

# **Inferring Human Pose and Motion from Images**

**Yifan Lu**

Research School of Engineering  
The College of Computer Sciences and Engineering  
The Australian National University

August 2011



---

# Acknowledgements

---

I am very grateful to be a member of the computer vision group at Australian National University. I would like to thank my supervisors Dr. Lei Wang, Professor Richard Hartley and Dr. Hongdong Li for invaluable direction and assistance in the development of this work. Thanks especially to Leslay Goldberg, Elspeth Davies and Marie Katselas for helping with conference travel and other issues in PhD life. Thank you also to Professor Henry Gardener for the encouragement and help to international students. Thank you to fellow students, Novi Quadrianto, Yuhang Zhang, Peter Carr, Tamir Yedidya, Luping Zhou and Cong Phuoc Huynh for their accompany. Last but not least, thanks to my parents for their continuous support.



---

# Abstract

---

As optical gesture recognition technology advances, touchless human computer interfaces of the future will soon become a reality. One particular technology, markerless motion capture, has gained a large amount of attention, with widespread application in diverse disciplines, including medical science, sports analysis, advanced user interfaces, and virtual arts. However, the complexity of human anatomy makes markerless motion capture a non-trivial problem: I) parameterised pose configuration exhibits high dimensionality, and II) there is considerable ambiguity in surjective inverse mapping from observation to pose configuration spaces with a limited number of camera views. These factors together lead to multimodality in high dimensional space, making markerless motion capture an ill-posed problem. This study challenges these difficulties by introducing a new framework. It begins with automatically modelling specific subject template models and calibrating posture at the initial stage. Subsequent tracking is accomplished by embedding naturally-inspired global optimisation into the sequential Bayesian filtering framework. Tracking is enhanced by several robust evaluation improvements. Sparsity of images is managed by compressive evaluation, further accelerating computational efficiency in high dimensional space.



---

# Contents

---

<b>Acknowledgements</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Problem Formalisation . . . . .	2
1.2 Challenges . . . . .	3
1.2.1 Ambiguity and Multimodality . . . . .	3
1.2.2 High Dimensionality . . . . .	5
1.2.3 Subject Specific Modelling . . . . .	6
1.2.4 Computational Performance . . . . .	6
1.3 Thesis Outline . . . . .	6
<b>2 Literature review</b>	<b>9</b>
2.1 Global Optimisation and Template based Generative Approach . . . . .	9
2.1.1 Global Optimisation . . . . .	10
2.1.1.1 Annealed Particle Filter . . . . .	10
2.1.1.2 Covariance Scaled Sampling . . . . .	11
2.1.1.3 Interacting Simulated Annealing and Hybrid Approach . . . . .	12
2.1.2 Template Model Generation . . . . .	15
2.1.2.1 Articulated ICP and Subject Specific Model Generation . . . . .	15
2.1.2.2 Performance Capture . . . . .	17
2.1.3 Object Localisation and Pose Estimation using a Graphic Model . . . . .	21
2.2 Learning Based Approach . . . . .	22
2.2.1 Tracking by Regression . . . . .	22
2.2.2 Gaussian Process Dynamical Model . . . . .	24
2.2.3 Dimensionality Reduction and Manifold Learning on Visual Tracking . . . . .	26
2.3 Graph Based Image Segmentation and Tracking . . . . .	29
2.3.1 Simultaneous Segmentation and Pose Estimation . . . . .	31
2.3.2 Transductive Image Segmentation . . . . .	32
2.3.3 Laplacian Matrix and Tracking . . . . .	36
<b>3 Architecture Overview and Sequential Tracking Pipeline</b>	<b>39</b>
3.1 Architecture of Human Motion Capture . . . . .	39
3.2 Sequential Bayesian Filtering Framework . . . . .	43
3.3 Particle Filter on Visual Tracking . . . . .	46

---

3.3.1	Recursive Bayesian Filtering . . . . .	46
<b>4</b>	<b>Subject Specific Body Shape Modelling and Automatic Initialisation</b>	<b>51</b>
4.1	Related Works . . . . .	52
4.2	Generic Human Body Skeleton . . . . .	55
4.3	Human body shape . . . . .	58
4.3.1	Needle based Body Shape Parameterisation . . . . .	59
4.3.1.1	Contour from Needle Projection . . . . .	66
4.3.2	Data-Driven Body Shape Parameterisation . . . . .	69
4.3.2.1	Dynamic Bone Length and Collision Bounding Box Adjustment . . . . .	71
4.4	Automatic Initialisation . . . . .	74
4.4.1	Using Needle based Body Parameterisation . . . . .	75
4.4.2	Using Data-Driven Shape Parameterisation . . . . .	77
<b>5</b>	<b>Nature Inspired Global Optimisation</b>	<b>81</b>
5.1	Simulated Annealing . . . . .	82
5.1.1	Simulated Annealing Particle Filter . . . . .	85
5.2	Particle Swarm Optimisation . . . . .	88
5.2.1	Algorithm Description . . . . .	91
5.3	Covariance Matrix Adaptation Evolution Strategy . . . . .	92
5.3.1	Evolution Strategy . . . . .	93
5.3.2	Covariance Matrix Adaptation . . . . .	95
5.3.2.1	Selection and Recombination . . . . .	95
5.3.2.2	Adapting the Covariance Matrix . . . . .	96
5.3.2.3	Step-Size Control . . . . .	102
5.4	Covariance Matrix Adaptation Annealing . . . . .	104
5.4.1	Problems in Dynamic Settings . . . . .	104
5.4.2	Covariance Matrix Adaptation Annealing Algorithm . . . . .	105
5.4.2.1	Perturbation Matrix and Particle Velocity Update . . . . .	107
5.4.3	Experiments with Benchmark Optimisation Problems . . . . .	110
5.4.3.1	Ackley Problem . . . . .	110
5.4.3.2	Rastrigin Problem . . . . .	111
5.4.3.3	Griewank Problem . . . . .	114
5.4.3.4	Rosenbrock Problem . . . . .	116
<b>6</b>	<b>Robust Evaluation Model</b>	<b>119</b>
6.1	Incremental Relaxation by Fast March Method . . . . .	120
6.1.1	Fast March Method . . . . .	123
6.1.2	Experiments . . . . .	126
6.2	Colour and Texture Incorporation . . . . .	128
6.2.1	Illumination Invariant Colour Difference . . . . .	133
6.2.2	Experiments . . . . .	136
6.3	Maximisation of Mutual Information . . . . .	140



---

6.4	Gradual Sampling for Annealed Particle Filter . . . . .	143
6.4.1	Connection between Gradual Sampling and Annealing Variable .	147
6.4.2	Experiments and Discussion . . . . .	149
<b>7</b>	<b>Compressive Evaluation</b>	<b>155</b>
7.1	Compressive Sensing . . . . .	156
7.1.1	Signal Sparse Representation . . . . .	157
7.1.2	L1 Minimisation and Reconstruction . . . . .	158
7.1.3	Incoherence Sampling . . . . .	159
7.1.4	Restricted Isometry Property . . . . .	161
7.1.5	RIP Random Sensing . . . . .	162
7.2	Discrete Wavelet Transform . . . . .	163
7.3	Compressive Annealed Particle Filter . . . . .	166
7.3.1	Restricted Isometry Property and Pairwise Distance Preservation	168
7.3.2	Multilevel Wavelet Likelihood Evaluation on Compressive Mea- surements . . . . .	170
7.3.2.1	Construct Increasing Wavelet Coefficient Image . . . . .	171
7.4	Experiments . . . . .	172
<b>8</b>	<b>Conclusion</b>	<b>177</b>
<b>A</b>	<b>Appendix</b>	<b>181</b>
A.1	Perspective Projection . . . . .	181
A.2	Importance Resampling . . . . .	183
A.3	Human Body Segments and Joint Angle Ranges . . . . .	185
A.4	Parameterisations of Three DOF Rotations . . . . .	185
A.4.1	Rotation Matrices . . . . .	185
A.4.2	Euler Angles . . . . .	191
A.4.3	Axis Angle . . . . .	192
A.4.4	Optimisation on Axis Angle . . . . .	193
	<b>Bibliography</b>	<b>197</b>



# Introduction

---

Human motion capture can be considered an evolution in human computer interaction that imparts understandability of human movement to computers. The traditional human interaction devices – keyboards, mice and game controllers – confine movements of human beings to limited and specific ranges so that the movements are simple enough to be detected. By contrast, human motion capture offers a powerful and flexible way to detect and recognise diverse movements in various circumstances of daily life. The development of such flexible techniques leads to an evolution of human computer interaction towards something much closer to the natural behaviour of human beings, enabling the next generation of human computer interfaces.

In the 1970s and 1980s, human motion capture was already used in biomechanics, and later expanded into education, training and sports. For instance, Calvert et al. [Calvert et al. 1982] at Simon Fraser University, attached potentiometers to a body and used the output to animate a computer character for choreographic studies and clinical assessment of movement abnormalities. The studies were limited to use attachable-sensor enhanced techniques. After the 1980s, optical capture technology advanced, and an increasing number of optical motion capture emerged in various fields, including computer animation, clinical medicine, virtual reality and the film industries. Such techniques require that the performer wears reflective markers that are captured by multiple cameras over time. The positions, angles, velocities and trajectories of the markers are then computed. The goal of this motion capture is to detect and record the motion and expression of moving subjects, which can be represented as poses of the subjects at any time, and then converted to abstract digital format.

Provided adequate computational power and number of cameras, recent marker based commercial systems are already matured enough to capture complex movements in real time. However, dedicated hardware and multiple expensive cameras are not suitable for deployment in an everyday environment. Markers may also restrict performer spontaneity, affecting realistic motion capture. Recent efforts [Hou et al. 2007; Vlasic et al. 2007; Sminchisescu et al. 2007; Rosenhahn et al. 2007; Agarwal and Triggs 2006; Balan and Black 2006; Lee and Elgammal 2006; Wang et al. 2008] in computer vision therefore have been focused on markerless human motion capture in order to realise a cost-effective and easily deployed motion capture system. Markerless motion capture has several desirable characteristics compared with marker-based

motion capture:

1. *Absence of markers*: This is the most desirable difference from the marker based system. The obvious imposed condition is that computers should recognise natural human actions, and this implicitly opens a very broad range of recognition applications in real life.
2. *Basic hardware requirements*: The common personal computer should be qualified for use as markerless motion capture system. In other words, markerless motion capture should be usable wherever a common personal computer is available.
3. *A distributed camera system*: This is the only requirement that needs some effort to setup and calibrate. However, with the availability of new Time-of-Flight camera technology, the number of cameras will be dramatically reduced. It is possible that a stereo camera will be sufficient for markerless motion capture in the near future.

## 1.1 Problem Formalisation

In this research, most attention is focused on optical and markerless motion capture using computer vision techniques. In particular, in an indoor setting with a few distributed cameras, the performer wears common clothes<sup>1</sup> and performs in the space visible from all cameras. While the performer is acting, his/her performance is recorded and stored as digital images or videos from different angles. Later, these digital images or videos are processed by computer vision techniques to reconstruct the original motion and convert it into abstract digital form.

The original motion is reconstructed by analysing each observed image and video frame in the sequence, in order to extract gesture features, estimate the best pose of the performer at a particular time, and eventually recover successive motion frame by frame. This essentially describes markerless human motion capture as an inference process that infers human pose and motion according to available observations through time. Within the context of this work, there are several key terms in markerless motion capture that should be clarified: 1) human motion, pose, and parameterisation; 2) temporal dependency; 3) the observation and its representation; and 4) markerless human motion capture with respect to the other definitions.

1. *Human Motion, Pose and Parameterisation*: Human motion itself is a complex process involving interactions of muscles, bones, external forces and other factors. To simplify it, we assume it only consists of a sequence of poses, each of which refers to certain time. Therefore, a description of human motion only requires us to be able to describe human pose. Naturally, the anatomy of the human

---

<sup>1</sup>But clothes should be a reasonable fit to his/her body shape and their colour should have a reasonable contrast with the background

body allows a definition of human pose using an articulated skeleton and its joint angles. The articulated skeleton is modelled based on anthropometric measurements of the real performer. Given this articulated skeleton, human pose is determined by a pose vector that includes a position, an orientation and joint angles. Human motion can then be described by a sequence of successive pose vectors over time.

2. *Temporal Dependency*: If human pose can be considered a description of the spatial structure of the human body, human motion naturally can be considered a description of both the spatial and the temporal structure of the human body. Over time, poses are not independent of each other, but each pose somehow depends on previous poses. In other words, a pose is a function of previous poses. This is so-called temporal dynamics or the dynamic function.
3. *Observation*: The observation is often regarded as the observable measurement or emitted signal that reflects the underlying pose. It may contain redundant and noisy information, and have various formats depending on the specific situation. In the context of this work, the observation often refers to the observed image or features extracted from the image.
4. *Markerless Human Motion*: The essential problem is finding the best poses that fit observations at each time, while maintaining consistency with previous poses.

## 1.2 Challenges

Generic image-based object detection and tracking techniques have been widely studied in the past several decades. Significant achievements in computer vision have already been seen as a consequence. Since such techniques inherit advantages of generalisations for coping with the characteristics of regular objects, they are able to handle various objects and situations. However, this appears not very fruitful and less effective when problems require more precise information and domain-specific knowledge. For example, markerless motion capture requires that each joint position and rotation of a skeleton be tracked. This type of problem has specialised characteristics and structures which cause difficulties for the generic techniques. Furthermore, markerless motion capture requires dealing with the irregular articulated structure, high dimensionality of human motion, variations of body shape among different people and interactions with the environment. These complications suggest the problem of human motion capture has to be modelled separately and attacked differently. This section outlines the four specific challenges in human motion capture.

### 1.2.1 Ambiguity and Multimodality

In the marker-based approach, target tracking is simplified, since markers have illuminative significance and known relative transformations with respect to the pose. Thus,

marker based motion capture has relatively better behaviour that tracks well-specified illuminative marker points as targets. Conversely, markerless motion capture does not have a well specified but rather poorly defined tracking target – the entire performer’s body – which has an irregular and deformable shape, and usually does not have the visual significance from the image processing point of the view. Moreover, the joint location and orientation must be inferred from indirect observation of clothes and body shape. Therefore, detecting and identifying the tracking target in markerless motion capture is much more challenging than marker-based approaches.

Common approaches in markerless motion capture rely on the shape-from-silhouette [Laurentini 1994] concept to detect the tracking target. A bounding geometry of the original 3D shape, the so called visual hull, can be determined by intersecting generalised cones that are formed by back-projecting each multi-view performer silhouette with its camera parameters. However, without a special setup<sup>2</sup>, silhouette segmentation may misclassify some background pixels with some foreground pixels. Even when accurate silhouettes can be acquired, with a small number of camera view silhouettes, multiple distinctive postures may still correspond to the same visual hull or same set of silhouettes. This is because the visual hull does not uniquely determine one posture, but rather it encompasses the maximum volume of the underlying object. As a result, there is considerable difficulty in determining a unique posture from a given observation. Another consequence of using a small number of cameras is that self-occlusions are likely to occur, resulting in gesture ambiguities. These together cause multimodality in the evaluation between the observations and a hypothetical pose. When multimodality (e.g. Figure 1.1) is present in the optimisation of the evaluation function, the local landscape of the evaluation function does not always have shape consistent with the landscape of the global optimum, like the convex function does. This makes markerless motion capture an ill-conditioned problem. Many methods, for example the gradient-based method, utilise the shape of the local landscape and would be easily trapped into local optimums. Hence, stochastic optimisation is often chosen to avoid local optimums.

Generally speaking, ambiguities occur because inadequate constraints and information are utilised. Considering an underdetermined system of linear equations, where the number of equations is smaller than the number of the unknowns, the unknowns have multiple solutions. Multiple solutions for each unknown can be regarded as ambiguous. If extra linearly independent equations can be introduced so that the number of equations is equal to the number of unknowns, then unknowns will have a unique solution, and ambiguity will be resolved. Similarly, if additional camera views can be introduced in markerless motion capture, multimodality can be resolved.

---

<sup>2</sup>For example, chroma keying lighting and background can be used to distinguish the foreground object to improve the quality of silhouette segmentation.

