and hand-drawn sketches as our test examples. Since we do not know the ground-truth attributes, we set the attribute vectors to neutral values, *i.e.*, 0.5. As shown in Fig. 12.8, even though

Table 12.3: Comparisons of FRR on the entire test dataset.

Method	Seen Styles	Unseen Styles	Unseen Sketch
[Johnson et al., 2016]	55.57%	50.48%	54.36%
[Shiri et al., 2017]	78.00%	66.89%	65.26%
[Isola et al., 2016]	76.03%	62.67%	64.64%
[Zhu et al., 2017]	36.07%	33.68%	32.75%
[Shiri et al., 2018]	84.51%	75.32%	75.44%
AFRP	**93.08%**	**83.14%**	**92.05%**



(a) (b) (c) (d)

Figure 12.8: Results for the original unaligned paintings and hand-drawn sketches. Right: the original portraits. Left: our results.

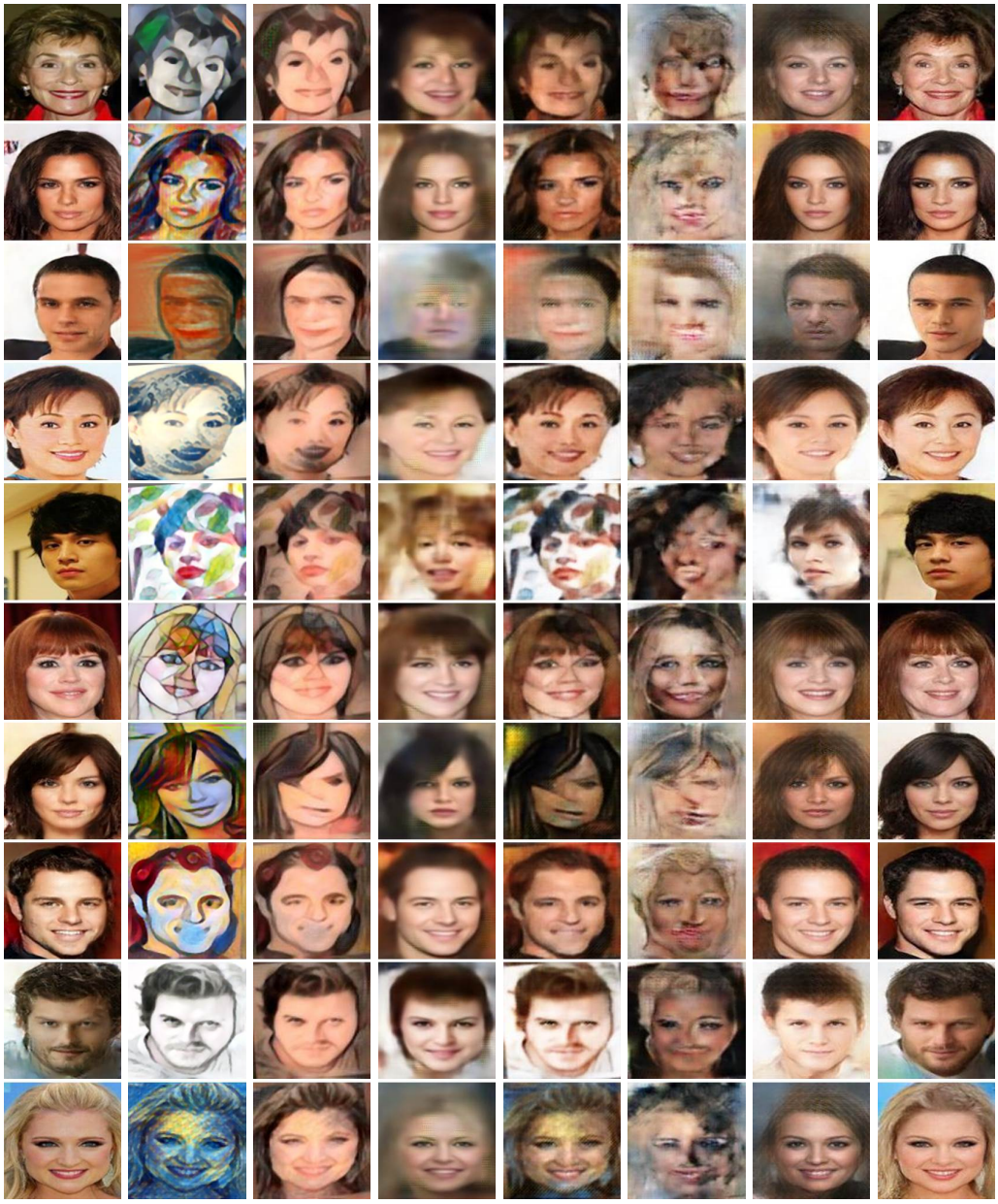
the attributes may be inaccurate, our method is still able to generate authentic face images regardless of their original styles.

12.8 Conclusion

We introduce an attribute guided generative-discriminative network to recover photorealistic faces from unaligned stylized portraits in an end-to-end fashion. With the help of the conditional discriminative network, our network successfully incorporates facial attribute vectors into the residual features of input portraits at the bottleneck of the autoencoder. Our network is able to preserve the identity of generated faces and it can post-edit the recovered results by adjusting the attribute information. Moreover, our algorithm demonstrates good generalization ability for recovery of portraits from unseen styles, real paintings as well as hand-drawn sketches.

12.9 Appendix

In Fig. 12.9, we provide more additional results demonstrating the performance of our AFRP network compared to the state-of-art approaches.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 12.9: Comparisons to the state-of-the-art methods. (a) The original RF images. (b) Input portraits (from the test dataset) including the unseen styles as well as the seen styles. (c) Results of Johnson et al. [2016]. (d) Results of Shiri et al. [2017] (e) Results of Isola et al. [2016] (pix2pix). (f) Results of Zhu et al. [2017] (CycleGAN). (g) Results of Shiri et al. [2018]. (h) Our results.