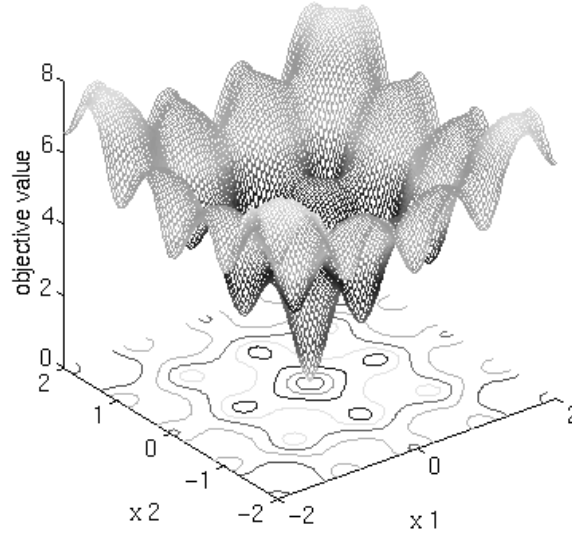


Figure 1.1: Multimodality in the two dimensional Ackley function [Ackley 1987]

### 1.2.2 High Dimensionality

Because the complexity of the anatomical structure of the human body, the pose vector usually resides in high dimensional space which often has more than 20 dimensions. This makes markerless motion capture suffer from the “curse of dimensionality” in the size of the search space, in which finding the solution increases exponentially with respect to dimensionality. As the relationship between the observed image and the pose is entangled with skin deformation, clothing colour, light reflection and other factors, the complete mapping function between them is hard to unfold using current representation techniques (e.g. manifold learning). The common solution in the literature is stochastic optimisation, for example using Markov chain Monte Carlo sampling to estimate maximum a posteriori (MAP) the observation likelihood. This requires dense sampling to cover a relatively small area [Sminchisescu and Triggs 2003] in which the global optimum has high probability of occurring. Both of the terms “small area” and “dense” are meant relative to the size of the search space. When the search space is high dimensional, the “small area” becomes massive, and “dense sampling” becomes infeasible. As a consequence, common sampling techniques become very sensitive and vulnerable to the coverage and number of samples, and they are not well scalable especially when the tracking subject has fast movement.

High dimensionality indirectly demands a good representation of the problem so that the problem can be solved by simple and effective algorithms. Moreover, it also demands that the evaluation can be quickly performed due to the fact that a large number of evaluations are often required for optimisation.

### 1.2.3 Subject Specific Modelling

When the precision required by human tracking reaches the level of joint angles and joint positions, it becomes necessary to consider subject specific modelling and anthropometric measurements. Although 3D laser scanning is able to provide accurate full body modelling, there are practical difficulties with using it on multiple people and with frequent clothing changes, and it is an additional equipment requirement. Moreover, the limb lengths of the tracking subject are also necessary for markerless motion capture, and 3D laser scanning cannot provide them. All in all, we require human body specific modelling which is suitable for markerless motion capture.

Additionally, interactions between external forces and the human body are often ignored to simplify the complex process. Hence human body movement can be formulated as rigid-body kinematics of articulated segments with joint angles. Since 3D joint rotation parameterisation involves mapping between a flat Euclidean space and a spherical space – Special Orthogonal Group ( $SO(3)$ ) – there always exists singularities and double cover in such a parameterisation. A suitable optimisation procedure has to handle these two special cases appropriately so that continuous exploration can be maintained everywhere.

### 1.2.4 Computational Performance

Most human computer interaction applications require instant responses and real time feedback, otherwise users could easily fail to engage with the immersive environment. Human motion capture as a human computer interfacing technique often needs to respond in excess of 25 times per second. This is a crucial requirement for all human computer interfacing applications. However, stochastic optimisation methods used in markerless motion capture suffer from issues associated with convergence rate and computational time. Many of them are practically infeasible when the dimensionality of problems is very high, since they have an exponential or higher order polynomial time complexity. The execution time of current algorithms for each frame on an average personal computer may take several seconds or even several minutes, several orders of magnitude slower than the real time requirement. Thus, designing an optimisation process specialised to human motion capture and improved computational performance is a very challenging task.

## 1.3 Thesis Outline

In this section, an overview is outlined for each chapter which proposes different approaches to challenge difficulties presented in the previous section.

1. Chapter 1 *Introduction*: We describe the motivation for markerless motion capture as an evolutionary development and briefly review the history of motion capture, followed by the problem formalisation which clarifies the key terms



and give formal explanations of markerless motion capture. Then, the major challenging issues in markerless motion capture are addressed. Finally, an overview of the thesis structure is presented.

2. Chapter 2 *Literature Review*: Recent state of the art studies in markerless motion capture are reviewed. The strengths and weaknesses of diverse approaches are analysed in depth, including generative, learning based approaches, and tracking by graphical image segmentation. Finally, we motivate our approach from the analysis of existing studies.
3. Chapter 3 *Architecture Overview*: An overview of the proposed framework is depicted, and the major components' characteristics and functionality are described related to overcoming the challenging issues. Major contributions to the architecture will be highlighted by referring to particular chapters.
4. Chapter 4 *Subject Specific Body Shape Modelling and Automatic Initialisation*: We begin with describing the generic articulated skeleton and standard skin deformation method used in the computer graphics community. Then, based on the skin deformation technique, we introduce two methods, Needle based and Data-Driven Body Shape Parameterisation to realise subject specific modelling. Finally, we describe how to automatically estimate the subject shape parameters, anthropometric measurements and the initial posture simultaneously by using our CMA-Annealing method.
5. Chapter 5 *Nature Inspired Global Optimisation*: To deal with multimodality and high dimensionality, global optimisation, Simulated Annealing, Particle Swarm Optimisation and Covariance Matrix Adaptation Evolution Strategy are described. Considering the characteristics and properties of these methods, we propose a novel hybrid optimisation method—Covariance Matrix Adaptation Annealing. It takes advantages of both fast convergence and robustness to multimodality to attack one specific class of problems. At the end of the chapter, four methods' behaviour and performance are validated by experiments with a series of benchmark multimodal functions.
6. Chapter 6 *Robust Evaluation Model*: We propose several improvements to the evaluations of annealing-based optimisation in this chapter. These include details of Incremental Relaxation by the Fast March Method, Colour and Texture Incorporation, Maximisation of Mutual Information and Gradual Sampling. These techniques improve the original algorithm from three perspectives—precision, robustness and computational performance.
7. Chapter 7 *Compressive Evaluation*: Noticing sparsity in the observation likelihood evaluation, we use the compressive sensing [Candès et al. 2006; Candès and Tao 2006; Candès and Wakin 2008; Donoho 2006] technique to eliminate irrelevant error terms in the observation likelihood. Multilevel wavelet decomposition is seamlessly integrated into the annealing schedule so that both computational speed and tracking accuracy are dramatically boosted.

8. In Chapter 8 *Conclusion and Discussion*: Conclusions are drawn by summarising the contributions of this thesis. Further discussions concerning limitations, remaining issues and possible future research are presented.

---

# Literature review

---

Many different methods have been used to perform markerless motion capture. These can be classified into two major approaches: 1) methods modelling image features (eg. silhouettes and edges) as a generative process from pose, and 2) methods based on learning regression from image observations of pose. The generative methods model image feature space as a function of pose, and inferring pose from a limited number of view based image features often involves solving a hard multimodal optimisation problem. Learning based techniques try to model the pose configuration as a function of image observations. However because of the problem's multimodality, such functional is very difficult to learn. The methods reviewed in this chapter will present diverse approaches to overcoming these issues. Apart from these two approaches, many works integrate image segmentation into the pose estimation process. This allows image segmentation and pose estimation to be improved iteratively, as the image is segmented with the current best pose estimate and the pose is then estimated based on the improved segmentation results.

## 2.1 Global Optimisation and Template based Generative Approach

Here we review recent template based methods that handle markerless motion capture in multi-view settings. This category of methods combines different basic components into the tracking pipeline. These methods implement different ideas to overcome challenges in markerless motion capture, which include interactive simulated annealing, covariance scale sampling, loose-limb using graphical models and articulated ICP.

## 2.1.1 Global Optimisation

### 2.1.1.1 Annealed Particle Filter

Deutscher et al's studies [Deutscher et al. 1999; Deutscher et al. 2000; Deutscher et al. 2001; Deutscher and Reid 2005] have developed a stochastic scheme, annealed particle filtering (APF), that incorporates particle filtering [Doucet et al. 2000] with simulated annealing to effectively search high dimensional space. Unlike the Kalman filter which can be provably optimal for sequential propagation of Gaussian probability densities (though it fails catastrophically in non-linear settings), particle filtering is able to approximate arbitrary densities and propagate multiple hypotheses, leading to much better behaviour in non-linear settings. Its robustness was demonstrated by the Condensation algorithms [Isard and Blake 1998a] in the context of visual tracking. However, in high dimensional space occurring in markerless motion capture and other disciplines, particle filtering experiences serious scalability problems, both in populating space and representing an arbitrary density using a manageably sized particle set. In fact it has been shown by MacCormick [MacCormick and Isard 2000] that the number of particles  $N$  has to satisfy  $N \geq D_{min}/\alpha^d$ , where  $D_{min} \in [1, N]$  denotes the minimum acceptable survival diagnostic and  $\alpha \ll 1$  is a survive rate powered by the number of the dimensions  $d$ . Clearly, when  $d$  is large, the number of particles becomes infeasible. This is essentially caused by the fact that importance sampling scales badly with dimension [MacKay 1998]. Moreover, the evaluation on the entire multimodal posterior distribution in high dimensional space is also computationally prohibitive. Therefore, Deutscher et al proposed a Simulated Annealing procedure to only approximate the global mode of the posterior distribution by estimating a Maximum A Posteriori of the observation likelihood.

The Simulated Annealing algorithm was proposed by Kirkpatrick et al [Kirkpatrick et al. 1983] to solve multivariate and combinatorial optimisation. Its process is analogous to iteratively attaining thermal equilibrium in statistical mechanics. Initially at high temperature, the particles with energy  $E_0$  are free to move into the new state with the energy  $E$ . New states are accepted if  $\delta E = E - E_0 < 0$  otherwise the acceptance probability is equal to  $\exp\{-\delta E/(k_b T)\}$  where,  $k_b$  denotes the Boltzmann constant and  $T$  denotes temperature. This is the so called Metropolis criterion. With the temperature gradually decreasing, the range of particles gradually concentrates on the states with the lowest energy, as high energy states become increasingly unlikely. Eventually,  $T$  becomes so low that the system "freezes", and if the temperature is lowered sufficiently slowly this frozen state will have minimum energy.

With Simulated Annealing, APF has a higher probability of escaping from local optimums. However, it is still computationally intensive. To reduce the computational time and effectively distribute particles according to the kinematic structure of the human figure, Deutscher et al imposed soft hierarchical partitioning by scaling the perturbation covariance matrix  $\mathbf{P}_m$  at the layer  $m$  and time  $t$ , proportional to the

covariance of the current particle set  $\{\mathbf{x}_{t,m}^i\}_{i=1}^N$ .

$$\mathbf{P}_m \propto \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_{t,m}^i - \bar{\mathbf{x}}_{t,m})(\mathbf{x}_{t,m}^i - \bar{\mathbf{x}}_{t,m})^T$$

Inspired by genetic algorithms, they also incorporate a cross operation to APF, resulting in further improvement.

### 2.1.1.2 Covariance Scaled Sampling

Theoretically, sufficiently dense sampling can approximate arbitrary distributions in high dimensions, but it is computationally prohibitive in practice. Therefore, reasonably dense coverage of the entire probability distribution may not be feasible, especially in high dimensional space. Methods like the particle filter are able to demonstrate a certain effectiveness in approximating the probability distribution in low dimensional space, but are very likely to miss the global mode during the iterative procedure over time [Sminchisescu and Triggs 2003]. Sminchisescu et al believed that random perturbation in the stochastic approach is an effective search strategy only for relatively low dimensional problems, where samples can cover the surrounding neighbourhood fairly densely. In high dimensions, volume increases very rapidly with radius, so random samples must be extremely sparse to cover the global mode (since any two modes far from each other). Hence, samples are very unlikely to hit the small core of the mode surrounding the local optimum, and are more likely to remain in the non-mode area. This is fatal for approaches of the importance resampling class: samples in non-mode areas are very unlikely to be resampled, so the new mode is almost certain to be missed.

Sminchisescu et al [Sminchisescu and Triggs 2003] proposed an approach using a combination of local optimisation and global controlled search to address the above issues by 1) wide tail sampling to gain large coverage, 2) adaptive local covariance scaling to focus on the important sample space, 3) and local optimisation in order to guarantee samples fall into the modes. For local optimisation, a second order Newton's method with a trust region (whose descent direction is chosen by solving a regularised subproblem) is used to accelerate sample convergence to the local optimums. At each iteration, the log-likelihood gradients and Hessians of the posterior are calculated to build proposed quadratic model  $q(x)$  imposed constraints.

Sminchisescu et al also believed that the locations of modes are based on underlying rules determined by particular domain knowledge. A sensible sampling technique should adapt this domain knowledge in order to reduce the sampling space. Covariance scaled sampling inherits this principle. It moderately inflates the local shape impact by rebuilding the covariance matrix along the most important directions, so that it removes sampling wastage along other directions. The local shape is formed from the previous posterior and temporal dynamics models. The important

directions usually point out the potential locations of modes, so covariance scaled sampling always scales the sampling space (illustrated in Figure 2.1) with respect to the shape of local distribution such that it has better chance to sample the locations having modes.

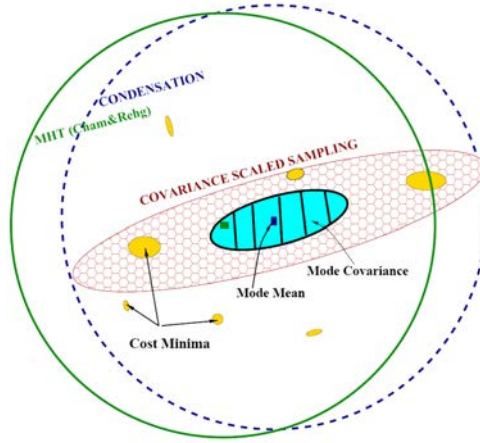


Figure 2.1: Sampling space (courtesy of [Sminchisescu and Triggs 2003])

In their study, they assume that the importance distribution and the posterior distribution are constructed by Gaussian mixtures. In this case Gaussian mixtures are fit to multimodality in high dimensional space. The particle is generalised as the Gaussian distribution  $x_t^i = \mathcal{N}(\mu_t^i, \Sigma_t^i)$ , and then the Particle Filter with covariance scaled sampling is outlined in Algorithm 1:

### 2.1.1.3 Interacting Simulated Annealing and Hybrid Approach

In Gall et al's work[Gall et al. 2007], an Interacting Simulated Annealing (ISA) algorithm was proposed based on the FeynmanKac model [Kac 1949], which is a more general form of the APF paradigm. They also showed APF is a special case of ISA where the "Updating" and "Resampling" steps are replaced by the Interacting Simulated Annealing algorithm (shown in Algorithm 2) with  $\epsilon_t = 0$ .  $\epsilon_t$  satisfies  $\epsilon_t \geq 0$  and  $\epsilon_t \|\pi_t\|_\infty \leq 1$ , where functions  $\pi_t$  are often unnormalised BoltzmannGibbs measures  $\pi_t(\mathbf{x}) = \exp(-\lambda E(\mathbf{x}))$ . It is interesting to note that the parameters  $\epsilon_t$  are allowed to depend on the current distribution. Their ISA algorithm is another special case that sets the parameters:

$$\epsilon_t = \epsilon'_t / \langle \eta_t, \pi_t \rangle \quad 0 < \epsilon'_t \leq 1/g$$

$$g = \sup_{t \in \mathbb{N}_0} \left( \sup_{\mathbf{x}, \mathbf{y} \in [0,1]} \left( \frac{\pi_t(\mathbf{x})}{\pi_t(\mathbf{y})} \right) \right) < \infty$$

---

**Algorithm 1** Covariance Scale Sampling

---

```

for  $i=1$  to  $N$  do
    Construct covariance matrix  $\widehat{\Sigma}_{t-1}^i$ , eigen-decompose  $\Sigma_{t-1}^i$ , select its  $k$  most uncertain eigenvectors  $v_j$ . Then  $\widehat{\Sigma}_{t-1}^i = \sum_{j=1}^k \lambda_j v_j v_j^T$ .
    Calculate importance distribution  $\pi(x_t^i | x_{t-1}^i, y_t^i) = \sum_{i=1}^M w_{t-1}^i \mathcal{N}(\mu_{t-1}^i, s \widehat{\Sigma}_{t-1}^i)$ , where  $s$  is a scale factor from 4 to 14
end for
repeat
    Choose single Gaussian  $\widehat{x}_t^i$  from  $\pi()$  according to the weight  $w_{t-1}^i$ ,
    Sample from  $\widehat{x}_t^i$  to obtain  $s_j$ , with respect to the observation likelihood at time  $t$ , perform local optimisation beginning with  $s_j$ . At its convergence,  $\mu_t^i$  and  $\Sigma_t^i = \mathbf{H}(\mu_t^i)^{-1}$  are found if they haven't been already.  $x_t^i = \mathcal{N}(\mu_t^i, \Sigma_t^i)$ .
    if the number of modes  $< N$  then
        Reduce  $N$ 
    end if
until  $N$  samples are generated from  $\pi(x_t^i | x_{t-1}^i, y_t^i)$ 
for  $i=1$  to  $N$  do
    Calculate normalised weights by  $w_t^i = \frac{p(\mu_t^i | y_t)}{\sum_{j=1}^N p(\mu_t^j | y_t)}$ 
end for
    Select the  $K$  most important  $x_t^i$  according to their weights  $w_t^i$ , re-normalise the selected weights  $w_t^i$ 
    for  $i=1$  to  $K$  do
        Find the closest  $x_{t-1}^{closest}$  to  $x_t^i$  according to a Bhattacharyya distance.
        Modify the weight  $w_t^i = w_t^i \times w_{t-1}^{closest}$  discard  $x_{t-1}^{closest}$  from further consideration
    end for
    Re-normalise the weights, compute the posterior mixture  $p(x_t | y_{1:t}) = \sum_{i=1}^K w_t^i x_t^i$ 

```

---

as proposed in [Moral and Doucet 2003]. It turns out when the number of particles  $N$  is greater than  $g$ ,  $\epsilon'_t$  can then be set to  $1/N$ . This leads to  $\epsilon_t = 1/\sum_{k=1}^N w_t^k$  where the sequence of posterior distribution approximations has strictly smaller variances than  $\epsilon_t = 0$ . In their experiments, ISA has better convergence results with the particle approximation if  $N > g$  is guaranteed.

---

**Algorithm 2** Interacting Simulated Annealing Algorithm

---

**Require:** : parameters  $\epsilon_t$ , number of particles  $N$ , initial distribution  $\eta_0$ , weighting functions  $\pi_t$ , transition kernels  $K_t$  and observations  $y_t$ .

1. Initialisation: Draw  $N$  samples  $x_0^i$  from  $\eta_0$
  2. Selection: Calculate all particle weights using  $w_t^i = \pi_t(x_t^i, y_t)$
  - for**  $i=1$  to  $N$  **do**
    - Sample  $\kappa$  uniformly from  $[0, 1]$
    - if**  $\kappa < \epsilon_t w_t^i$  **then**
      - $\hat{x}_t^i = x_t^i$
    - else**
      - $\hat{x}_t^i = x_t^j$  with probability  $\frac{w_t^j}{\sum_{k=1}^N w_t^k}$
    - end if**
  - end for**
  3. Perturbation: Obtain new particles by  $x_{t+1}^i = K_t(\hat{x}_t^i)$  and go to 2
- 

A continued work in [Gall et al. 2008] proposes an analysis-by-synthesis framework for tracking rigid and articulated models by using region-based and patch-based matching. In region-based matching, the difference between the projected surface of the model and the real object region extracted in the image is minimised through the correspondences of the model and real object contours. The contour of the real object is extracted by level-set segmentation in which the contour is given by the zero-line of a level-set function  $\Phi$ . As shown in [Rosenhahn et al. 2007], the contour, zero-line of  $\Phi$ , can be obtained by minimising the energy function:

$$E(\Phi, \hat{x}) = - \int_{\Omega} H(\Phi) \ln p_1 + (1 - H(\Phi)) \ln p_2 dx + v \int_{\Omega} |\nabla H(\Phi)| dx + \lambda \int_{\Omega} (\Phi - \Phi_0(\hat{x}))^2 dx$$

where  $H$  is a regularised version of the step function, and  $p_1$  and  $p_2$  are the densities of the fore- and background modelled by local Gaussian densities. While the first term maximises the likelihood, the second term regulates the smoothness of the contour with the parameter  $v = 2$ . The last term penalises deviations from the projected surface of the predicted pose  $\Phi_0(\hat{x})$  with  $\lambda = 0.06$ . In patch-based matching, correspondences between two successive frames for prediction, and between the current image and a synthesised image, are found by applying PCA-SIFT [Ke and Sukthankar



2004] as a local descriptor. The local descriptor is trained for the object by building the patch eigenspace from the object's texture. Therefore pose estimation can be accomplished by solving a simple least squares problem with respect to correspondences. Since the accurate correspondences can be extracted, their experimental results with both rigid and articulated models are very robust to the accumulation of estimation errors and the tracking drift.

More recent works [Gall et al. 2009; Gall et al. 2010] focused on a hybrid approach combining local and global optimisation. Global optimisation usually is more robust to fast motions and ambiguities, but the price is that the computational effort is very high. On the other hand, local optimisation has advantages in convergence speed. If the initial pose is close to the global optimum, local optimisation is able to find a reasonable solution using much fewer evaluations. [Gall et al. 2009] finds the contour and SIFT texture correspondences, transforms 2D-2D to 3D-2D correspondences, and solves local optimisation as a weighted least squares problem. If the errors of local optimisation exceed a predefined threshold, global optimisation ISA is executed to improve results. This usually happens when the top branch of the skeleton hierarchy (torso) is not well estimated by local optimisation. In [Gall et al. 2010], ISA global optimisation is adopted at first to locate a good initial position for local optimisation, then the contour correspondences are found by an iterated closest point (ICP) approach. Finally pose estimation is refined by the least squares solution using these correspondences. These hybrid approaches show very encouraging results in diverse activity tracking and performance capture tests.

## **2.1.2 Template Model Generation**

### **2.1.2.1 Articulated ICP and Subject Specific Model Generation**

The works [Corazza et al. 2010; Corazza et al. 2009; Mundermann et al. 2007] by Stefano Corazza et al investigate the problem with a different approach which relies on

registering a subject specific model to the sequence of visual hulls. The subject specific model is built by following this procedure (also shown in Figure 2.2):

1. Rigid segments of the articulated model obtained from the SCAPE method [Anguelov et al. 2005a] are registered into the laser scanned subject mesh. The global transformation registering every body segment is estimated;
2. The scanned mesh is divided according to the different body parts with a proximity criteria check. For any point in the scanned mesh, if the closest point in the model belongs to the part  $i$ , then it is labelled as belonging to that part;
3. For every body part in the scanned mesh, the inverse of the global transformation calculated in 1 is applied. This registers the scanned mesh to the reference pose of the articulated model;
4. The above steps are repeated until convergence.

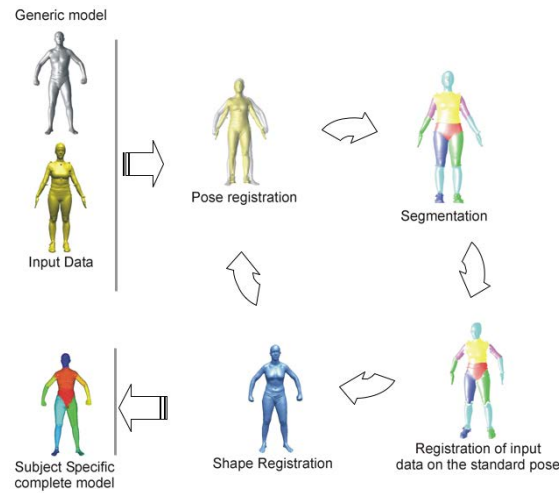


Figure 2.2: Subject specific modelling using iterative shape registration (courtesy of [Corazza et al. 2009])

The Articulated ICP presented in [Corazza et al. 2010] extends the ICP algorithm to articulated objects. Each segment with a 6 DOF joint is hierarchically connected so that the motion is propagated along the kinematic chain, unlike their previous

work [Mundermann et al. 2007] that treats each rigid segment independently. The improved algorithm allows extraction from the tracker of anatomically meaningful and rigorous data. Each point on the model surface is expressed as a function of joint parameters in the articulated model. This function is linearised so that the registration problem can be attacked by minimising a series of least correspondence residuals in the ICP paradigm:

$$H = \sum_{i=1}^N \|P_i - CP_i\|_2$$

where,  $N$  denotes the number of correspondences, and  $CP_i$  denotes a point on the scanned data which corresponds to point  $P_i$  on the model. At each iteration  $CP_i$  points are estimated through a closest point criterion with respect to points  $P_i$ . The normal on the visual hull surface in proximity of  $CP_i$  is compared to the model surface normal in proximity of  $P_i$ . If these norms differ excessively (typically more than 90 degrees), the pair of corresponding points are excluded from the minimisation problem. When provided with more than 8 camera views, their method is able to track very fast movements, including a cricket bowl, handball throw, and gymnastic flip. However when the number of cameras is less than 8, for example HumanEvaII four colour camera views, it can not track very well.

### 2.1.2.2 Performance Capture

Detailed appearances are difficult to capture but they are visually compelling and have a great impact on visual quality. Several works [Starck and Hilton 2007b; de Aguiar et al. 2008; Vlasic et al. 2009; Vlasic et al. 2008] have focused on high quality and small scale details capture, and their impressive quality proves that the accuracy of markerless motion capture is competitive with marker-based systems, and therefore adequate for studio performance capture.

In [Starck and Hilton 2007b], Starck et al proposed a fully automated surface capture system for recording a human's shape, appearance and motion from multiple

video cameras to create highly realistic animated content from an actor's performance in full wardrobe. Their method has seven distinctive steps:

1. A multiple-view video sequence of an actor's performance is recorded.
2. Chroma-key matting is performed to extract foreground silhouettes from the camera images, separating the image's foreground pixels from the known background colour.
3. An alpha matte for each image is extracted, which defines the foreground opacity at each pixel and the foreground colour where pixels in the original image are mixed between foreground and background.
4. The shape-from-silhouette technique is used to define the visual hull of the rough (maximal) volume in the scene.
5. The visual-hull defines an upper-bound on the scene's true volume and so constrains the feasible space.
6. Canny edge detection is used to find local discontinuities which correspond to surface edge features. Then, the correspondence is found between each surface feature in an image with that of an adjacent camera view. The correspondences are constrained to be located inside of the visual hull and satisfy the camera epipolar geometry defining the relationship between observations in pairs of cameras. Finally, the connected set of pixel correspondences with the adjacent view is maximised in terms of the image correlation.
7. Appearance consistencies (features, silhouettes) and temporal consistencies (between successive frames) are maximised together using a global optimisation technique-graph cut.
8. The scene is represented by a texture spherical remeshing technique, extending the work of [Praun and Hoppe 2003]. Multiple resolution blending with

spherical surface continuity is used to construct a single seamless texture. This ensures that the extent of texture blending corresponds to the spatial frequency of the image's features, preserving the higher frequency detail that can become blurred with simple linear texture-blending techniques.

The work in [de Aguiar et al. 2008] uses a detailed static laser scan of the subject as the template model. Performances are captured in a coarse-to-fine way. First, a global model pose is inferred using a lower-detail tetrahedral model. Subsequently, smaller-scale shape and motion detail is estimated based on a high-quality tetrahedral model. Global pose capture employs a new analysis-through-synthesis scheme based on image and silhouette cues. It estimates the global pose of an actor at each frame on the basis of the lower-detail tetrahedral model. From the input footage, this scheme robustly extracts a set of positions and constraints of physically plausible shape deformations in order to make the scan mimic the motion of its real-world counterpart. After global pose recovery in each frame, a model-guided multi-view stereo and contour alignment method reconstructs finer surface details at each time step. Their results show very reliable reconstructions of very complex motion exhibiting speed and dynamics that would even challenge the limits of traditional marker-based optical capturing approaches.

Vlasic et al [Vlasic et al. 2008] use a multi-view studio to acquire a set of synchronised high-definition silhouette videos by recording a performance from multiple distributed calibrated cameras. The silhouette from each viewpoint corresponds to a cone of rays from the camera origin through all points of the subject. The intersection of these cones reconstructs the subject's visual hull. They also used a template mesh rigged with a skeleton that matches the physical dimensions of the performer: the skeleton is positioned within the template mesh and each vertex is assigned a weight that is used to deform the template with linear blend skinning (LBS). Initially their method uses the visual hulls to optimise the skeletal pose of the performer so that it

positions bones deeply into the visual hull and maintains temporal smoothness. During the iteration over frames, it provides visual feedback of the frame progress to the user and allows user corrections for especially difficult frames. The user can specify constraints for joint positions, allowing for more robust tracking than is possible with fully automatic methods. The method deforms a template mesh of the performer to fit the recovered pose and silhouettes at each frame. Then an iterative algorithm is used for non-rigid shape matching using Laplacian coordinates [Alexa 2003]. It begins with a smoothed version of the LBS mesh as the initial guess. At each iteration, it reintroduces part of the original template detail and vertex position constraints to bring the shape closer to the contours in each camera. To enhance temporal consistency, their method interleaves a bilateral filter on the meshes for each iteration. The resulting shapes match the silhouettes while still resembling the undeformed template. This allows the preservation of detail in the template and capture of secondary deformation, such as flapping clothing, that makes the motion appear natural. The quality of their output is suitable for video editing because the template ensures frame-to-frame correspondence.

The study [Vlasic et al. 2009] describes a novel hardware system by Charles-Felix et al [Chabert et al. 2006] which creates active illumination. The lighting hardware consists of the top two-thirds of an 8-metre, 6th-frequency geodesic sphere with 1,200 regularly-spaced and individually controllable light sources, of which 901 are on the sphere and the rest are placed on the floor. A central area is reserved for the subject. Eight Vision Research V5.1 cameras are placed on the sphere around the subject, at an approximate height of 1.7 metres relative to the central performance area. Then image processing algorithms are used to obtain high-quality normal maps and silhouettes from multiple viewpoints at video rates. The surface reconstruction algorithms process this data to derive high-quality mesh sequences. The resulting mesh sequences can be used in biomechanics to analyse complex motions, in computer games to create next-generation characters, and in movies to create digital doubles.

### 2.1.3 Object Localisation and Pose Estimation using a Graphic Model

In the work by [Sigal et al. 2004; Sigal and Black 2006b], a loose-limbed representation for the human body similar to a graphical model was presented, in which limbs are connected via learned probabilistic constraints, facilitating initialisation and failure recovery. The tracking and pose estimation problem is formulated as one of inference in the graphical model, and belief propagation is used to estimate the pose of the body at each image frame. Each node in the graphical model represents the 3D position and orientation of a limb in Figure 2.3. Undirected edges between nodes represent statistical dependencies and these constraints between limbs are used to form messages that are sent to neighbouring nodes in space and time. Additionally, each node has an associated likelihood defined over a set of image features. The combination of highly non-Gaussian likelihoods and a six-dimensional continuous parameter space (3D position and orientation) for each limb makes standard belief propagation algorithms infeasible. Conditional probabilities relating the 3D pose of connected limbs are learned from motion captured training data. Similarly, probabilistic models for the temporal evolution of each limb (forward and backward in time) are also learned. The human pose and motion estimation are then solved with non-parametric belief propagation [Isard 2003; Sudderth et al. 2003] using a variation of the Particle Filter that can be applied over a general loopy graph.

The main advantages in this approach are that the complexity of the search task is linear rather than exponential in the number of body parts, and bottom-up limb and head detection are integrated at every frame allowing automatic initialisation and recovery from transient tracking failures. However, the loose-limbed representation has difficulty in handling poses where one limb penetrates another. When self-occlusions appear, multiple limbs may fit the same image region resulting in incorrect pose estimations that poorly explain the overall image observations. Occlusion-sensitive local image likelihoods were introduced by [Sigal and Black 2006b] that approximate the

global likelihood by accounting for occlusions and competing explanations of image observations by multiple limbs. Since occlusion reasoning involves interactions between non-adjacent body parts which create loops in the graphical model structure representing the body, a variant of approximate belief propagation (BP) that is able to infer the real-valued pose of the person in 2D was introduced.

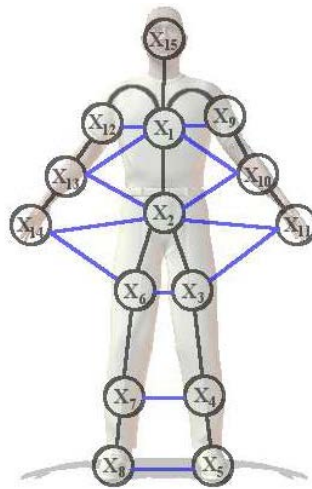


Figure 2.3: Graphic model of the articulated human structure: Nodes represent limbs and arrows represent statistical dependencies between limbs. Black and blue edges correspond to kinematic and interpenetration constraints, respectively. (courtesy of [Sigal 2008])

## 2.2 Learning Based Approach

In this section, we review several learning based methods in markerless motion capture, including the relevance vector machine, Gaussian process dynamical models and dimensionality reduction techniques to identify low dimensional representations.