Chapter 13

Conclusion and Future Work

13.1 Conclusion

This thesis mainly addresses the face hallucination problems including super-resolving high-resolution (HR) faces from their very low-resolution (LR) counterparts and recovering realistic face images from stylized portrait images. The contributions of this thesis are seven aspects: (1) we propose an Ultra-Resolution Discriminative Generative Network (URDGN) as well as a deconvolutional network to super-resolve very LR aligned faces by leveraging large-scale face images, where similar facial patterns are learned and used to generate HR facial details (seeing chapter 2 and 3); (2) we incorporate spatial transformer networks into our upsampling networks to align LR faces in the procedure of super-resolution. In this manner, our method does not require the input LR faces to be aligned beforehand, thus mitigating the artifacts caused by misalignments of LR face images (seeing chapter 4); (3) we develop a decoder-encoder-decoder architecture to super-resolve noisy LR face images, where the first decoder and the encoder are designed to obtain a noise-free LR faces and the second decoder is used to achieve high-quality upsampled HR face images (seeing chapter 5); (4) we propose a multi-scale upsampling network architecture to hallucinate LR face images in different resolutions while preserving all the information in the LR inputs (seeing chapter 6); (5) we present to embed mid-level face structure information and high-level semantic information into the process of face super-resolution instead of only using low-level intensity similarity as a constraint (seeing chapter 7 and 8); (6) we propose a transformative autoencoder to jointly frontalize and super-resolving very LR face images. We also employ a triplet loss function to train the network, which aims to minimize the distances between the projected codes of side faces and their corresponding frontal ones (seeing chapter 9); (7) we develop face destylization methods to recover photorealistic face images from stylized portrait images in terms of appearance similarity and identity similarity (seeing chapter 10, 11 and 12).

To be specific, in chapter 2, we present a new and very capable discriminative generative network to ultra-resolve very small LR face images. Our algorithm can both increase the input LR image size significantly, *i.e.*, $8\times$, and reconstruct much richer facial details. By introducing a pixel-wise ℓ_2 regularization on the generated face images into the framework of URDGN, our method is able to generate authentic

HR faces. Since our method learns an end-to-end mapping between LR and HR face images, it preserves the global structure of faces well. Furthermore, in training, we only assume the locations of eyes to be approximately aligned, which significantly makes the other face datasets more attainable. As an alternative to URDGN, we also present an effective method to super-resolve very small LR face images by exploiting deconvolutional neural networks in chapter 3. We demonstrate that using a single deconvolutional-convolutional network is able to ease the training difficulty of URDGN as well as reduce artifacts caused by the deconvolutional layers and the discriminative networks in URDGN. However, a post-processing step is required to enhance the visual quality of the upsampled faces since only using an ℓ_2 loss tends to produce overly smooth results.

In chapter 4, we develop a Transformative Discriminative Network (TDN) to super-resolve unaligned very LR face images in an end-to-end manner. By incorporating spatial transformer networks into our upsampling network, our network learns how to align faces while upsampling. In this manner, our method does not require the input LR faces to be aligned beforehand and thus alleviates the artifacts caused by the misalignments of LR face images. In chapter 5, we present a Transformative Discriminative AutoEncoder (TDAE) to upsample noisy LR face images while reducing artifacts caused by the noise. Since directly denoising LR faces may corrupt the LR facial patterns, the deteriorated facial patterns will lead to distortions in the upsampled HR faces. Instead of removing noise in LR images, we leverage on a new decoder-encoder-decoder architecture to super-resolve unaligned and noisy very LR face images with a challenging upsampling factor of $8\times$. Our networks jointly align, remove noise, and discriminatively hallucinate input images, thus achieving high-quality upsampled HR face images. In chapter 6, we present a multiscale transformative discriminative network to super-resolve very small LR face images. By designing a two branch input neural network, we can upsample LR images in various resolutions without discarding the residuals of resized input images. In this manner, our method is able to utilize all the information from inputs for face super-resolution.

In chapter 7, we present a novel multi-task upsampling network to super-resolve very small LR face images. We not only employ the image appearance similarity but also exploit the face structure information estimated from LR input images themselves in the super-resolution. In this manner, we preserve the spatial relationships between facial components, thus producing more authentic face images. With the help of our facial component heatmap estimation branch, our method super-resolves faces in different poses and does not suffer from distortions caused by erroneous facial landmark localization in LR inputs. In chapter 8, we introduce an attribute embedded discriminative upsampling network to super-resolve very LR unaligned face images by a large magnification factor (*i.e.*, $8\times$) in an end-to-end fashion. With the help of the conditional discriminative network, we successfully embed facial attribute information into the upsampling network, and thus reduce the inherit ambiguity in super-resolution. After training, our network is not only able to super-resolve LR faces but also able to fine-tune the upsampled results by adjusting the attribute in-

formation. In this way, our network can generate HR face images much closer to their corresponding ground-truth ones, thus achieving superior face hallucination performance.

In chapter 9, we introduce a Transformative Adversarial Nerual Network (TANN) to upsample and frontalize very LR unaligned face images jointly in an end-to-end fashion. Our network learns how to frontalize and align LR faces while upsampling them, *i.e.*, 8×. With the help of our proposed triplet loss, we can enforce the representations of input LR profile faces to be close to the representations of their frontal counterparts and far away from the representations of other frontal faces in the latent subspace. In this way, the frontalized faces are much closer to their corresponding frontal ones since the same upsampling network is used. By exploiting the intra-class discriminative information and the feature constraints, our network generates realistic facial details.

In chapter 10, we present a face destylization method that extracts features of a stylized portrait and then exploits them to generate its corresponding photo-realistic face. Thus, our network learns a mapping from stylized facial feature maps to realistic facial feature maps. Moreover, our network can successfully extract facial features from different styles and thus is able to destylize unseen style portraits as well. In chapter 11, we employ an identity-preserving loss to further encourage our network to generate identity trustworthy faces. Regarding that stylized portraits may be unaligned, spatial transformer networks are incorporated into our style removal network, motivated by the work in chapter 4. Therefore, our style removal network can not only remove various styles from unaligned portraits but also preserve the identity information of the portraits. In chapter 12, we introduce an attribute guided generative-discriminative network to recover photorealistic faces, inspired by the work in chapter 8. Our network successfully incorporates facial attribute vectors into the residual features of input portraits. In this way, our network can not only preserve the identities of the generated faces but also post-edit the recovered results by adjusting the attribute information. Therefore, we significantly increase the flexibility of our style removal network and thus recover realistic face images much closer to the latent face images.

13.2 Future Work

In our previous works, we mainly use bicubic interpolation to downsample ground-truth HR face images and then construct the LR and HR face image pairs for training our networks. However, LR faces may undergo different degradation models. For instance, blur or mosaic effects may appear in the LR face images. Even though our proposed TDAE is able to mitigate some artifacts caused by the blur and mosaic effects, learning the latent degradation models from real LR face images is more desirable. In the future, we aim to not only learn the mapping between the LR faces and their HR counterparts but also learn the degradation process of LR faces by taking different degradation effects into account. Therefore, we can make our face

hallucination networks more robust and practical in real-world applications.

Similar to other supervised deep learning based methods, our neural networks may fail to super-resolve high-quality HR faces when the testing domain, *i.e.*, target domain, is significantly different from the training one, *i.e.*, source domain. For example, our network is trained on a dataset of face photographs, while the test images are sampled from surveillance videos. Even though we can re-train our network on the target domain, transferring our learned network to other domains for face super-resolution is also desirable, especially when there are not sufficiently many training examples in another domain.