### 2.2.1 Tracking by Regression

[Agarwal and Triggs 2006; Agarwal and Triggs 2004a; Agarwal and Triggs 2004b] take a learning-based approach to attack markerless motion capture in the monocular view setting, using regression as a basic tool to distill a large training database of



---

3D poses and corresponding images into a compact model that has good generalisation to unseen examples. They use a bottom-up approach in which the underlying pose is predicted directly from a feature-based image representation, without directly modelling the generative process of image formation from the body configuration. The method is purely data-driven and does not make use of any explicit human body model or prior labelling of body parts in the image. The pose and observed image are represented by the vectors  $\mathbf{y}$  and  $\mathbf{x}$ , respectively. Given the high dimensionality and intrinsic ambiguity of the monocular pose estimation problem, the active selection of appropriate image features is critical for success. They use the training images to learn suitable image representations specific to capturing human body shape and appearance. Two kinds of representation have been employed. 1) When the silhouette is available, the silhouette is encoded in terms of the distribution of its softly vector quantised local shape context descriptors [Belongie et al. 2002], with the vector quantisation centres being learned from a representative set of human body shapes. This transforms each silhouette to a point in a 100D space of characteristic silhouette shapes. Such an encoding allows 3D body pose to be recovered by using direct regression on the descriptors. 2) When the silhouette is not available, the observed image is used to directly compute histograms of gradient orientations on local patches densely over the entire image (using the SIFT and HOG descriptors). These are then re-encoded to suppress the contributions of background clutter using a basis learned using Non-negative Matrix Factorisation [Lee and Seung 1999] on training data.

The pose recovery problem reduces to estimating the pose from the vectorial image representation. Given a set of labelled  $n$  training examples  $(\mathbf{x}, \mathbf{y})$ , the Relevance Vector Machine [Tipping 2001] is used to learn a smooth reconstruction function  $\mathbf{y} = r(\check{\mathbf{y}}, \mathbf{x})$ , valid over the region spanned by the training points. This function directly encodes the inverse mapping from image to body pose. The forward mapping from body pose to image observations can be more easily explained by projecting a human body model or learning image likelihoods. The function is a weighted linear combina-

---

tion  $r(\mathbf{x}) = \sum_k \alpha_k \phi_k(\check{\mathbf{y}}, \mathbf{x})$  of a presanctified set of scalar basis functions  $\{\phi_k(\check{\mathbf{y}}, \mathbf{x}) | k = 1 \dots p\}$ , where,  $\alpha_k$  denotes the weight vectors. They are damped to control over-fitting, and sparse in the sense that many of them are zero. Sparsity is ensured by the use of the Relevance Vector Machine that actively selects the most “relevant” basis functions. At each time step  $t$ , a state estimate  $\check{\mathbf{y}}$  is obtained from the previous two pose vectors using an autoregressive dynamical model. Incorporating this preliminary pose estimate  $\check{\mathbf{y}}$  to the regression model helps to maintain temporal continuity and to disambiguate pose in cases where there are several possible reconstructions.

### 2.2.2 Gaussian Process Dynamical Model

The prior dynamic model of pose and motion plays a central role in 3D monocular human tracking. It can resolve problems caused by ambiguities, occlusions, and image measurement noise. Effective models for human tracking can be learned using the Gaussian Process Dynamical Model (GPDM) [Wang et al. 2006], even when modest amounts of training data are available. The GPDM is a latent variable model with a nonlinear probabilistic mapping from latent positions  $\mathbf{x}$  to human poses  $\mathbf{y}$ , and a nonlinear dynamic mapping on the latent space. It provides a continuous density function over poses and motions that is generally non-Gaussian and multimodal. Given training sequences, one simultaneously learns the latent embedding, the latent dynamics, and the pose reconstruction mapping. With Bayesian model averaging a GPDM can be learned from relatively small amounts of data, and it generalises gracefully to motions outside the training set. [Urtasun et al. 2006a] proposed a balanced GPDM for learning smooth models from training motions with stylistic diversity, and showed that they are effective for monocular human tracking. Therefore, the tracking problem is formulated as a MAP estimator on short pose sequences in a sliding temporal window. Estimates are obtained with deterministic optimisation, and look remarkably good despite very noisy, missing or erroneous image data and significant occlusions.

---

In the line of GPDM, [Ek et al. 2008] takes a learning based approach where it models both silhouette observations, joint angles and their dynamics as generative models from shared low dimensional latent representations using the Gaussian Process Latent Variable Model [Lawrence 2005]. Each image is background subtracted to get its silhouette. As in [Agarwal and Triggs 2006] each silhouette is represented using shape context histograms [Belongie et al. 2002]. Each contour is subsampled with one pixel spacing, acquiring about 100 - 150 histograms for each image. To reduce dimensionality of the descriptor and remove the effects of ordering, they vector-quantise the histograms using K-means clustering, resulting in a 100D silhouette descriptor. Generative methods model the space of silhouettes as a function of pose. This correctly reflects the structure of the problem as each silhouette could have been generated by several different poses but each pose can only generate one single silhouette. Learning a low-dimensional representation of the pose is not required, so it is not necessary to fall-back on approximative methods for solving the inverse of this generative mapping. Therefore the latent representation reflects the dynamics of the data and can predict poses over time in a simple manner. The model requires no manual initialisation when predicting sequential data but automatically initialises from training data.

Wang et al's approach [Wang et al. 2008] is also inspired by the Gaussian process latent variable model [Lawrence 2005]. The GPLVM models the joint distribution of the observed data and their corresponding representation in a low-dimensional latent space. It is not, however, a dynamical model; rather, it assumes that data are generated independently, ignoring temporal structure of the input. Here the GPLVM is augmented with a latent dynamical model, which gives a closed-form expression for the joint distribution of the observed sequences and their latent space representations. The incorporation of dynamics not only enables predictions to be made about future data, but also helps to regularise the latent space for modelling temporal data in general. The unknowns in the GPDM consist of latent trajectories and hyperparameters. Generally, if the dynamics process defined by the latent trajectories is smooth, then

the models tend to make good predictions. Initially a maximum a posteriori (MAP) algorithm is introduced for estimating all unknowns. To learn smoother models, three alternative learning algorithms are used, namely, the balanced GPDM [Urtasun et al. 2006a], manually specifying hyperparameters, and a two-stage MAP approach. These algorithms present trade-offs in efficiency, synthesis quality, and generalisability.

### 2.2.3 Dimensionality Reduction and Manifold Learning on Visual Tracking

Human motion capture involves searching in high dimensional space (often more than 30 dimensions). Unavoidably, it suffers from the “curse of dimensionality”. That is, the computational complexity of searching the entire space increases exponentially as a function of dimensionality. One possible way to deal with the “curse of dimensionality” is to carefully design algorithms to explore the solution in a sensible and efficient way. An alternative way is by decreasing the dimensionality of the original problem and solving the problem in a lower dimension space. The class of these approaches is often referred to as dimensionality reduction. Dimensionality reduction is constructed on the premise that many high dimensional problems have a considerable amount of redundancy and irrelevance to the solution, with only a portion of information crucial for the solution. In fact, lower dimensional variables are often most necessary to present and describe the intrinsic information of problems. These lower dimensional variables are usually called latent or hidden variables. Latent variables usually can be figured out by mapping original data from high dimensional space to low dimensional space via some function. Mathematically, where  $x_i \in R^d$  denotes latent variables,  $y_i \in R^D$  ( $i = 1 \dots N$ ) denotes the original data in the observation space, there is some function  $f$  such that:

$$y_i = f(x_i, \Theta) + \sigma_i$$

---

where,  $\sigma_i$  is random noise, and  $\Theta$  are parameters of the function  $f$ . The objective is to estimate the appropriate parameters  $\Theta$ . This resembles the canonical parametric regression framework.

Despite the high dimensionality of pose configuration space, many human activities lie intrinsically on low dimensional manifolds. Exploiting this property is essential to reveal the high degree of correlation in human motion and constrain the solution space for tracking and posture estimation. There are many studies [Tangkuampien and Suter 2006; Lee 2007; Elgammal and Lee 2009] that show interest in learning low dimensional representations of the pose configuration manifolds.

[Tangkuampien and Suter 2006] proposed an efficient technique based on kernel principal component analysis (KPCA) [Bernhard et al. 1999], which is defined for out-of-sample points. KPCA is used to learn two feature space representations, which are derived from the synthetic silhouettes and relative skeleton joint positions of a single generic human mesh model. After training, novel silhouettes of previously unseen actors (and of unseen poses) are projected through the two manifolds using Locally Linear Embedding [Roweis and Saul 2000] reconstruction. The captured pose is then determined by calculating the pre-image [Scholkopf et al. 1998] of the projected silhouettes. An inherent advantage of KPCA is its ability to de-noise input images before processing, as shown in [Bernhard et al. 1999] with images of handwritten characters. There is, however, no previous work on the de-noising of human silhouettes for human motion capture using the KPCA projection. This novel concept allows the inference of relatively accurate poses from noisy unseen silhouettes by using only one synthetic human training model. A limitation of this approach is that silhouette data will be projected onto the subspace spanned by the training pose, hence restricting the output to within this subspace.

Lee et al [Lee 2007] proposed an approach to model the visual manifold of an articulated object observed from different view points. The model introduced here is

generative. However, it generates observations for a certain motion as observed from different view points without any explicit 3D body model. Rather, this is achieved through modelling the visual manifold corresponding to different postures and views. An embedding of the kinematics and the motion manifold invariant to the view are acquired by using joint angles data. Then a parameterisation of the motion manifold in the embedding space and the dynamics are obtained through learning a flow field. Given view-based observations, view-based nonlinear mapping functions from the kinematic manifold embedding space to the observations in each of the views are constructed. The view factor can then be factorised using high order singular value decomposition [Lathauwer et al. 2000] according to the coefficients of view based functions, arranged as a tensor. With the view factor, the view manifold can be explicitly modelled in coefficient space, which leads to a representation of the view manifold invariant to pose configuration. Meanwhile individuals' shape variabilities are factorised within the same model. This results in two low-dimensional embeddings for the pose configuration and the view, as well as a generative model that can generate observations given the two manifolds' parameterisations. This fits perfectly into the Bayesian tracking framework as it provides: 1) a low dimensional state representation for each of the view and pose configurations, 2) a constrained dynamic model since the manifolds are modelled explicitly, and 3) an observation model, which comes directly from the generative model used.

Elgammal et al [Elgammal and Lee 2009] considered motion observed from a camera (stationary or moving), with motion representation as a kinematic sequence coming from a sequence of observations. With an accurate 3D body model, camera calibration, and geometric transformation information, they can explain  $Y$  as a projection of an articulated model. The dynamic kinematic sequence lies on a manifold called the kinematic manifold, and the observations lie on a manifold labelled the visual manifold. In fact, observations lie on a product of the body configuration and view manifolds. The relation between the kinematic manifold and the visual input mani-

---

fold can be regarded as a graphical model connecting the two manifolds through two latent variables: a body configuration variable and a view point variable. The body configuration variable is shared between both the kinematic manifold and the visual manifold. The view point variable represents the relative camera location to a human centred coordinate system. Another variable affecting the observation is the shape variability among different subjects, i.e. the human shape space, or the shape style. Their method is able to relate the kinematic manifold with the visual input manifold in order to infer configuration from input, while also modelling the visual manifold with all its variabilities due to motion, view point, and shape style. In particular, it can deal with both body configuration and view points as continuous variables. This facilitates tracking subjects with varying view points due to camera motion or changing subject view with respect to the camera.

This view variant human motion tracking is formulated as tracking on a torus surface. The torus is used as a state space for both pose configuration and view. The torus is deformed to the actual visual manifold and to the kinematic manifold through two nonlinear mapping functions. The torus model is suitable for one dimensional manifold motions, whether periodic, such as walking and running, or non periodic, for example golf swings or jumping. The experimental results showed that this model is superior to other representations for the task of tracking and pose/view recovery since it provides a low dimensional, continuous, uniformly spaced state representation.

## **2.3 Graph Based Image Segmentation and Tracking**

The performance of silhouette-based human motion capture is heavily affected by the results of image segmentation. When low colour contrast between the background and foreground is present, the background and foreground colour histograms are overlapped and it becomes hard to distinguish and classify them. As a conse-

quence, noticeable artifacts appear in the image segmentation. This often causes a multimodal landscape and corrupted global optimum in the observation likelihood function. Thus, many wrong poses may be considered equally likely to be the good pose. Ultimately, inaccurate image segmentation can lead to a high chance of mis-tracking or even complete failure of tracking. Fortunately, human motion capture has a strong temporal correlation, i.e., that activity is continuous over time. It is therefore wise to analyse human motion capture by accounting for temporal information. A natural research direction would utilise historical information to help with image segmentation. These kinds of methods would be beneficial at least from two standpoints. Firstly, the previous pose estimation suggests a reasonably good initial human body position and human body shape. This means current image segmentation should be temporally consistent with the previous pose estimation. If the sampling rate (the frame rate) is high, the temporal correlation is very strong and useful. Furthermore, more accurate image segmentation, and therefore improved pose estimation, results in better image segmentation for the next frame. These kinds of recursively temporal-dependent relationships have been addressed by Bray et al [Bray et al. 2006; Kohli et al. 2008]. Their method incorporates the historical pose as a prior in the dynamic graph cut [Kohli and Torr 2007] to improve image segmentation, and uses improved image segmentation to help the current pose estimation.

There are various ways to segment the image. The graph based approach has shown to be a powerful framework which provides a general formalisation based on graph theory, and a probabilistic construction based on Markov Random Field. This allows many discrete combinatorial optimisation schemes to be used for solving image segmentation.



### 2.3.1 Simultaneous Segmentation and Pose Estimation

Bray et al [Bray et al. 2006; Kohli et al. 2008] proposed a novel algorithm for performing integrated segmentation and 3D pose estimation of a human body from multiple views. The estimation and segmentation are formulated in a Bayesian framework building on the object-specific Markov Random Field (MRF) and provides an efficient method for its solution called PoseCut. A human pose-specific shape prior is incorporated by a stick figure in a pose-specific MRF for image segmentation, to obtain high quality segmentation results. Given the label configuration  $\mathbf{x}$  and pose configuration  $\Theta$ , image segmentation on the pose-specific MRF can be done by minimising an energy function:

$$\Psi(\mathbf{x}, \Theta) = \sum_i \phi(L_{bg}|x_i) + \phi(L_{fg}|x_i) + \phi(x_i|\Theta) + \sum_j \phi(\mathbf{D}|x_i, x_j) + \psi(x_i, x_j) \quad (2.3.1)$$

where,  $\phi(L_{bg}|x_i)$  and  $\phi(L_{fg}|x_i)$  denotes the unary term which imposes individual penalties for assigning the background and foreground label to pixel  $i$ .  $\phi(x_i|\Theta)$  denotes the shape-prior term given by:

$$\phi(x_i|\Theta) = -\log \left( \frac{1}{1 + \exp(\mu * (d(i, \Theta) - d_r))} \right)$$

where,  $d(i, \Theta)$  denotes the distance of a pixel  $i$  from the shape defined by  $\Theta$ . The parameter  $d_r$  decides how “fat” the shape should be, while parameter  $\mu$  determines the ratio of the magnitude of the penalty that points outside the shape have compared to the points inside the shape. The contrast term  $\phi(\mathbf{D}|x_i, x_j)$  of the energy function is defined as:

$$\phi(\mathbf{D}|x_i, x_j) = \begin{cases} \lambda \exp \left( \frac{-g^2(i, j)}{2\sigma^2} \right) \frac{1}{dist(i, j)} & \text{if } x_i \neq x_j \\ 0 & \text{otherwise} \end{cases}$$

where  $g^2(i, j)$  measures the difference in the RGB values and  $dist(i, j)$  gives the spatial

distance between pixels  $i$  and  $j$ .  $\psi(x_i, x_j)$  takes the form of a Generalised Potts model:

$$\psi(x_i, x_j) = \begin{cases} K_{ij} & \text{if } x_i \neq x_j \\ 0 & \text{otherwise} \end{cases}$$

Since the equation (2.3.1) satisfies the sub-modularity condition [Kolmogorov and Zabih 2002], it can be minimised by the fast dynamical graph cut algorithm. Furthermore, the pose estimation can then be formulated in nested optimisation with respect to the label configuration:

$$\Theta_{\text{opt}} = \arg \min_{\Theta} \left( \min_{\mathbf{x}} \left( \sum_{N_{\text{views}}} \Psi(\mathbf{x}, \Theta) \right) \right)$$

Their experiments demonstrate that this approach is able to simultaneously obtain excellent segmentation and pose estimation results.

### 2.3.2 Transductive Image Segmentation

In the context of image segmentation, some graph terminologies in graph theory are reinterpreted below. Given an  $m$  by  $n$  image, let a undirected graph  $G = (V, E)$  with  $N = m \times n$  nodes. Its node  $v_i$  denotes each pixel on the image and an edge  $e_{ij}$  denotes a connection between a pixel  $v_i$  and its neighbouring pixel  $v_j$ <sup>1</sup>. The  $N$  by  $N$  similarity matrix  $W$  is defined to describe the similarity between the pair of pixels. Particularly,  $w_{ij}$  is the similarity measurement between pixels  $v_i$  and  $v_j$ . The similarity matrix actually is a generalised adjacency matrix that describes the connectivity between nodes. The degree matrix  $D$  is an  $N$  by  $N$  diagonal matrix, whose the  $i$ th diagonal element  $d_i = \sum_j w_{ij}$ . The graph cut is a partition that separates the original graph into two disconnected subgraphs. The cost of the graph cut is equal to the summation of similarity values over cutting edges,  $\text{cut}(A, A') = \sum_{i \in A, j \in A'} w_{ij}$ . The Laplacian matrix is an  $N$  by

<sup>1</sup>Depending on the definition of the neighbourhood, the graph is varied. The 4-connected neighbourhood and 8-connected neighbourhood are commonly used.

$N$  symmetric matrix with one row and column for each node defined by  $L = D - W$ .

It holds many favourable properties of the graph listed below:

1.  $L$  is always positive semidefinite.
2. The number of times 0 appears as an eigenvalue in the Laplacian is the number of connected components in the graph.
3. The smallest eigenvalue is always 0.
4. The second smallest eigenvalue is called the algebraic connectivity.
5. The smallest non-trivial eigenvalue of  $L$  is called the spectral gap or Fiedler value.

Optimisation of image segmentation often involves integration over the entire feature space. This is usually analytically intractable. In their work [Duchenne et al. 2008], Duchenne et al point out that as the image segmentation problem deals with finite space, the integration over the entire feature space can be approximated by a discrete summation. Particularly, a laplace Beltrami operator is approximated by a Laplacian matrix. The transductive approach reduces an intractable integration problem to a discrete approximation, and eventually simple, solvable linear equations.

In their work, segmentation is treated as statistical transductive inference, in which some pixels are already classified correctly, and remaining ones need to be classified. The method utilises a laplacian graph regulariser, a powerful manifold learning tool based on the estimation of variants of the laplace Beltrami operator and tightly related to diffusion processes. The distinction between transductive and inductive inference is that there is not any unknown input, rather all inputs have known class labels. Thus, given a set of classified pixels, the task is to infer the class label of remaining pixels rather than infer the class label of novel pixels from different images. In this case, the generalisation process to avoid overfitting becomes less important, and a better fit decision boundary is more desirable.

In traditional optimisation of image segmentation, the objective is to search for a smooth function  $f$  from the input space into the output space such that  $f(x_i)$  is close to the associated output  $y_i$  (class label) on a training set. In Duchenne et al's work, it is assumed that the points are generated by a probability distribution<sup>2</sup>  $p$  with a support on a submanifold  $M$  of Euclidean space. Further, they believe the function value in low density regions (equivalently, the segmentation boundary regions) should be allowed to vary more than in other regions, since those are places where misclassifications occur. Hence, by imposing a control parameter  $s \geq 0$ , their approach can control how low the density should be to allow large variations of  $f$ . With consideration of the confidence  $c_i$  of the training pixel assignment, the inference problem can be summarised as follows:

$$\min_f \sum_{i \in \text{Train}} c_i (y_i - f(x_i))^2 + \int_M \|\nabla f\|^2 p^s dv$$

Minimisation to find a smooth function which infers the output label  $y_i$ , given  $x_i$ , minimises estimate errors while penalising overfitting, accounting for the density of the input probability distribution as well as a low density control parameter.  $c_i$  are positive coefficients measuring how much the training pair  $(x_i, y_i)$  will contribute to overall errors, and also reflecting the confidence that the class label  $y_i$  is correct. When  $p^s$  is small it allows a large magnitude of curvature to occur at a particular point, otherwise it encourages small changes at that point. However, the integral in the above formulation is mathematically intractable. An alternative from Hein et al's results [Hein et al. 2005] is an equivalent discrete approximation of this problem (more details can be found in [Duchenne et al. 2008]). It can be given by:

$$\min_{F \in \mathbb{R}^n} \sum_{i \in \text{Train}} c_i (y_i - F_i)^2 + F^T L_{un} F$$

---

<sup>2</sup> $p$  can be viewed as a density with respect to the Lebesgue measure of Euclidean space

where,  $F_i = f(x_i)$   $i = 1 \dots N \times N$  is an estimate label for  $x_i$ .  $L_{un}$  is a unnormalised Laplacian matrix. Further, it can be written as:

$$\min_{F \in \mathbb{R}^n} \sum_{i \in \text{Train}} (F - Y)^T C (F - Y) + F^T L_{un} F$$

where,  $C$  is an  $N$  by  $N$  diagonal matrix for which the  $i$ th diagonal element is  $c_i$  for a training pixel, and 0 for the remaining pixels. Similarly  $Y$  is an  $N$ -dimensional vector for which the  $i$ th element is  $y_i$  for a training pixel, and 0 for the remaining pixels. For the above quadratic minimisation problem, it is simply reduced to the solution of the following linear system by assigning the its gradient to zero:

$$(L_{un} + C)F = CY \quad (2.3.2)$$

By assuming  $F_i = y_i, c_i = \infty, i \in \text{Train}$ , segmentation is obtained by solving this simple linear system. Once again, the nature of the image segmentation problem is a linear combinatorial problem accounting for non-linear neighbourhood smoothness. Overall the transductive segmentation algorithm is outlined by Algorithm 3.

---

**Algorithm 3** Transductive Segmentation

---

1. Calculate the kernel  $k(x_i, x_j) = \exp\{-\frac{\|x_i - x_j\|^2}{2\sigma_g^2} - \frac{\|C(x_i) - C(x_j)\|^2}{2\sigma_c^2}\}$ , and degree  $d(x_i) = \sum_{j=1}^n k(x_j, x_i)$ , where  $\sigma_g$  and  $\sigma_c$  are scales for the geometric and chromatic neighbourhoods, respectively.  $C(x_i)$  denotes the RGB levels of a square patch of size  $2m + 1$  around the pixel  $x_i$ .
  2. Calculate the normalised kernel and degree by  $K(x_i, x_j) = \frac{k(x_i, x_j)}{(d(x_i)d(x_j))^\lambda}$ , where  $\lambda = 1 - s/2$ .  $D(x_i) = \sum_{j=1}^n K(x_j, x_i)$
  3. Compute the unnormalised Laplacian matrix  $L = D - W$
  4. Solve the linear system 2.3.2
  5. Threshold the output to  $1/2$ , if  $F_j \geq 1/2$ , then  $F_j = 1, j \in \text{Remain}$
-

### 2.3.3 Laplacian Matrix and Tracking

The Laplacian matrix corresponding to a finite sample of data points asymptotically approaches a continuous Laplacian operator. This is generally believed to be true, provided the sample size is increasing. On the other hand, the Laplace filter which uses difference to approximate the Laplacian is often used to detect large variations in an image, and it serves as an edge detector in image processing. Depending on the definition of the neighbourhood, both Laplacians capture the similarity or dissimilarity in neighbourhoods.

By definition the Laplacian matrix is equal to the degree matrix minus the similarity matrix  $L = D - W$ , and the summation of each row is equal to zero. This corresponds to a meaningful interpretation appearing in the image segmentation literature. If  $F$  denotes an estimated segmentation label, and  $L$  denotes the Laplacian matrix, then the regularisation term  $F^T L F$  means any segmentation label  $F$  should be consistent with the neighbouring smoothness which is encoded in Laplacian matrix  $L$ . Otherwise, it incurs the residual cost  $F_i d_{ii} F_i - \sum_{k \in i'sneigh} F_i w_{ik} F_k$ . Apparently, if all elements in  $F$  have the same label, the residual cost is minimised and equal to zero. However, this is not the desired image segmentation result. Hence, prior knowledge is added via an extra term, called the data term, to constrain the segmentation result. Thus image segmentation becomes a problem that uses prior knowledge data as a clue, while conforming to the naturally grouped regions in the image. Optimal segmentation is found when both the data and smooth term are minimised.

For human motion capture, the objective is slightly different from image segmentation. It requires a more accurate (usually more difficult) estimate of the human pose  $\theta$ . Therefore, it often incorporates stronger prior knowledge (such as foreground/background subtraction) to guarantee better tracking performance than image segmentation does. This actually involves strong human judgement as a prior, for example that the foreground pixel belongs to a tracking subject. This assumption is

---

simple for a human to make, but a computer cannot come up with it by itself. Thus, considering the neighbouring smoothness and foreground assumption, an appropriate objective function for human motion capture can be given by:

$$\min_{\theta \in R^n} \|F(\theta) - Y\|^2 + F(\theta)^T L F(\theta)$$

The equation means segmentation generated from the estimated pose  $\theta$  should obey the prior judgement  $Y$  (foreground subtraction) as well as conform to the natural image regions.





# Architecture Overview and Sequential Tracking Pipeline

---

This work adopts the generative approach rather than the discriminative approach, because with this approach it is possible to generate synthetic data points and approximate posterior distributions in temporal manner. This has better behaviour than recovering pose configurations from image observation space. Herein is presented a big picture of the entire architecture, and an introduction to the functionalities of the major components, including template modelling & automatic initialisation, observation likelihood evaluation, pose estimation and the sequential Bayesian tracking pipeline. The relationships and inter-operations between basic building blocks are explained in the sequential Bayesian filtering framework. Subsequent chapters will separately elaborate each component and address more technical details.

## 3.1 Architecture of Human Motion Capture

As with many other tracking systems, markerless motion capture can be regarded as a dynamic system, in which the current event has very strong temporal connections to preceding and successive events. The Contextualised Dynamical Architecture in Figure 3.1 visually captures the framework architecture in the temporal domain as well as functional components below:

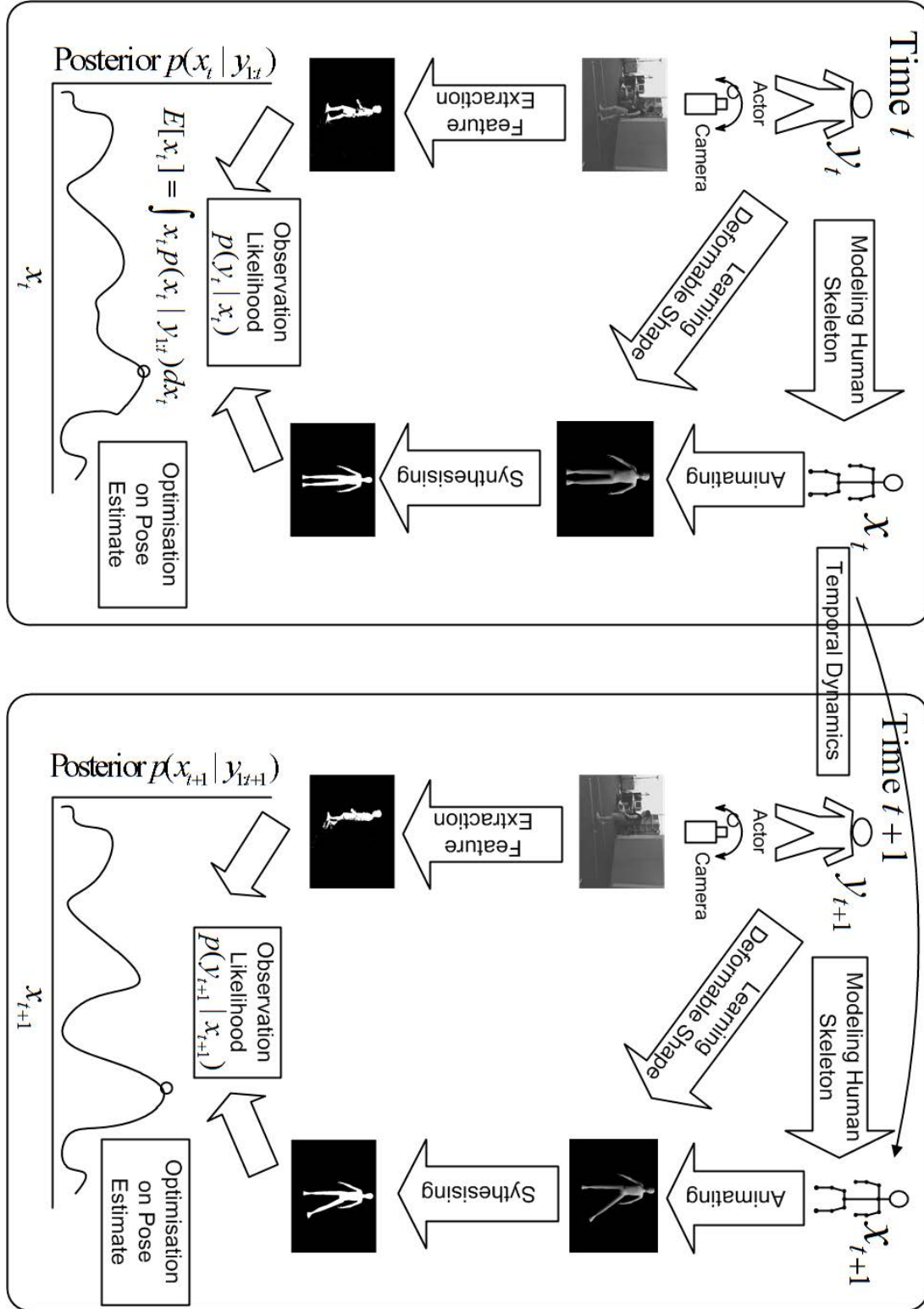


Figure 3.1: Contextualised Dynamical Architecture

- 
1. True posture and observation: There is an actor/subject performing. At any given instant time  $t$ , the actor has a posture  $\mathbf{x}_t^*$  (a vector including a position and joint angles), and an observation  $\mathbf{y}_t$  (e.g. multiview images) about the posture. The ultimate goal is to estimate the true posture  $\mathbf{x}_t^*$  for every instant
  2. Digital Acquisition: Actor's performance is captured by multiple distributed digital cameras, and stored as the sequence of digital images or videos.
  3. Skeleton Template Model: To be able to describe the posture of the subject, we adopt the standard articulated skeleton and kinematics routines in computer graphics. The posture estimate  $\mathbf{x}_t$  at time  $t$  is described by a template skeleton associated with a series of joint angles. Ideally, we hope to find the best match  $\mathbf{x}_t$  to  $\mathbf{y}_t$ . The details of how to build the generic skeleton is described in Chapter 4.
  4. Subject Specific Modelling: To improve tracking accuracy and robustness, a more advanced template body model is built according to the real subject appearance. This technique captures the gender, height, weight, shape and muscular tone appearance features and incorporates them into the pose deformable model. As a result, given a pose  $\mathbf{x}_t$ , we can render a virtual character in the corresponding posture. This can provide much richer information about the subject and helps reduce ambiguities in tracking. The technique details can be found in Chapter 4 and Section 6.2.
  5. Observation Likelihood: Directly observing or obtaining the true pose is not possible. The core of the framework is to find the best possible estimate  $\mathbf{x}_t$  for  $\mathbf{y}_t$ . In other words, we want to find the maximum observation likelihood  $p(\mathbf{y}_t|\mathbf{x}_t)$  in the sense of the Bayesian paradigm (more analytic details will be elaborated in following sections). The observation likelihood often takes the observations (information related to the true pose  $\mathbf{y}_t$ ) and a hypothesis estimate  $\mathbf{x}_t$  as inputs,

evaluates their similarity and outputs the similarity score. This usually requires that the observations and a hypothesis estimate essentially have a comparable form. The evaluation of the observation likelihood is a crucial part of marker-less motion capture, essentially related to the optimisation process, ultimately the tracking quality, and computational performance. We have proposed several novel strategies found in Chapter 6 and 7 to boost accuracy, robustness and performance.