For both face super-resolution methods and face destylization methods, we do not design specific components or losses to recover the occluded facial parts if the faces are partially occluded by other objects. One possible research topic is to remove occluded regions in the input images while upsampling or destylizing. In particular, face parsing methods can be firstly utilized to analyze the facial components as well as the occluded regions in the images. Then, we can combine high-level semantic information to inpaint the occluded or missing regions while upsampling LR faces or removing the styles in the portraits. Therefore, we can achieve realistic occlusion-free face images, thus facilitating human observation as well as machine perception.

Bibliography

2017. Archibald prize; art gallery of nsw. https://www.artgallery.nsw.gov.au/prizes/archibald/. (cited on page 217)

Arandjelović, O., 2014. Hallucinating optimal high-dimensional subspaces. *Pattern Recognition*, 47, 8 (2014), 2662–2672. (cited on pages 56 and 90)

Arjovsky, M.; Chintala, S.; and Bottou, L., 2017. Wasserstein generative adversarial networks. In *International Conference on Machine Learning (ICML)*, 214–223. (cited on pages 8, 143, and 190)

Asthana, A.; Marks, T. K.; Jones, M. J.; Tieu, K. H.; and Rohith, M., 2011. Fully automatic pose-invariant face recognition via 3d pose normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 937–944. (cited on pages 6 and 165)

Baker, S. and Kanade, T., 2000. Hallucinating faces. In *Proceedings of 4th IEEE International Conference on Automatic Face and Gesture Recognition, (FG)*, 83–88. (cited on pages 1, 3, 5, 14, 16, 17, 33, 34, 70, 95, 118, 139, 140, and 162)

Baker, S. and Kanade, T., 2002. Limits on super-resolution and how to break them. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 9 (2002), 1167–1183. (cited on pages 1, 5, 14, 17, 33, 56, 57, 70, 72, 90, 95, 121, 140, 162, and 166)

Berthelot, D.; Schumm, T.; and Metz, L., 2017. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, (2017). (cited on pages 8 and 143)

Blanz, V. and Vetter, T., 1999. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 187–194. (cited on pages 6, 162, and 165)

Bousmalis, K.; Silberman, N.; Dohan, D.; Erhan, D.; and Krishnan, D., 2017. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 7. (cited on page 190)

Bruna, J.; Sprechmann, P.; and LeCun, Y., 2016. Super-resolution with deep convolutional sufficient statistics. In *International Conference on Learning Representations (ICLR)*. (cited on pages 4, 17, 30, 33, 58, 73, and 94)

Bulat, A. and Tzimiropoulos, G., 2017a. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *International Conference on Computer Vision (ICCV)*. (cited on pages 90, 118, 120, 123, and 129)

Bulat, A. and Tzimiropoulos, G., 2017b. Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans. *arXiv preprint arXiv:1712.02765*, (2017). (cited on page 6)

Bulat, A. and Tzimiropoulos, G., 2018. Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 109–117. (cited on pages 96, 98, 122, and 142)

Cao, Q.; Lin, L.; Shi, Y.; Liang, X.; and Li, G., 2017. Attention-aware face hallucination via deep reinforcement learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 690–698. (cited on pages 118, 119, and 138)

Chang, F.-J.; Tran, A. T.; Hassner, T.; Masi, I.; Nevatia, R.; and Medioni, G., 2017. Faceposenet: Making a case for landmark-free face alignment. In *Proceeding of the IEEE International Conference on Computer Vision Workshop (ICCVW)*, 1599–1608. (cited on page 7)

Chen, D.; Yuan, L.; Liao, J.; Yu, N.; and Hua, G., 2017. Stylebank: An explicit representation for neural image style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 4. (cited on pages 9, 208, and 226)

Chen, T. Q. and Schmidt, M., 2016. Fast patch-based style transfer of arbitrary style. *arXiv preprint arXiv:1612.04337*, (2016). (cited on pages 9, 188, 191, 205, 208, and 226)

Chen, Y.; Tai, Y.; Liu, X.; Shen, C.; and Yang, J., 2018. Fsrnet: End-to-end learning face super-resolution with facial priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2492–2501. (cited on pages 6, 96, 98, 122, and 142)

Cole, F.; Belanger, D.; Krishnan, D.; Sarna, A.; Mosseri, I.; and Freeman, W. T., 2017. Face synthesis from facial identity features. *arXiv preprint arXiv:1701.04851*, (2017). (cited on pages 7 and 165)

Dabov, K.; Foi, A.; Katkovnik, V.; and Egiazarian, K., 2007. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16, 8 (2007), 2080–2095. (cited on pages xxv, 71, and 80)

Dahl, R.; Norouzi, M.; and Shlens, J., 2017. Pixel recursive super resolution. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 5439–5448. (cited on pages 6, 96, 118, 119, 120, 122, 141, and 167)

Denton, E.; Chintala, S.; Szlam, A.; and Fergus, R., 2015. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 1486–1494. (cited on pages 8, 15, 18, 22, 33, 60, 77, 100, 143, 163, 189, 190, 191, 194, 205, 206, 211, and 225)

Dong, C.; Deng, Y.; Change Loy, C.; and Tang, X., 2015. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 576–584. (cited on page 36)

Dong, C.; Loy, C. C.; and He, K., 2016a. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38, 2 (2016), 295–307. (cited on pages xxi, xxii, xxiii, xxiv, xxv, xxvi, 4, 14, 15, 17, 22, 23, 24, 25, 26, 27, 30, 33, 36, 39, 42, 43, 44, 45, 46, 48, 49, 58, 60, 61, 63, 64, 65, 67, 73, 80, 81, 85, 86, 92, and 94)

Dong, C.; Loy, C. C.; and Tang, X., 2016b. Accelerating the super-resolution convolutional neural network. In *European Conference on Computer Vision (ECCV)*, 391–407. (cited on pages 4, 30, and 33)

Dovgard, R. and Basri, R., 2004. Statistical symmetric shape from shading for 3d structure recovery of faces. In *European Conference on Computer Vision (ECCV)*, 99–113. (cited on pages 6 and 165)

Dumoulin, V.; Shlens, J.; and Kudlur, M., 2016. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, (2016). (cited on pages 9, 188, 191, 205, 208, and 226)

Fasel, B. and Luettin, J., 2003. Automatic facial expression analysis: a survey. *Pattern Recognition*, 36, 1 (2003), 259–275. (cited on pages 118 and 138)