6. **Feature Extraction:** Directly utilising digital images usually is not very effective. Feature extraction can be used to retrieve much more pose-relevant information, and remove irrelevant information and noise interference. For instance, the silhouette feature is often extracted and used in human tracking applications. Some techniques are introduced in Chapter 6 and 7 along with algorithms.
7. **Synthesis:** To make observation and hypothesis estimates comparable, a virtual character must be synthesised. A common approach is to perform perspective projection, using camera calibration parameters, to generate these images. This is described in Section A.1.
8. **Optimisation on Pose Estimation:** optimisation is performed to maximise the posterior probability. It uses the pose from the previous time as an initial position, iteratively evaluates the observation likelihood and ideally converges to the global optimum. The converged result is then regarded as the pose estimation for the current time  $t$ . However, because of the high dimensionality of skeleton parameterisation, ambiguities associated with the limited number of cameras and self-occlusions, this is a multimodal, high dimensional optimisation problem. As the conventional gradient-based method has difficulties in solving this problem, a stochastic approach is often used instead. In Chapter 5, we describe several nature-inspired algorithms to conquer this problem.

- 
9. Temporal Dynamics: If temporal transitional information is available, the current pose can be transformed to the successive time, with the same procedure repeated each time. Many studies have been focused on learning the temporal model for regression. Some examples are introduced in Section 2.2.

## 3.2 Sequential Bayesian Filtering Framework

Having a big picture of how pose estimation performs in the context of human motion capture, this section is intended to outline how state estimation is handled by the sequential sampling method. The objective is to estimate the state  $\mathbf{x}_t$  by calculating the expectation with respect to the posterior probability.

Intuitively, markerless motion capture is constructed on the basis of the dynamic system that explicitly characterises the causality (between the state and the observation) and the dependency (between the state and the prior state in the temporal domain), given a pair of hidden states and an observation corresponding to a certain point in time. Provided a state is independent of the other states, the first-order Hidden Markov Model [Baum et al. 1970] is sufficient to capture the sequential characteristics of states. The first order Hidden Markov Model assumes only a dependency between the current state and the previous state. All other states are ignored. Therefore, estimating the current state no longer requires storing all historical states. On the other hand, human motion can be considered as a sequence of states (human poses) and signals (associated observations) emitted from these states. The above framework can be reinterpreted as a dynamic system contextualised by human motion capture. At a certain point in time  $t$ , there is an observation  $\mathbf{y}_t$  that is the observable evidence of the human pose, and a hidden state  $\mathbf{x}_t$  that is an underlying true pose. The goal is to find the true state, given current and historical observations.

In reality, where dynamic systems are often analytically intractable, it is impossible to determine the exact value of the true state  $\mathbf{x}_t$ . Hence, an approximate estimate

$\hat{\mathbf{x}}_t$  is calculated instead. From the computer vision literature, a recursive Bayesian formulation [Doucet et al. 2000; Arulampalam et al. 2002], which recursively calculates the expectation of  $\mathbf{x}_t$  over the posterior  $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ , has been proven to be a reliable estimate and is widely employed. It starts with the previous posterior distribution, then maximises it by considering the product of the observation likelihood and the prior one in the sense of the Bayesian paradigm. The optimal estimate is found when the posterior probability is maximised.

In working with the posterior probability, one of common methods uses a sampling technique to approximate the posterior distribution. It is not practical to generate a large number of dense samples from the posterior distribution, but observation indicates that any sample has a similar probability to its nearby samples, so applying the variance reduction<sup>1</sup> to a moderate number of samples does not compromise the statistical significance of the original posterior distribution. With this idea, importance sampling [Denny 2001] can be used to approximate the posterior distribution by introducing a relatively small number of samples and associated importance weights  $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^N$ . The empirical estimate of the posterior probability can then be given by  $p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{j=1}^N w_t^j \delta_{\mathbf{x}_t^j}(\mathbf{x}_t)$ .

Figure 3.2 illustrates a procedure for estimating the state at time  $t$  and  $t + 1$ . Given a posterior distribution  $\{\mathbf{x}_{t-1}^i, w_{t-1}^i\}_{i=1}^N$  at time  $t - 1$ , samples are denoted by black round symbols whose positions correspond to the values of states and sizes reflect the values of weights. The first step is to redraw samples from the previous posterior distribution where samples with larger weights are more likely to be drawn. Consequently, samples with larger weights are more likely to be selected as new samples than others. These new samples are subsequently perturbed by Gaussian random noise in order to explore new positions. Weights are then updated by comparing new positions with observations. For instance,  $w_t = \exp\{-D(\mathbf{x}_t, \mathbf{y}_t)\}$  can be used

<sup>1</sup>That is equating the probability of the sample to the probability of its nearby samples and normalising over all possibilities, provided its nearby samples have approximately the same probability. In other words, overall the distribution has a smooth variation.

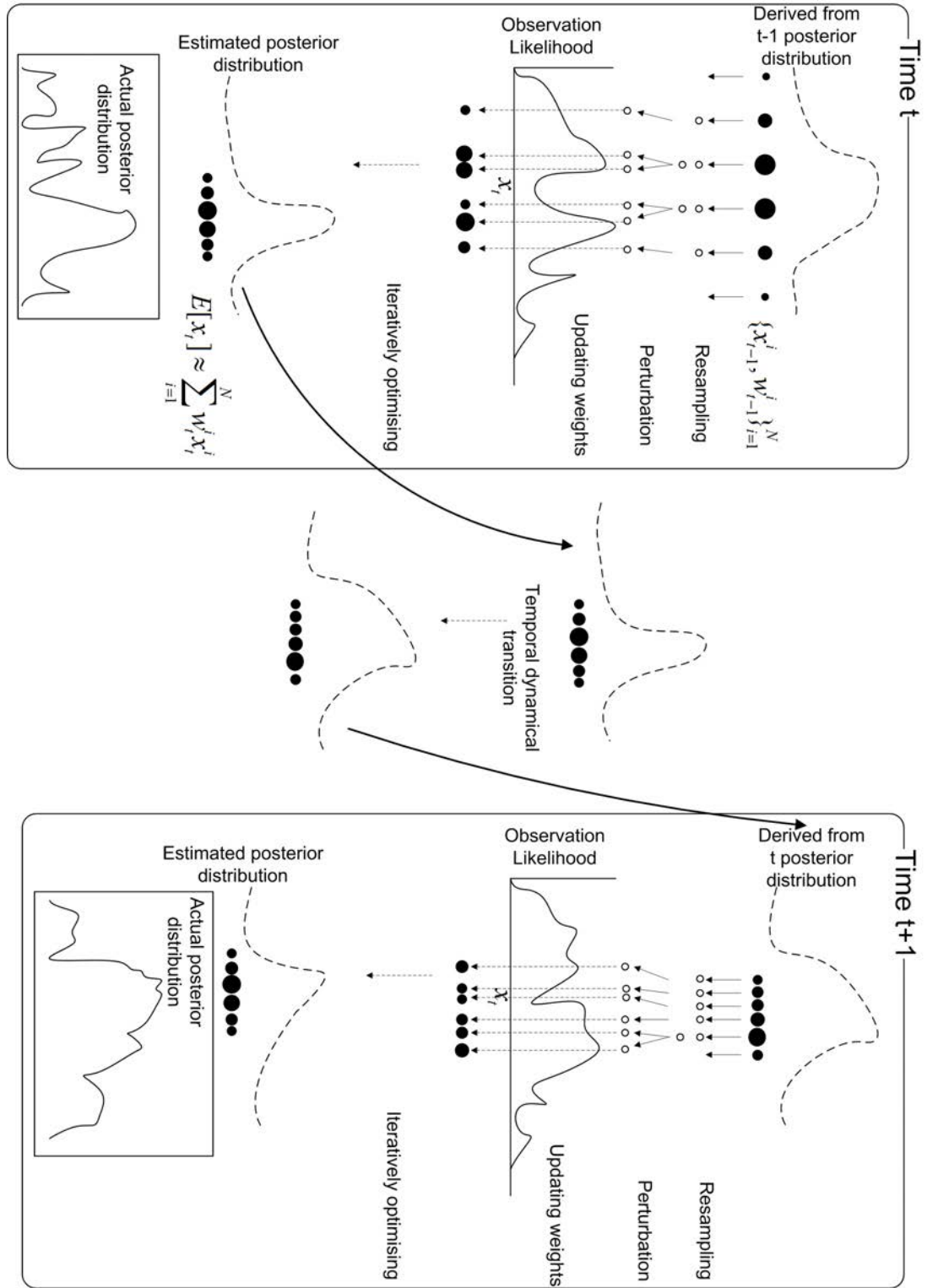


Figure 3.2: Sequential Sampling Framework

as an update equation, where  $D(\cdot)$  is a metric. The above procedure is repeated until samples converge on the optimal mode of the posterior distribution. Finally, the expectation can then be calculated from  $E[\mathbf{x}_t] \approx \sum_{i=1}^N w_t^i \mathbf{x}_t^i$ .

Through application of the temporal dynamical model, the posterior distribution at time  $t$  evolves to the posterior distribution at time  $t + 1$ . The above procedure is then repeated until samples converge on the optimal mode.

### 3.3 Particle Filter on Visual Tracking

Particle filter methods have become a very popular class of algorithms to solve these estimation problems numerically in an online manner, i.e. recursively as observations become available, and are now routinely used in fields as diverse as computer vision, econometrics, robotics and navigation. Point masses or particles, with corresponding weights, are used to form an approximation of a probability density function (PDF). The particles are propagated over time by Monte Carlo simulation to obtain new particles and weights (usually as new information is received), thus forming a series of PDF approximations over time. Particle filter methods allow the Bayesian estimation to be carried out in an approximate but structured manner. This is clearly useful in situations where some required posterior PDFs do not yield an analytical form. It is based on a sequential Monte Carlo method used for Recursive Bayesian Filtering.

#### 3.3.1 Recursive Bayesian Filtering

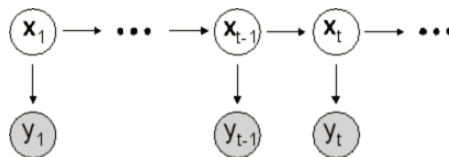


Figure 3.3: First-order Hidden Markov Model

Many real systems share a common underlying dynamic structure that consists of



a sequence of states at successive times, with each state associated with observations or measurements. In reality, the true states are usually unknown and hidden; only the observations are accessible. Therefore the challenge of inferring the true states and analysing the behaviour of the system, provided the a priori observations, appears substantial. The first-order Hidden Markov Model is often used to model these dynamic and sequential properties of systems. It can be represented in the graphical form illustrated in Figure 3.3. Given a set of sequential states  $\{\mathbf{x}_t; t \in \mathcal{N}\}$  and its associated observations  $\{\mathbf{y}_t; t \in \mathcal{N}\}$  (conforming in the sense of temporal propagation), the causal relationships of state  $\mathbf{x}_t$  at time  $t$  and its predecessors  $\mathbf{x}_{1:t-1}$  ( $\mathbf{x}_{1:t-1}$  denotes the set  $\{\mathbf{x}_i; i = 1..t-1\}$ ), and observations  $\mathbf{y}_{1:t}$  are characterised as the propagative conditional probabilities accounting for the uncertainty of dependencies. Substantially, the first-order Hidden Markov Model assumes the true state is conditionally independent of all states prior to the immediately previous state, and the observation is dependent solely upon the associated true state but is conditionally independent of all states other than the current state. These are mathematically formulated as:

$$p(\mathbf{x}_t | \mathbf{x}_{1:t-1}) = p(\mathbf{x}_t | \mathbf{x}_{t-1}) \quad (3.3.1)$$

$$p(\mathbf{y}_t | \mathbf{x}_{1:t}) = p(\mathbf{y}_t | \mathbf{x}_t) \quad (3.3.2)$$

With the Hidden Markov Model, the useful posterior  $p(\mathbf{x}_t | \mathbf{y}_{1:t})$  probability in the Bayesian framework can be reformulated in the context of the state space model. Let's go through the induction by starting with a joint probability equation:

$$\begin{aligned} p(\mathbf{x}_t, \mathbf{y}_{1:t}) &= p(\mathbf{x}_t, \mathbf{y}_{1:t}) \\ p(\mathbf{x}_t | \mathbf{y}_{1:t}) p(\mathbf{y}_{1:t}) &= p(\mathbf{y}_t | \mathbf{y}_{1:t-1}, \mathbf{x}_t) p(\mathbf{y}_{1:t-1}, \mathbf{x}_t) \\ p(\mathbf{x}_t | \mathbf{y}_{1:t}) &= \frac{p(\mathbf{y}_t | \mathbf{y}_{1:t-1}, \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1})}{p(\mathbf{y}_{1:t})} \\ p(\mathbf{x}_t | \mathbf{y}_{1:t}) &= \frac{p(\mathbf{y}_t | \mathbf{y}_{1:t-1}, \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1})} \end{aligned}$$

With Markov assumptions 3.3.1 and 3.3.2 the equation can be rewritten:

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1})} \quad (3.3.3)$$

By the Chapman Kolmogorov equation:

$$p(\mathbf{x}_t | \mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1}$$

Again using Markov assumptions:

$$p(\mathbf{x}_t | \mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1} \quad (3.3.4)$$

Combining 3.3.3 and 3.3.4, the recursive formula can be written as:

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t | \mathbf{x}_t) \int p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1}}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1})}$$

And by removing the normalising constant  $p(\mathbf{y}_t | \mathbf{y}_{1:t-1})$ :

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) \propto p(\mathbf{y}_t | \mathbf{x}_t) \int p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1} \quad (3.3.5)$$

This is called the recursive Bayesian filter formula. It actually states that the predictive posterior is dependent upon the likelihood-weighted expectation of temporal dynamics / transition priori  $p(\mathbf{x}_t | \mathbf{x}_{t-1})$  with respect to the previous posterior  $p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1})$ .

As the integral in the recursive Bayesian filter equation does not always have a closed form solution, researchers have investigated various sampling techniques to approximate the posterior calculation. The optimum way to explore search space is to allocate evaluations on samples in proportion to their observation likelihood related to the rest of the population. In this way, good candidates receive an exponentially increasing number of evaluations in successive resampling. Sequential Importance

Resampling is one of the Importance Sampling A.2 techniques which construct a set of samples with associated weights (called importance weights) to approximate a probability distribution. Over time, it automatically updates and adjusts the samples and weights according to current available observations so that the posterior distribution at any time can be approximated by the samples and their weights.

Overall, a generic Particle Filter algorithm is summarised in Algorithm 4 and visually depicted in Figure 3.4.

---

**Algorithm 4** A generic Particle Filter algorithm

---

```

for  $i = 1$  to  $N$  do draw  $\mathbf{x}_t^i$  from  $\pi(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i)$  end for
for  $i = 1$  to  $N$  do  $\bar{w}_t^i = w_{t-1}^i \frac{p(\mathbf{y}_t^i | \mathbf{x}_t^i) p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i)}{\pi(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i)}$  end for

for  $i = 1$  to  $N$  do compute the normalised importance weights  $w_t^i = \bar{w}_t^i / \sum_{j=1}^N \bar{w}_t^j$ 
end for
Compute an estimate of the effective number of samples as  $\widehat{N}_{eff} = 1 / \sum_{i=1}^N (w_t^i)^2$ 
if  $\widehat{N}_{eff} < N_{thr}$  then
  Draw  $N$  samples from the current sample set with probabilities proportional to
  their weights. Replace the current sample set with the new one.
  for  $i = 1$  to  $N$  do set  $w_t^i = 1/N$  end for
end if
Use the temporal model to predict the next state  $\mathbf{x}_{t+1}$  with Gaussian noise to simulate uncertainty

```

---

It is worth noting that step five of above Algorithm 4 is a resampling step. The purpose of it is to deal with the degeneracy problem of the common importance sampling algorithm. After a few iterations, all but one sample will have negligible normalised weight. This has almost zero contribution to the estimate of the posterior  $p(\mathbf{x}_t | \mathbf{y}_{1:t})$  but reduces the effectiveness of the algorithm. It is sensible to eliminate samples when their weights fall below a certain threshold. A measurement of the effectiveness of the estimate, called the number of effective samples  $N_{eff}$ , was proposed in [Liu, Jun S. and Chen, Rong 1995; Liu and Chen 1998]. It was defined as:

$$N_{eff} = \frac{N}{1 + \text{Var}(w_t^{*i})}$$

where  $w_t^{*i} = p(\mathbf{x}_t^i | \mathbf{y}_{1:t}) / \pi(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t)$ . Although  $N_{eff}$  can not be evaluated exactly, an estimate  $\widehat{N}_{eff}$  can be approximated by:

$$\widehat{N}_{eff} = \frac{1}{\sum_{i=1}^N (w_t^i)^2} \quad (3.3.6)$$

The number of effective samples indicates the degree of uniformness in spreading the probability mass amongst all the samples. For instance, the even distribution of probability mass has a large value of  $N_{eff}$ , whereas the peaked distribution of probability mass has a small value of  $N_{eff}$  which also indicates severe degeneracy. At the fifth step of above Algorithm 4, when  $N_{eff}$  is smaller than the threshold  $N_{thr}$ , the resampling procedure is performed in order to eliminate samples that have small weights and therefore concentrate on samples with large weights. Alternatively, selecting a better importance distribution  $\pi(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i)$  also can minimise the variance of  $w_t^{*i}$ , resulting in larger  $N_{eff}$ . Theoretically, when  $\pi(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i) = p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i)$ , the importance distribution is optimal with a minimum of variance. However, it is always difficult to be resolved.