Fischer, P.; Dosovitskiy, A.; Ilg, E.; Häusser, P.; Hazirbaş, C.; Golkov, V.; van der Smagt, P.; Cremers, D.; and Brox, T., 2015. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2758–2766. (cited on pages 31 and 59)

Freedman, G. and Fattal, R., 2010. Image and video upscaling from local self-examples. *ACM Transactions on Graphics*, 28, 3 (2010), 1–10. (cited on pages 4, 17, 33, and 94)

Freeman, W. T.; Jones, T. R.; and Pasztor, E. C., 2002. Example-based super-resolution. *IEEE Computer Graphics and Applications*, 22, 2 (2002), 56–65. (cited on pages 3, 4, 16, 17, 32, 33, and 94)

Gatys, L.; Ecker, A. S.; and Bethge, M., 2015. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 262–270. (cited on page 226)

Gatys, L. A.; Bethge, M.; Hertzmann, A.; and Shechtman, E., 2016a. Preserving color in neural artistic style transfer. *arXiv preprint arXiv:1606.05897*, (2016). (cited on pages 9, 191, and 207)

Gatys, L. A.; Ecker, A. S.; and Bethge, M., 2016b. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2414–2423. (cited on pages xxxii, xxxiii, 8, 9, 188, 189, 190, 196, 197, 198, 200, 207, 209, 213, 215, 216, 224, and 226)

Gatys, L. A.; Ecker, A. S.; Bethge, M.; Hertzmann, A.; and Shechtman, E., 2017. Controlling perceptual factors in neural style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3730–3738. (cited on pages 9, 191, 204, 205, 207, and 226)

Glasner, D.; Bagon, S.; and Irani, M., 2009. Super-Resolution from a Single Image. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 349–356. (cited on pages 4, 16, 17, 32, 33, and 94)

Gonzalez, R. and Wintz, P., 1977. Digital image processing (second edition). (1977), 187–191. (cited on page 37)

Goodfellow, I.; Pouget-Abadie, J.; and Mirza, M., 2014. Generative Adversarial Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2672—-2680. (cited on pages 3, 6, 7, 15, 18, 22, 33, 48, 60, 77, 94, 97, 100, 106, 122, 140, 143, 154, 163, 167, 174, 176, 189, 190, 194, 195, 205, 206, 211, and 224)

Gu, S.; Zuo, W.; Xie, Q.; Meng, D.; Feng, X.; and Zhang, L., 2015. Convolutional sparse coding for image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1823–1831. (cited on pages 17, 37, and 94)

Gupta, A.; Johnson, J.; Alahi, A.; and Fei-Fei, L., 2017. Characterizing and improving stability in neural style transfer. *arXiv preprint arXiv:1705.02092*, (2017). (cited on pages 10 and 208)

Hassner, T., 2013. Viewing real-world faces in 3d. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 3607–3614. (cited on pages 6, 162, and 165)

Hassner, T.; Harel, S.; Paz, E.; and Enbar, R., 2015. Effective face frontalization in unconstrained images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4295–4304. (cited on pages xxx, xxxi, 6, 162, 163, 165, 169, 174, 175, 176, 177, 178, 179, 180, 181, and 183)

Hennings-Yeomans, P. H.; Baker, S.; and Kumar, B. V. K. V., 2008. Simultaneous super-resolution and feature extraction for recognition of low-resolution faces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–8. (cited on pages 56, 70, 90, and 162)

Hertzmann, A.; Jacobs, C. E.; Oliver, N.; Curless, B.; and Salesin, D. H., 2001. Image analogies. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 327–340. (cited on page 191)

Hinton, G., 2012. Neural Networks for Machine Learning Lecture 6a: Overview of mini-batch gradient descent Reminder: The error surface for a linear neuron. (2012). (cited on pages 20, 37, 62, 78, 103, 128, 147, 172, 194, and 211)

Hong Chang; Dit-Yan Yeung; and Yimin Xiong, 2004. Super-resolution through neighbor embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 275–282. (cited on pages 4, 16, 17, 32, 33, and 94)

Huang, G. B.; Ramesh, M.; Berg, T.; and Learned-Miller, E., 2007. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07–49, University of Massachusetts, Amherst. (cited on pages 2, 14, 31, 91, 95, 119, 129, 164, 174, and 195)

Huang, H.; He, R.; Sun, Z.; and Tan, T., 2017a. Wavelet-srnet: A wavelet-based cnn for multi-scale face super resolution. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1689–1697. (cited on pages 92, 96, 141, and 167)

Huang, J.-B.; Singh, A.; and Ahuja, N., 2015. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5197–5206. (cited on pages 4, 16, 17, 32, 33, and 94)

Huang, R.; Zhang, S.; Li, T.; and He, R., 2017b. Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis. *arXiv preprint arXiv:1704.04086*, (2017). (cited on pages 7, 165, and 206)

Huang, X. and Belongie, S. J., 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1510–1519. (cited on pages 9, 188, 205, and 208)

Huang, X.; Li, Y.; Poursaeed, O.; Hopcroft, J. E.; and Belongie, S. J., 2017c. Stacked generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 3. (cited on page 190)

Ioffe, S. and Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 448–456. (cited on pages 36, 174, and 190)

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A., 2016. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, (2016). (cited on pages xxxii, xxxiii, xxxiv, xxxv, 8, 188, 190, 191, 192, 197, 198, 200, 205, 206, 213, 214, 215, 216, 222, 223, 225, 228, 233, 235, 236, 237, 238, and 239)

Jaderberg, M.; Simonyan, K.; Zisserman, A.; et al., 2015. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2017–2025. (cited on pages xxv, xxx, 35, 57, 59, 66, 71, 75, 92, 96, 99, 112, 119, 121, 122, 142, 144, 163, 167, 168, 205, 209, and 228)

Jayasumana, S.; Hartley, R.; Salzmann, M.; Li, H.; and Harandi, M., 2013. Kernel methods on the riemannian manifold of symmetric positive definite matrices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 73–80. (cited on pages 224 and 232)

Jia, K. and Gong, S., 2008. Generalized face super-resolution. *IEEE Transactions on Image Processing*, 17, 6 (2008), 873–886. (cited on pages 1, 5, 14, 17, 30, and 33)

Jin, Y. and Bouganis, C.-S., 2015. Robust multi-image based blind face hallucination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5252–5260. (cited on pages xxiii, xxiv, 5, 34, 42, 43, 44, 45, 48, 49, 50, and 121)

Jing, Y.; Yang, Y.; Feng, Z.; Ye, J.; and Song, M., 2017. Neural style transfer: A review. *arXiv preprint arXiv:1705.04058*, (2017). (cited on page 191)