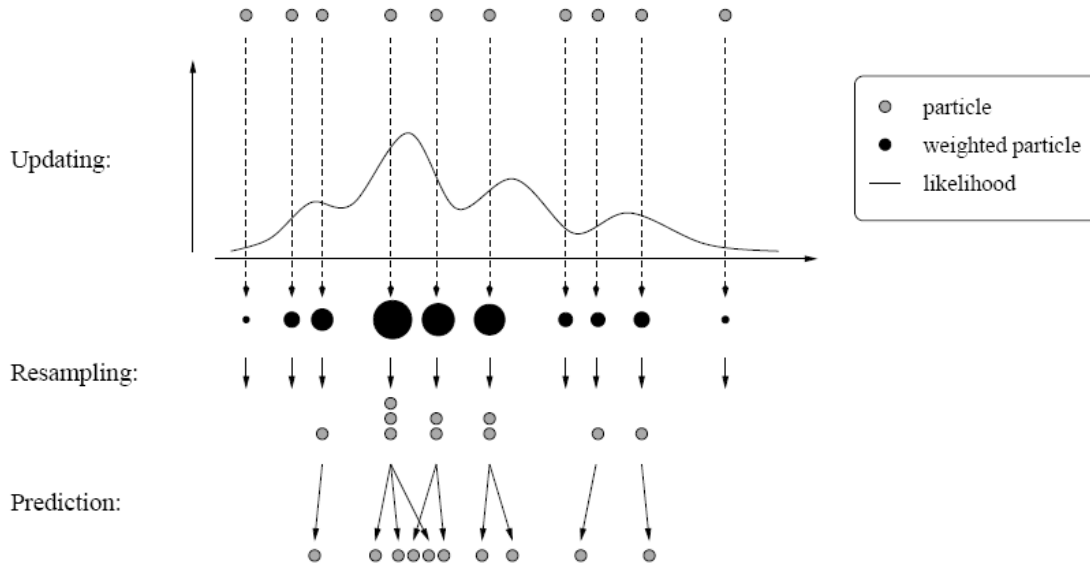


Figure 3.4: Particle Filter Algorithm (courtesy of [Gall et al. 2007])

---

# Subject Specific Body Shape Modelling and Automatic Initialisation

---

We start with a fundamental question: how much reliable information about the target is available at the upper bounds of tracking accuracy, and is it possible to track without knowing anything about the target? To some extent, the information acquired offline is a fundamental prerequisite for guaranteeing successful tracking. Considering tracking as an inference process, the goal is to infer the current state of a subject via available information which is often hidden in both observations and prior knowledge. How much effort should be put into extracting information from observations and prior knowledge is situationally dependent. Sometimes, it is easier to extract the current state directly from observation even without prior knowledge. On the other hand, if observations are entangled or information contained in prior knowledge is easier to acquire, it is wise to put more effort into prior knowledge. We believe human tracking belongs to the later case in which the relationship between the current posture and image observations is a complicated function in high dimensional space. Thus, the incorporation of more prior knowledge, such as building a subject specific appearance model for the tracking subject, is a sensible way to attack the problem.

The appearance model can be learned online and dynamically updated through the tracking process. This is an ideal scenario for many tracking problems. However, this scenario requires much more computational burden online, and tracking errors

are easily accumulated in the appearance model, leading to compromised tracking accuracy. On the other hand, subject specific modelling is a one-time effort that pre-builds a high resemblance model of the tracking subject (possibly in a controlled environment). The precision and details of subject specific modelling are guaranteed and not achievable by the online learning method. One-time modelling shifts the expensive online computations to the initialisation stage, as well as providing sound prior knowledge so that tracking becomes more robust and more accurate. This chapter begins with a review in some related works in human body modelling, and describes general articulated human body skeleton and its 3D joint rotation parameterisation in the form of Euler angles, the axis angle and the exponential map, as well as how to avoid rotation parameterisation singularities in optimisation. Two body shape parameterisations, needle based and data-driven based, will be described to generate the subject specific model. Finally, an automatic initialisation is proposed to simultaneously estimate the body shape and initial posture through a hierarchical optimisation method.

## 4.1 Related Works

Tremendous efforts have been made to study how to model the human body so that its digital version can be processed and utilised in different scenarios. In the computer graphics community, for example, researchers are concerned with the automatic generation and animation of realistic human models. The applications of a system that simultaneously models pose and body shape include crowd generation for movie or game projects, creation of custom avatars for visual assistants, and usability testing of virtual prototypes. The studies in this category [Allen et al. 2003; Magnenat-Thalmann and Seo 2004; Angelov et al. 2005a; Allen et al. 2006; Hasler et al. 2009] often use a general data driven approach to build a parameterised template model. In this, hundreds of high resolution 3D human body scans are fit with a template model mesh to

---

register correspondences. Subsequently, the parameterisation is derived for non-rigid surface deformation as a function of pose, body shape, or some combination, and solved by a nonlinear optimiser under correspondence constraints. The established parameterisation allows the template model to simulate a great variety of distinctive individual shapes and poses in realistic details.

To model body shape variation across different people, Allen et al. [Allen et al. 2003] morph a generic template shape into 250 3D scans of different humans in the same pose. The variability of human shape is captured by performing principal component analysis (PCA) over displacements of the template points. The model is used for hole-filling of scans and fitting a set of sparse markers for people captured in standard poses. A framework is introduced by Thalmann and Seo [Magenat-Thalmann and Seo 2004] for collecting and managing a set of range scan data to build a modeller that synthesises the realistic appearance of the body model directly from the control parameters. The developed tools are used to help annotate landmarks, automatically estimate skeletal structures for animation, and establish correspondences within the population of captured data. Their modeller then uses this structurally annotated data and synthesises new body models, by blending different models in a way that statistics are implicitly exploited. Consequently, their technique offers a time-saving generation of realistic, animatable body models with high realism, primarily for real-time applications. The SCAPE (Shape Completion and Animation of People) model [Anguelov et al. 2005a] represents both articulated and non-rigid deformation of the human body. The pose deformation model captures how the body shape of a person varies as a function of their pose and is parameterised by a set of 3D rotations of skeletal bones. The shape deformation model captures the variability in body shape across different people by shape parameters—a coefficient vector corresponding to a point in a low-dimensional shape space obtained by Principal Component Analysis. More recently, Allen et al. in [Allen et al. 2006] presented a method that learns skinning weights for corrective enveloping from 3D scans using a maximum a posteriori

estimation for solving a highly nonlinear function which simultaneously describes the pose, skinning weights, bone parameters, and vertex positions. The weight or height of a character can be changed during the animation and muscle deformation looks significantly more realistic than with linear blend skinning. Recently, a fully differential model describing the pose and shape has been presented by [Hasler et al. 2009]. The representation is uniform in pose and shape but involves solving two equation systems to reconstruct one mesh. Since pose and shape variations are expressed by a differential encoding invariant to rotation and translation, the main drawback of this approach is that the pose and shape cannot be analysed independently.

On the other hand, template based human tracking or articulated pose estimation in some relatively early studies of markerless motion capture focuses more on modelling prior information about the tracking subject, avoiding complicated and expensive geometric computations. Many studies [Sminchisescu and Triggs 2003; Balan and Black 2006; Kehl and Van Gool 2006; Sigal et al. 2004] use simple geometric primitives (e.g. cylinders and boxes) to approximate the shape of the human body. With the advancement of hardware computational capability, increased numbers of studies begin to utilise more complex and highly detailed template models to incorporate more prior knowledge. Vlastic et al. [Vlastic et al. 2008] address the capture of details in mesh animation from multi-view video recordings. Their approach performs fast pose tracking with minor manual interaction, providing meshes readily usable for editing operations including texturing, deformation transfer, and deformation model learning. In the work [Gall et al. 2009], Gall et al. recover the movement of the skeleton as well as non-rigid temporal deformation of the 3D mesh, providing a laser scanned 3D mesh available. Balan et al. [Balan et al. 2007] extended Anguelov et al.'s SCAPE [Anguelov et al. 2005a] deformation scheme to recover detailed human shape and pose from images. Their results show the SCAPE model is 10% better at explaining image foreground silhouettes than the cylindrical model and makes the likelihood function better behaved. Corazza and Mundermann et al.'s work [Corazza et al. 2010]



---

was also built on the development of the SCAPE model. This allows them to generate a subject specific model and employ a silhouette based and articulated ICP [Besl and McKay 1992] method to register body segments with a sequence of visual hulls. Meanwhile, high quality performance capture [de Aguiar et al. 2008] presented by de Aguiar et al. proposes a new mesh-based framework which does not only capture gestures of the subject, but also recovers small-scale shape details and handles complex types of apparel including ones that are very difficult to handle by marker-based techniques. A similar study [Starck and Hilton 2007a] from Starck and Hilton focuses on highly realistic surface capture that recovers the detailed surface shape with high fidelity texture. Their approach adapts a global optimisation process for smooth volume reconstruction and spherical parameterisations for texture remapping.

## 4.2 Generic Human Body Skeleton

To date, there are two primary standards to describe modelling of the human body in computer animation, H-Anim (Humanoid Animation) 1.1 standard [Human Animation Working Group ] and Body Animation MPEG4 standard [ISO/IEC Moving Picture Experts Group 2008]. Three major motion capture data formats C3D, BVH/BVA, ASF/AMC are used to store and retrieve the motion capture data. These two standards and three data formats have in common a way of defining a generic human skeleton structure. This is not surprising since any simple and compact representation of the human body should naturally fit to the anatomy of the human body. The proposed human body skeleton is also designed to conform to the H-Anim standard and the ASF/AMC format. This consideration enables natural integration of a skeleton-based animation scheme with the computer animation standard.

From simple stick figures, boxes, spheres and articulated cylinders to complex superquadric spheres, meshes and the more advanced deformable Laplacian mesh, various human body models has been used in markerless motion capture studies. As

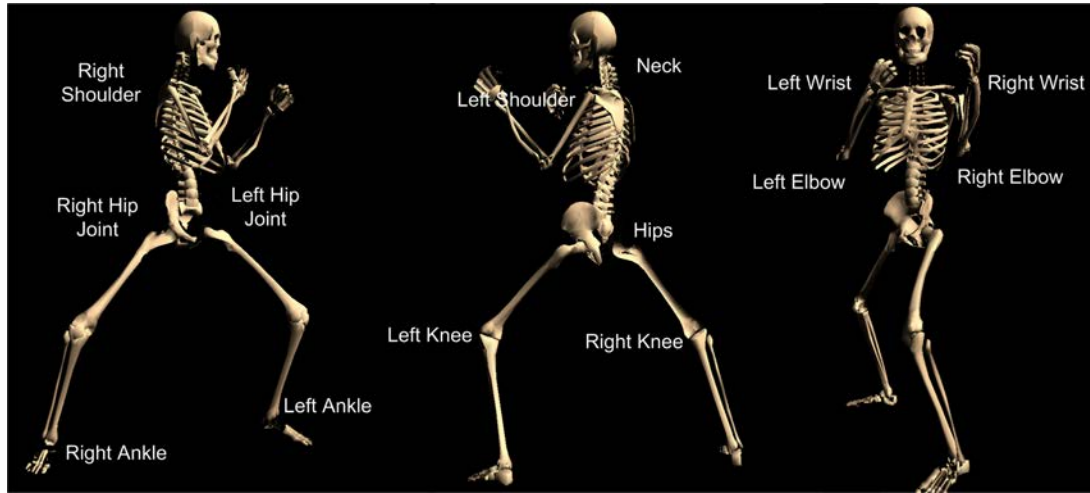


Figure 4.1: Human Skeleton and Joint Angles

a consequence of such diversity of body models, outcomes from different research groups become very difficult to analyse, evaluate and compare with each other. Although it is still possible to compare the outcomes from different studies, it is a time-consuming task to transform all output into a comparable format. Our approach emphasises on the establishment of a generic skeleton-based body model with a set of changeable parameters. Any specific body shape and posture can be adapted by adjusting the corresponding parameters without modifying the basic body skeleton structure. This attempts to provide a universal template in markerless motion capture, allowing easy comparisons. The proposed framework is constructed bearing in mind three considerations: I) the animation standard in computer graphics, II) the motion capture data standard, and III) existing skeleton models in markerless motion capture.

The human model used in this work is based on the skeleton illustrated in Figure 4.1, which has a total of 27 segments and 167 degrees of freedom (DOFs). To avoid too complex a representation, only 10 articulated segments (ankle and wrist joints are optional) and 25 DOFs are considered important and modelled for tracking. The translation and orientation of the entire model are described by 6 DOFs. The rest of the 19 DOFs are used to describe the joint angles of limbs. Thereby, any point  ${}^bP$  in

a local kinematic coordinate frame  $b$  can be transformed to the point  ${}^wP$  in the world coordinate frame  $w$  by:

$${}^wP = \prod_i^N T(\theta_i) {}^bP$$

where,  $N$  is the number of transformations.  $T(\theta_i)$  is a homogeneous transformation matrix specified by  $\theta_i$ , a particular set of joint angles and a translation. As illustrated in Figures (A.3, A.4, A.5, A.6, and A.7), human joint angles are limited in some movement ranges according to anatomical structure. Enforcing these movement limits not only guarantees the human model always presents sensible human poses, but also reduces the configuration space of human poses, which is beneficial to acceleration of the optimisation process.

The joint angle estimation involves the three DOF rotation parameterisation, which is non-Euclidean in nature. In fact, the three DOF rotation forms the Special Orthogonal Group  $SO(3)$  which is a naturally occurring example of a manifold. In terms of the manifold, the rotation parameterisation is a process of finding charts between  $SO(3)$  and  $R^3$  to explain the rotation. Since  $SO(3)$  is diffeomorphic to  $RP^3$  (real projective space), and charts on  $SO(3)$  try to model a manifold using  $R^3$  [Jacobson 2009], multiple-valued problems and singularities are inevitable. This explains why Euler angles have multiple combinations representing a given 3D rotation and suffer from gimbal lock, while the quaternion representation is always a double cover, with  $q$  and  $-q$  giving the same rotation. Different parameterisations have their advantages and disadvantages and using a particular parameterisation often depends on its performance in an application of interest. Appendix A.4 describes the three basic rotation parameterisations and a method that preserves smooth search space on Axis Angle and maintains the single 3D rotation parameterisation as 3 DOF rather than 4 DOF.

### 4.3 Human body shape

A skeleton alone may not be adequate to describe all the details of the human body shape. In fact, a more human-like model not only improves the appearance of the model, it also elevates the performance of markerless motion capture. However, it is a common that markerless motion capture has to track people with different heights and body shapes. The human-like model usually has to be built for a particular person. If a particular model is designed into the implementation level, functional methods often have tight dependencies on this particular model, and the entire approach becomes dependent on the model and has difficulty generalising. In the computer graphics community, there already exists a skeletal animation scheme which can move the human-like model design to a high level interface, so that the dependencies between functional methods and the human-like model are decoupled. In this skeletal animation scheme, the human body is represented in two parts: a surface representation used to draw the human body (the skin), and a hierarchical set of segments (or bones) used only for animation (the skeleton). In the most common case of a polygonal mesh character, each segment in the skeleton is associated with a group of vertices. The movements of segments control the transformations of vertices and ultimately deform the skin. The skin does not explicitly depend on the skeleton, and different skin types can be attached to different skeletons. This can be done in most 3D graphics packages. The skeletal animation scheme is a natural framework for skeleton based markerless motion capture.