Johnson, J.; Alahi, A.; and Fei-Fei, L., 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 694–711. (cited on pages xxxii, xxxiii, xxxiv, xxxv, 4, 9, 33, 93, 94, 101, 144, 164, 170, 171, 188, 189, 191, 195, 196, 197, 198, 200, 204, 205, 207, 209, 212, 213, 214, 215, 216, 226, 233, 235, 237, 238, and 239)

Karacan, L.; Akata, Z.; Erdem, A.; and Erdem, E., 2016. Learning to generate images of outdoor scenes from attributes and semantic layouts. *arXiv preprint arXiv:1612.00215*, (2016). (cited on pages 8, 191, and 206)

Kim, J.; Kwon Lee, J.; and Mu Lee, K., 2016a. Accurate image super-resolution using very deep convolutional networks. (2016), 1646–1654. (cited on pages xxiii, xxvi, xxvii, xxviii, xxix, xxxi, 4, 14, 17, 30, 33, 41, 42, 43, 44, 45, 46, 48, 58, 73, 91, 92, 94, 100, 105, 106, 107, 108, 109, 114, 129, 130, 131, 134, 135, 139, 149, 150, 151, 152, 154, 176, 177, 178, 179, and 180)

Kim, J.; Kwon Lee, J.; and Mu Lee, K., 2016b. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1637–1645. (cited on pages xxiii, 4, 30, 33, 42, 43, 44, 45, 46, 48, 94, 100, and 105)

Kim, T.; Cha, M.; Kim, H.; Lee, J.; and Kim, J., 2017. Learning to discover cross-domain relations with generative adversarial networks. *arXiv preprint arXiv:1703.05192*, (2017). (cited on page 190)

Kingma, D. P. and Welling, M., 2013. Auto-Encoding Variational Bayes. *arXiv:1312.6114*, , Ml (2013), 1–14. (cited on pages 8, 18, 33, 190, 206, and 225)

Kolouri, S. and Rohde, G. K., 2015. Transport-based single frame super resolution of very low resolution face images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4876–4884. (cited on pages 5, 17, 30, 33, 34, 56, 58, 70, 72, 90, 95, 121, 141, 162, and 166)

Koniusz, P. and Cherian, A., 2016. Sparse coding for third-order super-symmetric tensor descriptors with application to texture recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5395–5403. (cited on pages 191 and 205)

Koniusz, P.; Tas, Y.; and Porikli, F., 2017a. Domain adaptation by mixture of alignments of second-or higher-order scatter tensors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7139–7148. (cited on pages 191 and 205)

Koniusz, P.; Yan, F.; Gosselin, P.-H.; and Mikolajczyk, K., 2017b. Higher-order occurrence pooling for bags-of-words: Visual concept detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 2 (2017), 313–326. (cited on pages 191 and 205)

Lai, W.-S.; Huang, J.-B.; Ahuja, N.; and Yang, M.-H., 2017. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 624–632. (cited on page 94)

Larsen, A. B. L.; Sønderby, S. K.; Larochelle, H.; and Winther, O., 2016. Autoencoding beyond pixels using a learned similarity metric. In *International Conference on Machine Learning (ICML)*, 1558–1566. (cited on page 8)

Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A. P.; Tejani, A.; Totz, J.; Wang, Z.; et al., 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 4. (cited on pages xxvi, xxvii, xxviii, xxix, xxxi, 4, 33, 91, 92, 94, 100, 105, 106, 107, 108, 109, 114, 125, 126, 129, 130, 131, 134, 135, 149, 150, 151, 152, 154, 176, 177, 178, 179, 180, 206, and 225)

Lee, C.-H.; Zhang, K.; Lee, H.-C.; Cheng, C.-W.; and Hsu, W., 2018. Attribute augmented convolutional neural network for face hallucination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 721–729. (cited on page 142)

Li, C. and Wand, M., 2016a. Combining markov random fields and convolutional neural networks for image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2479–2486. (cited on pages 9, 191, 207, and 226)

Li, C. and Wand, M., 2016b. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*, 702–716. (cited on pages xxxii, xxxiii, 9, 188, 190, 191, 197, 198, 200, 205, 206, 207, 213, 214, 215, 216, 225, and 226)

Li, Y.; Cai, C.; Qiu, G.; and Lam, K. M., 2014. Face hallucination based on sparse local-pixel structure. *Pattern Recognition*, 47, 3 (2014), 1261–1270. (cited on pages 5, 17, 34, 56, 58, 70, 73, 90, 95, 121, 141, 162, and 166)

Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; and Yang, M.-H., 2017a. Diversified texture synthesis with feed-forward networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 266–274. (cited on pages 9, 188, 205, 208, and 226)

Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; and Yang, M.-H., 2017b. Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems (NIPS)*, 386–396. (cited on page 226)

Lin, Z.; He, J.; Tang, X.; and Tang, C.-K., 2008. Limits of learning-based superresolution algorithms. *International journal of computer vision*, 80, 3 (2008), 406–420. (cited on page 94)

Lin, Z. and Shum, H. Y., 2006. Response to the comments on "Fundamental limits of reconstruction-based superresolution algorithms under local translation". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28, 5 (2006), 83–97. (cited on pages 4, 17, 32, and 94)

Liu, C.; Shum, H.; and Zhang, C., 2001. A two-step approach to hallucinating faces: global parametric model and local nonparametric model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 192–198. (cited on pages 1, 5, 14, 17, 33, 56, 70, 90, 95, and 162)

Liu, C.; Shum, H. Y.; and Freeman, W. T., 2007. Face hallucination: Theory and practice. *International Journal of Computer Vision*, 75, 1 (2007), 115–134. (cited on pages xxi, xxii, xxiii, xxiv, 1, 5, 14, 17, 22, 24, 25, 26, 27, 30, 33, 34, 42, 43, 44, 45, 48, 56, 57, 63, 64, 65, 67, 70, 72, 90, 95, 121, 141, 162, and 166)

Liu, C.; Yuen, J.; and Torralba, A., 2011. Sift flow: Dense correspondence across scenes and its applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33, 5 (2011), 978–994. (cited on pages 58, 73, 95, 121, 141, and 166)

Liu, M.-Y.; Breuel, T.; and Kautz, J., 2017. Unsupervised image-to-image translation networks. *arXiv preprint arXiv:1703.00848*, (2017). (cited on page 190)

Liu, M.-Y. and Tuzel, O., 2016. Coupled generative adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, 469–477. (cited on page 190)

Liu, Z.; Luo, P.; Wang, X.; and Tang, X., 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 3730–3738. (cited on pages 2, 14, 23, 31, 42, 63, 80, 91, 95, 105, 119, 129, 150, 164, 174, 175, 195, 212, and 231)

Long, J.; Shelhamer, E.; and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3431–3440. (cited on pages 31, 59, 144, and 228)

Ma, X.; Zhang, J.; and Qi, C., 2010. Hallucinating face by position-patch. *Pattern Recognition*, 43, 6 (2010), 2224âĂŞ–2236. (cited on pages xxi, xxii, xxiii, xxiv, xxv, xxvi, xxvii, xxviii, xxix, xxxi, 2, 5, 14, 17, 22, 23, 24, 25, 26, 27, 33, 34, 42, 43, 44, 45, 48, 49, 50, 56, 58, 63, 65, 67, 70, 73, 79, 80, 81, 85, 86, 90, 95, 105, 106, 107, 108, 109, 120, 121, 128, 129, 130, 131, 134, 135, 139, 141, 149, 150, 151, 152, 154, 162, 166, 174, 175, 176, 177, 178, 179, and 180)

Maas, A. L.; Hannun, A. Y.; and Ng, A. Y. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, vol. 30, 3. (cited on page 211)

Mao, X.; Shen, C.; and Yang, Y.-B., 2016. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in Neural Information Processing Systems (NIPS)*, 2802–2810. (cited on pages xxii, xxiii, xxiv, 4, 30, 31, 33, 39, 42, 43, 44, 45, 46, 48, 49, and 50)

Masi, I.; Tran, A. T.; Hassner, T.; Leksut, J. T.; and Medioni, G., 2016. Do we really need to collect millions of faces for effective face recognition? In *European Conference on Computer Vision (ECCV)*, 579–596. (cited on pages 6, 164, 165, 170, and 174)

Nejati, H. and Sim, T., 2011. A study on recognizing non-artistic face sketches. In *IEEE Winter Conference Applications of Computer Vision Workshop (WACVW)*, 240–247. (cited on pages 8, 207, and 225)

Newell, A.; Yang, K.; and Deng, J., 2016. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision (ECCV)*, 483–499. (cited on pages 124 and 125)

Odena, A.; Dumoulin, V.; and Olah, C., 2016. Deconvolution and checkerboard artifacts. *Distill*, (2016). (cited on pages 35 and 36)

Parkhi, O. M.; Vedaldi, A.; and Zisserman, A., 2015. Deep face recognition. In *British Machine Vision Conference (BMVC)*, vol. 1, 6–17. (cited on pages 181, 200, 215, and 237)

Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; and Efros, A. A., 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2536–2544. (cited on pages 206 and 225)

Peleg, T. and Elad, M., 2014. A statistical prediction model based on sparse representations for single image super-resolution. *IEEE Transactions on Image Processing*, 23, 6 (2014), 2569–2582. (cited on pages 3, 16, and 32)

Perarnau, G.; van de Weijer, J.; Raducanu, B.; and Álvarez, J. M., 2016. Invertible Conditional GANs for image editing. In *NIPS Workshop on Adversarial Training*. (cited on pages 8, 143, 150, and 225)

Phillips, P. J.; Wechsler, H.; Huang, J.; and Rauss, P. J., 1998. The feret database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing*, 16, 5 (1998), 295–306. (cited on page 217)

Radford, A.; Metz, L.; and Chintala, S., 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434*, (2015), 1–15. (cited on pages 6, 8, 9, 18, 33, 60, 77, 100, 106, 122, 125, 143, 145, 154, 174, 176, 191, 194, 195, 207, 211, and 226)

Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; and Lee, H., 2016. Generative adversarial text to image synthesis. *arXiv preprint arXiv:1605.05396*, (2016). (cited on pages 8, 143, and 190)

Ren, M.; Liao, R.; Urtasun, R.; Sinz, F. H.; and Zemel, R. S., 2017. Normalizing the normalizers: Comparing and extending network normalization schemes. In *International Conference on Learning Representations (ICLR)*. (cited on page 41)

Risser, E.; Wilmot, P.; and Barnes, C., 2017. Stable and controllable neural texture synthesis and style transfer using histogram losses. *arXiv preprint arXiv:1701.08893*, (2017). (cited on pages 9, 191, 207, and 226)

Sagonas, C.; Panagakis, Y.; Zafeiriou, S.; and Pantic, M., 2015. Robust statistical face frontalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 3871–3879. (cited on pages 7, 162, and 165)

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; and Chen, X., 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems (NIPS)*, 2234–2242. (cited on pages 190 and 191)

Sangkloy, P.; Lu, J.; Fang, C.; Yu, F.; and Hays, J., 2017. Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6836–6845. (cited on pages 8, 191, 206, and 225)

Schroff, F.; Kalenichenko, D.; and Philbin, J., 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 815–823. (cited on pages 172 and 228)

Schulter, S. and Leistner, C., 2015. Fast and Accurate Image Upscaling with Super-Resolution Forests. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3791–3799. (cited on pages 4, 16, and 32)

Selim, A.; Elgharib, M.; and Doyle, L., 2016. Painting style transfer for head portraits using convolutional neural networks. *ACM Transactions on Graphics (ToG)*, 35, 4 (2016), 129. (cited on pages 9 and 207)

Sharma, A. and Jacobs, D. W., 2011. Bypassing synthesis: Pls for face recognition with pose, low-resolution and sketch. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 593–600. (cited on pages 8, 207, and 225)

Shen, W. and Liu, R., 2016. Learning residual images for face attribute manipulation. *arXiv preprint arXiv:1612.05363*, (2016). (cited on pages 8, 143, 150, and 225)

Shi, W.; Caballero, J.; Huszár, F.; Totz, J.; Aitken, A. P.; Bishop, R.; Rueckert, D.; and Wang, Z., 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1874–1883. (cited on pages 4, 31, 33, and 94)

Shiri, F.; Porikli, F.; Hartley, R.; and Koniusz, P., 2018. Identity-preserving face recovery from portraits. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 102–111. (cited on pages xxxiv, xxxv, 222, 223, 225, 226, 228, 229, 231, 233, 235, 236, 237, 238, and 239)

Shiri, F.; Yu, X.; Koniusz, P.; and Porikli, F., 2017. Face destylization. In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 1–8. (cited on pages xxxiv, xxxv, 206, 222, 223, 226, 228, 233, 235, 237, 238, and 239)

Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, (2014). (cited on pages 102, 127, 144, 147, 171, 209, and 232)

Singh, A.; Porikli, F.; and Ahuja, N., 2014. Super-resolving noisy images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2846–2853. (cited on pages 4, 17, 33, and 94)

Tai, Y.; Yang, J.; and Liu, X., 2017. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1. (cited on page 94)

Taigman, Y.; Yang, M.; Ranzato, M.; and Wolf, L., 2014. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1701–1708. (cited on pages 6, 118, 162, and 165)

Tang, X. and Wang, X., 2003. Face sketch synthesis and recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 687–694. (cited on pages 8, 207, and 225)

Tappen, M. F. and Liu, C., 2012. A Bayesian Approach to Alignment-Based Image Hallucination. In *European Conference on Computer Vision (ECCV)*, vol. 7578, 236–249. (cited on pages 1, 5, 14, 17, 18, 30, 33, 34, 56, 58, 70, 72, 73, 91, 95, 121, 141, and 166)

Tappen, M. F.; Russell, B. C.; and Freeman, W. T., 2003. Exploiting the sparse derivative prior for super-resolution and image demosaicing. In *In IEEE Workshop on Statistical and Computational Theories of Vision*. (cited on page 94)

Thies, J.; Zollhöfer, M.; Stamminger, M.; Theobalt, C.; and Niessner, M., 2016. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (cited on page 162)

Tipping, M. E. and Bishop, C. M., 1999. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11, 2 (1999), 443–482. (cited on page 47)

Tran, A. T.; Hassner, T.; Masi, I.; and Medioni, G., 2017a. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1493–1502. (cited on pages 7 and 166)

Tran, L.; Yin, X.; and Liu, X., 2017b. Disentangled representation learning gan for pose-invariant face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 4, 7. (cited on pages 7 and 165)

Ulyanov, D.; Lebedev, V.; Vedaldi, A.; and Lempitsky, V. S., 2016a. Texture networks: Feed-forward synthesis of textures and stylized images. In *International Conference on Machine Learning (ICML)*, 1349–1357. (cited on pages 9, 188, 191, 195, 205, 207, 225, and 226)

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V., 2016b. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, (2016). (cited on pages 9, 10, 188, 191, 195, 205, 207, and 208)

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V., 2017. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 3–11. (cited on pages 9, 10, 191, 207, 208, and 226)

Van Den Oord, A.; Kalchbrenner, N.; and Kavukcuoglu, K., 2016. Pixel recurrent neural networks. In *International Conference on Machine Learning (ICML)*, 1747–1756. (cited on pages 6, 96, 122, 141, 167, 190, 206, and 225)

Wang, N.; Tao, D.; Gao, X.; Li, X.; and Li, J., 2014. A comprehensive survey to face hallucination. *International Journal of Computer Vision*, 106, 1 (2014), 9–30. (cited on pages 5, 14, 17, 33, 70, 72, 95, 141, 162, and 166)

Wang, X.; Oxholm, G.; Zhang, D.; and Wang, Y.-F., 2017. Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 7. (cited on pages 9, 207, and 226)

Wang, X. and Tang, X., 2005. Hallucinating face by eigen transformation. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 35, 3 (2005), 425–434. (cited on pages 1, 5, 14, 17, 30, 33, 34, 56, 57, 70, 72, 90, 95, 121, 141, 162, and 166)

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13, 4 (2004), 600–612. (cited on pages 199, 214, and 236)

Wang, Z.; Yang, Y.; Wang, Z.; Chang, S.; Han, W.; Yang, J.; and Huang, T., 2015. Self-tuned deep super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–8. (cited on pages 58 and 73)

Xiong, X. and De la Torre, F., 2013. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 532–539. (cited on pages 118 and 129)

Xu, X.; Sun, D.; Pan, J.; Zhang, Y.; Pfister, H.; and Yang, M.-H., 2017. Learning to super-resolve blurry face and text images. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 251–260. (cited on pages 6, 96, 121, 125, 126, 127, 141, 163, and 167)

Yan, X.; Yang, J.; Sohn, K.; and Lee, H., 2016. Attribute2image: Conditional image generation from visual attributes. (2016), 776–791. (cited on pages 8, 140, 143, 146, 150, 225, and 230)

Yang, C. Y.; Liu, S.; and Yang, M. H., 2013. Structured face hallucination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1099–1106. (cited on pages xxi, xxii, xxiii, xxiv, 1, 5, 14, 17, 18, 22, 23, 24, 25, 26, 27, 30, 33, 34, 42, 43, 44, 45, 48, 49, 50, 56, 58, 63, 65, 67, 70, 72, 73, 91, 95, 119, 121, 141, and 166)

Yang, C.-Y.; Liu, S.; and Yang, M.-H., 2017a. Hallucinating compressed face images. *International Journal of Computer Vision*, (2017), 1–18. (cited on pages 91, 95, and 166)

Yang, C.-Y.; Ma, C.; and Yang, M.-H., 2014. Single-image super-resolution: A benchmark. In *European Conference on Computer Vision (ECCV)*, 372–386. (cited on pages 1, 14, and 30)

Yang, C.-Y. and Yang, M.-H., 2013. Fast Direct Super-Resolution by Simple Functions. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 561–568. (cited on pages 3, 16, and 32)

Yang, F.; Wang, J.; Shechtman, E.; Bourdev, L.; and Metaxas, D., 2011. Expression flow for 3d-aware face component transfer. In *ACM Transactions on Graphics*, vol. 30, 60. (cited on pages 6, 162, and 165)

Yang, J.; Wright, J.; Huang, T. S.; and Ma, Y., 2010. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19, 11 (2010), 2861–73. (cited on pages xxi, xxii, xxiii, xxiv, 1, 2, 4, 5, 14, 16, 17, 22, 23, 24, 25, 26, 27, 30, 32, 33, 42, 43, 44, 45, 48, 56, 63, 64, 65, 67, 70, 73, 90, 94, 121, 141, 162, and 166)

Yang, S.; Luo, P.; Loy, C.-C.; and Tang, X., 2016. Wider face: A face detection benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5525–5533. (cited on pages 95, 114, and 185)

Yang, Z.; Zhang, K.; Liang, Y.; and Wang, J., 2017b. Single image super-resolution with a parameter economic residual-like convolutional neural network. In *International Conference on Multimedia Modeling (MMM)*, 353–364. (cited on page 41)

Yeh, R.; Chen, C.; Lim, T. Y.; Hasegawa-Johnson, M.; and Do, M. N., 2016. Semantic image inpainting with perceptual and contextual losses. *arXiv preprint arXiv:1607.07539*, (2016). (cited on page 225)

Yim, J.; Jung, H.; Yoo, B.; Choi, C.; Park, D.; and Kim, J., 2015. Rotating your face using multi-task deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 787–796. (cited on pages 7 and 165)

Yin, R., 2016. Content aware neural style transfer. *arXiv preprint arXiv:1601.04568*, (2016). (cited on pages 9, 191, and 207)

Yin, X.; Yu, X.; Sohn, K.; Liu, X.; and Chandraker, M., 2017. Towards large-pose face frontalization in the wild. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1–10. (cited on pages 7, 165, and 166)

Yu, X.; Fernando, B.; Hartley, R.; and Porikli, F., 2018. Super-resolving very low-resolution face images with supplementary attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 908–917. (cited on page 96)

Yu, X. and Porikli, F., 2016. Ultra-resolving face images by discriminative generative networks. In *European Conference on Computer Vision (ECCV)*, 318–333. (cited on pages xxiii, xxv, xxvi, 34, 35, 37, 42, 43, 44, 45, 48, 70, 71, 72, 73, 74, 75, 76, 77, 79, 80, 81, 84, 85, 86, 92, 93, 95, 98, 100, 101, 102, 111, 113, 118, 121, 122, 125, 126, 127, 138, 139, 141, 144, 145, 156, 157, 160, 162, 163, 166, 167, 170, 171, 177, 184, 189, 191, 193, 205, and 206)

Yu, X. and Porikli, F., 2017a. Face hallucination with tiny unaligned images by transformative discriminative neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*, vol. 2, 3. (cited on pages 92, 93, 96, 98, 101, 102, 112, 113, 121, 139, 141, 144, 151, 157, 159, 163, 167, 170, 171, 205, 206, and 225)

Yu, X. and Porikli, F., 2017b. Hallucinating very low-resolution unaligned and noisy face images by transformative discriminative autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3760–3768. (cited on pages xxvi, xxvii, xxviii, xxix, xxx, xxxi, 35, 91, 92, 93, 96, 98, 100, 102, 105, 106, 107, 108, 109, 110, 111, 112, 118, 119, 120, 121, 122, 125, 127, 129, 130, 131, 134, 135, 138, 139, 142, 143, 144, 149, 150, 151, 152, 155, 156, 157, 163, 167, 169, 170, 171, 176, 177, 178, 179, 180, 184, 206, and 225)

Yu, X. and Porikli, F., 2018. Imagining the unimaginable faces by deconvolutional networks. *IEEE Transactions on Image Processing*, (02 2018), 1–1. (cited on pages 92, 95, 99, 113, 139, 141, 144, 162, and 166)

Yu, X.; Xu, F.; Zhang, S.; and Zhang, L., 2014. Efficient patch-wise non-uniform deblurring for a single image. *IEEE Transactions on Multimedia*, 16, 6 (2014), 1510–1524. (cited on pages 37 and 95)

Yuen, P. C. and Man, C., 2007. Human face image searching system using sketches. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37, 4 (2007), 493–504. (cited on pages 8, 207, and 225)

Zafeiriou, S.; Trigeorgis, G.; Chrysos, G.; Deng, J.; and Shen, J., 2017. The menpo facial landmark localisation challenge: A step towards the solution. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition workshops (CVPRW)*, 2116–2125. (cited on pages 119 and 129)

Zeiler, M. D. and Fergus, R., 2014. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, 818–833. (cited on pages 31, 74, 99, 145, 158, and 229)

Zeiler, M. D.; Krishnan, D.; Taylor, G. W.; and Fergus, R., 2010. Deconvolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2528–2535. (cited on pages 31, 74, and 99)

Zeiler, M. D.; Taylor, G. W.; and Fergus, R., 2011. Adaptive deconvolutional networks for mid and high level feature learning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2018–2025. (cited on pages 19 and 20)

Zhang, H. and Dana, K., 2017. Multi-style generative network for real-time transfer. *arXiv preprint arXiv:1703.06953*, (2017). (cited on pages 9, 188, 205, and 208)

Zhang, H.; Sindagi, V.; and Patel, V. M., 2017a. Image de-raining using a conditional generative adversarial network. *arXiv preprint arXiv:1701.05957*, (2017). (cited on pages 206 and 225)

Zhang, H.; Xu, T.; and Li, H., 2017b. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 5908–5916. (cited on pages 8, 140, 143, 146, and 230)

Zhang, W.; Wang, X.; and Tang, X., 2011. Coupled information-theoretic encoding for face photo-sketch recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 513–520. (cited on page 216)

Zhang, Z.; Luo, P.; Loy, C. C.; and Tang, X., 2014. Facial landmark detection by deep multi-task learning. In *European Conference on Computer Vision (ECCV)*, 94–108. (cited on pages xxxiii, xxxiv, 204, and 222)

Zhao, J.; Mathieu, M.; and LeCun, Y., 2016. Energy-based generative adversarial network. *arXiv preprint arXiv:1609.03126*, (2016). (cited on pages 8 and 143)

Zhao, W.; Chellappa, R.; Phillips, P. J.; and Rosenfeld, A., 2003. Face recognition: A literature survey. *ACM Computing Surveys (CSUR)*, 35, 4 (2003), 399–458. (cited on pages 118 and 138)

Zhou, E. and Fan, H., 2015. Learning Face Hallucination in the Wild. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 3871–3877. (cited on pages 5, 6, 14, 17, 18, 33, 35, 56, 58, 70, 72, 73, 95, 118, 121, 141, and 166)

Zhu, J.-Y.; Krähenbühl, P.; Shechtman, E.; and Efros, A. A., 2016a. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision (ECCV)*, 597–613. (cited on page 225)

Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2242–2251. (cited on pages xxxiii, xxxiv, xxxv, 8, 213, 214, 215, 216, 223, 228, 233, 235, 236, 237, 238, and 239)

Zhu, S.; Liu, S.; Loy, C. C.; and Tang, X., 2016b. Deep cascaded bi-network for face hallucination. In *European Conference on Computer Vision (ECCV)*, 614–630. (cited on pages xxiii, xxiv, xxv, xxvi, xxvii, xxviii, xxix, xxx, xxxi, 5, 6, 30, 34, 42, 43, 44, 45, 48, 49, 50, 70, 72, 73, 76, 81, 85, 86, 91, 92, 96, 98, 105, 106, 107, 108, 109, 118, 119, 122, 129, 130, 131, 134, 135, 138, 139, 142, 149, 150, 151, 152, 154, 156, 167, 169, 176, 177, 178, 179, and 180)

Zhu, X.; Lei, Z.; Yan, J.; Yi, D.; and Li, S. Z., 2015. High-fidelity pose and expression normalization for face recognition in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 787–796. (cited on pages 6, 7, and 165)

Zhu, X. and Ramanan, D., 2012. Face detection, pose estimation, and landmark localization in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2879–2886. (cited on pages xxx, 90, and 163)

Zhu, Z.; Luo, P.; Wang, X.; and Tang, X., 2014. Recover canonical-view faces in the wild with deep neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2014). (cited on pages 7 and 165)