As a result of conforming to the humanoid skeleton standard, rather than using geometry primitives to approximate the skin, the skin can be imported from any 3D object format. The 3D skin mesh can then be associated with the hierarchical skeleton by assigning a group of vertices to each bone. This is sometimes referred to as rigging. Each vertex in the mesh is associated and controlled by multiple bones with scaling

factors called vertex weights<sup>1</sup>. As a result, portions of the skin can be deformed to account for transformations of multiple bones. Instead of animating each vertex individually, the skeleton is manipulated, and the skin is deformed automatically. In the example illustrated in Figure 4.2, vertices are assigned to the bones according to geometric distances. As the child bone is rotated, its associated vertices are transformed with vertex weight scaling. Therefore, the vertices which are far from the parent bone are transformed further. Conversely, the vertices close to the parent bone remain close to their previous position.

This is formally stated in the Skeletal Subspace Deformation (SSD) algorithm [Magnenat-Thalmann et al. 1988] which is based on the weighted blending of an affine transformation of each joint by:

$$v_d = \left( \sum_{i=1}^M w_i T(\theta_i) \right) v_0$$

where,  $M$  is the number of joints,  $v_d$  is a vertex after deformation,  $w_i$  is a vertex weight and  $v_0$  is a vertex in the registered initial pose. Although SSD suffers from the inherent limitations of linear blending [Lewis et al. 2000] (known as the “collapsing joints” and “twisting elbow” problem, where in general, the mesh deformed by SSD loses volume as the joint rotation increases), this simple algorithm still remains the most popular deformation scheme because of its computational efficiency.

#### 4.3.1 Needle based Body Shape Parameterisation

When silhouettes, image textures and other features are available, body shape estimation is a problem that integrates information from disparate sources to obtain a consistent solution. This is a process of information fusion which can be solved in many different ways; recent trends are optimisation by GraphCut and the deformable models technique. GraphCut constructs the fusion problem into a MRF framework and solves it by maximum-flow/minimum-cut on a weighted graph where an exact

<sup>1</sup>Vertex weights are often assigned by the computer graphics software.

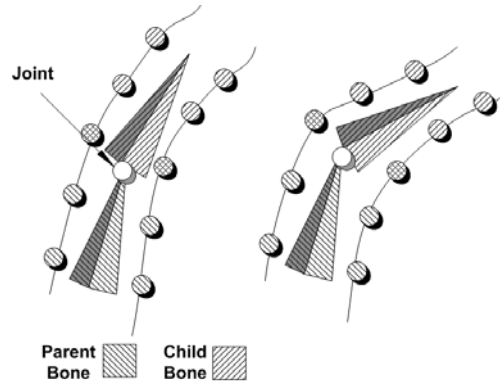


Figure 4.2: Vertex blending. The bones are drawn as triangular solids, vertices are drawn as circles. Vertices are shaded according to their associated bones. The movements of bones drive the vertices to be transformed in a manner scaled by vertex weights, ultimately leading to skin deformation.

global optimum is guaranteed to be found. For recent studies see [Tung et al. 2008; Starck et al. 2006]. The deformable models technique is derived from a very classic active contour model [Kass et al. 1988]. It builds on an energy minimisation framework, where the energy of the system is controlled by internal and external forces. The internal forces usually result in some constraint forces in the model itself (e.g smoothness constraints), and the external forces represent some constraints coming from the outside of the model, and attentional constraints, regularisation constraints, or high-level integrations. Ultimately, deformation is ceased when the energy of the system is minimised. This technique has also been extended to the model-based approach, which has a number of important advantages: (i) the model can be designed for rendering and manipulation as a standard computer graphics model; (ii) the model can be fit with a kinematic structure for animation; (iii) the model provides prior shape information to regularise multiple view reconstruction; and (iv) reconstruction then provides a consistent structure for all surfaces. Considering the model surface  $v$ , the initial surface  $v_0$  that may be defined according to a visual hull estimated by the shape-from-silhouette method, the internal energy  $E_{smooth}$ , and the external energies enforcing stereo consistency  $E_{stereo}$ , silhouette information energy  $E_{sil}$ , and user defined en-

ergies  $E_{user}$ , the objective is to minimise the total energy  $E_{total}$ :

$$E_{total}(v) = E_{smooth}(v) + E_{stereo}(v) + E_{sil}(v) + E_{user}(v)$$

equivalently:

$$\begin{aligned} \nabla E_{total}(v_{opt}) &= \nabla E_{smooth}(v_{opt}) + \nabla E_{stereo}(v_{opt}) + \nabla E_{sil}(v_{opt}) + \nabla E_{user}(v_{opt}) \\ &= F_{smooth}(v_{opt}) + F_{stereo}(v_{opt}) + F_{sil}(v_{opt}) + F_{user}(v_{opt}) = 0 \end{aligned}$$

By introducing a time variable  $t$ , the above equation can be solved via a differential equation:

$$v_t = F_{smooth}(v) + F_{stereo}(v) + F_{sil}(v) + F_{user}(v)$$

for the discrete case, it becomes:

$$v^{k+1} = v^k + \Delta t [F_{smooth}(v^k) + F_{stereo}(v^k) + F_{sil}(v^k) + F_{user}(v^k)]$$

Where  $k$  denotes the number of iterations, the key to solving the problem is to process  $E_{smooth}(v^k)$ ,  $E_{stereo}(v^k)$ ,  $E_{sil}(v^k)$  and  $E_{user}(v^k)$  with respect to the parameterised human body model.  $E_{smooth}$  is used to maintain the smoothness and the original shape of the model. It is usually defined in each local reference frame for each vertex so that the optimisation process also, to the some extent, conforms to the local geometric shape.  $E_{stereo}$  is defined as the error between a vertex of the model and the reconstructed location obtained by stereo matching between adjacent cameras.  $E_{sil}$  can be defined as the error between the model generated silhouette and the original silhouette produced by background subtraction. Alternatively, it can be defined as the error between a model vertex position and the closest surface element on the visual hull. The use of  $E_{user}$  is encouraged when the above energy terms are not able to provide adequate constraints for optimisation. The exact feature locations and correspondences are specified by the

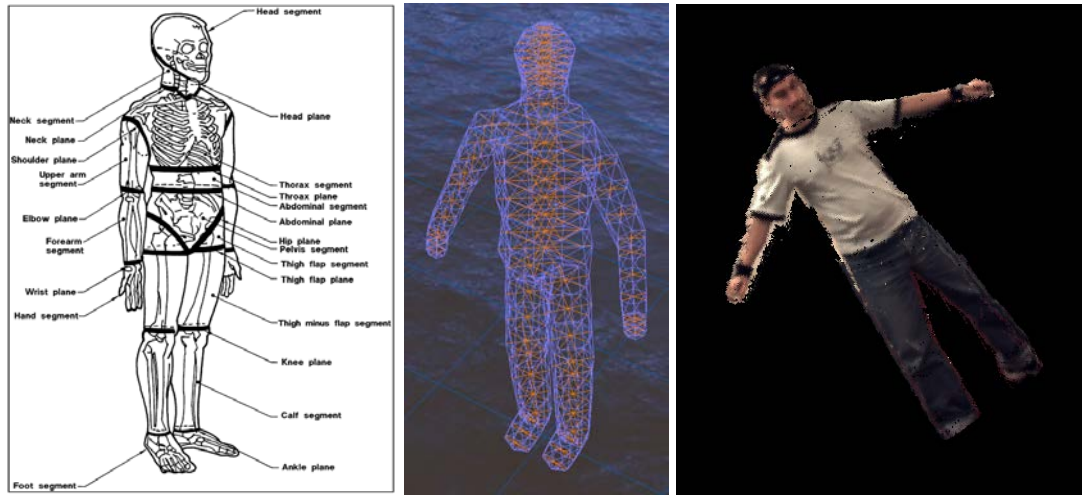


Figure 4.3: Body Segments, Body shape parameterisation and Visual Hull

user interactively. Then the error is defined between each vertex position and its correspondence.

The above framework and the study from [Sand et al. 2003] together inspire our needle based body shape parameterisation. Considering the skin mesh shown in Figure 4.3.1, a vertex  $i$  is regarded as a node. The shortest line from the vertex  $i$  to the bone axis is defined as the needle  $L_i$  of length  $l_i$ . To ensure that sufficient anthropometric details of the real human body shape are captured, a template human model consists of 10 segments, including the torso segment, head & neck segment, upper arm segment, forearm segment, hand segment, thigh segment, calf segment and foot segment, on both left and right sides. Since each segment has different dimensions, and variations within one segment are not necessarily uniform, the surface mesh is represented by discrete points whose resolution is controlled by the number of disks and slices. The needle lengths (shortest distance between vertices and their skeleton axis) are adjusted to capture the shape of each part. Overall, the needle-like representation (shown in Figure 4.3) well captures the variation in human body shape. Different configurations of needle lengths define different human body shapes. The endpoints of needles are triangulated to form the skin mesh which then can be deformed by SSD



[Magnenat-Thalmann et al. 1988] mentioned in Section 4.3.

Given observed multiview images and silhouette images at a particular instant, estimation of pose and body shape simultaneously is a very challenging problem. To simplify the problem, we assume the pose of the articulated skeleton is known. Then, the shape configuration parameters are the only variables that need to be estimated. The straightforward energy function can be formulated as a data term which evaluates the projection of a hypothesised shape configuration with the observed silhouette image from multiviews for each needle, plus a regularisation term, which smooths the surface mesh in the local region:

$$E_{total} = \frac{1}{N_{view}} \sum_{i=1}^{N_{view}} \sum_{x,y} (I_{sil}^{i,x,y} - I_{proj}^{i,x,y}(L))^2 + \alpha \sum_{k,(i,j) \in L_{adjacent}} w_k (l_i - l_j)^2 \quad (4.3.1)$$

subject to:

$$C_{min} < L < C_{max}$$

$$L < C'L$$

where  $L = \{l_i, i = 1, 2, \dots, n\}$  represents needle lengths, and  $L_{adjacent}$  includes pairs of adjacent neighbour indices.  $\alpha$  is adjusted to weight importance of the local smoothness, and  $w_k$  is introduced to allowing the smoothness variation at different regions.  $I_{sil}^{i,x,y}$  is the pixel intensity of the  $i$ th silhouette image at location  $(x, y)$  and  $I_{proj}^{i,x,y}(L)$  is the pixel intensity of the silhouette image at the location  $(x, y)$  generated by  $i$ th view projection with a given needle length  $L$ .  $C_{min}$  and  $C_{max}$  are the vectors containing the minimum and maximum values allowed for needle lengths.  $C'$  is the sparse matrix which imposes constraints on the needle lengths; for instance, the diameter of the torso should be greater than the diameter of the arm. When the energy of the system is minimised, the optimal value of all needle lengths should correspond to the

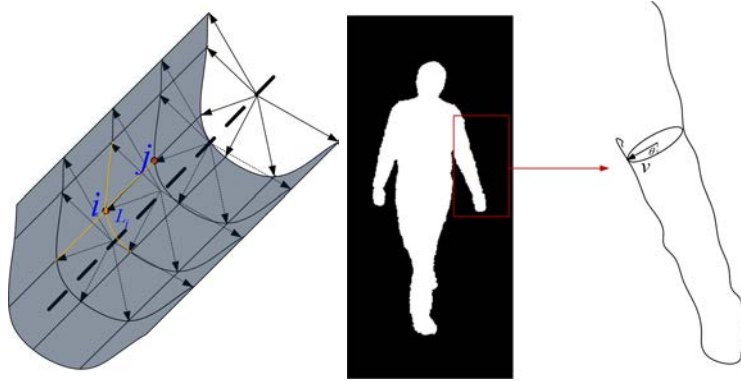


Figure 4.4: Left diagram: The needle based skin mesh. Right diagram: The silhouette contour consists of endpoints of needles which are orthogonal to the visual ray for a particular perspective

desirable body shape configuration.

The first term of the equation (4.3.1) is discrete rather than continuous, and very possibly has competitive local optimums. Moreover, the dimension of the shape configuration is equal to the number of needles, which could be more than one hundred. Consequently, optimisation of the equation (4.3.1) is often not well conditioned and difficult to solve. To handle this, we assume that needles within the same disk have uniform length, or equivalently that the edge of the disk is a circle with a given radius. The energy of each disk becomes the minimum distance, among all views, between the endpoint of the projected needle and its corresponding point on the contour of the silhouette. We also assume that it is always possible to find a corresponding point which is an intersecting point between the contour of the silhouette and the prolonged projection of the needle. This corresponding point is relatively stable and can be approximated by a fixed point on the contour of the silhouette. These assumptions are beneficial for reformulating the problem in the sense of least squares, converting the discrete image pixel-wise evaluation to continuous distance errors, and solving efficiently. The continuous energy function can be formulated by replacing the first term of the equation 4.3.1 with floating precision distance measurements (between

projected needle endpoints and the ground truth silhouette contour).

$$E_{total} = \sum_{j=1}^{N_{disk}} \min_{i \in View} d_{huber}(P_{cor}^{i,j} - P_{proj}^i(l_j)) + \alpha \sum_{k, (i,j) \in L_{adjacent}} w_k (l_i - l_j)^2 \quad (4.3.2)$$

subject to:

$$\begin{aligned} C_{min} &< L < C_{max} \\ L &< C'L \end{aligned}$$

where,  $P_{proj}^i(l_j)$  denotes the floating 2D position of projecting the endpoint of the needle  $l_j$  in the  $i$ th camera view, and  $P_{cor}^{i,j}$  denotes the correspondence of  $l_j$ 's endpoint on the silhouette contour in the  $i$ th camera view.  $d_{huber}(\delta)$  is the Huber distance, robust to outliers. It is defined by:

$$d_{huber}(\delta) = \begin{cases} \delta^2 & for \|\delta\| < b \\ 2b\|\delta\| - b^2 & for \|\delta\| \geq b \end{cases} \quad (4.3.3)$$

where  $b$  is the outlier threshold.

When the pose does not change and cameras are static,  $P_{cor}^{i,j}$  can be easily derived by intersecting the prolonged projection of the needle with the contour (i.e., we use Bresenham's line algorithm [Bresenham 1965] to search for the intersection point.), provided  $P_{proj}^i(l_j)$  can be determined. The problem then becomes one of calculating  $P_{proj}^i(l_j)$ . In fact, it turns out  $P_{proj}^i(l_j)$  can be calculated in a similar way to the point on the silhouette contour [Cipolla and Giblin 2000]. As in Figure 4.3.1, any point on the silhouette contour can be obtained by finding the endpoint of the needle which is orthogonal to the visual ray; more details will be given in the next section 4.3.1.1 .

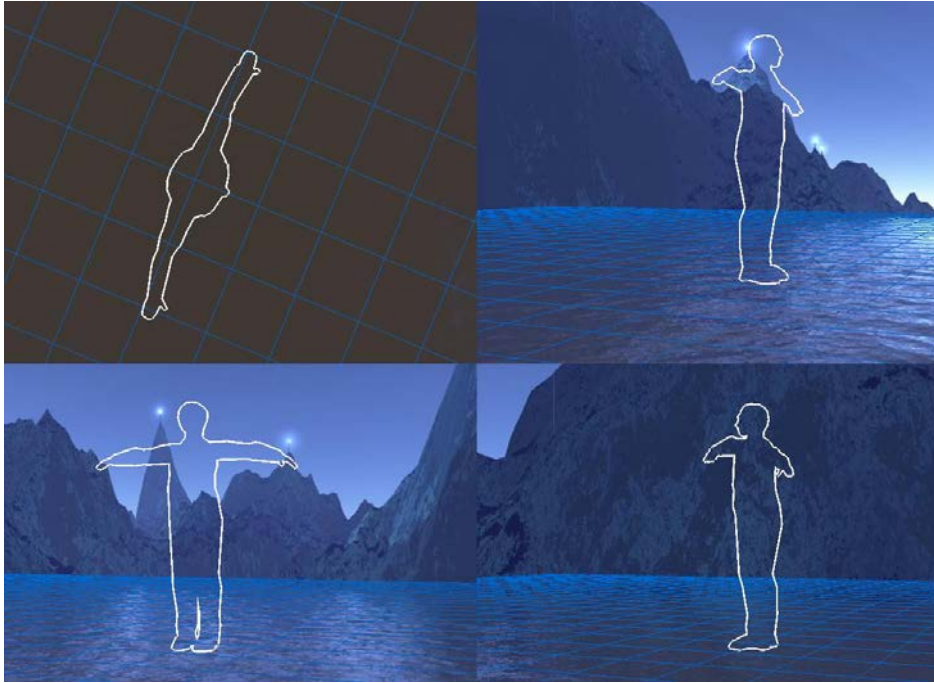


Figure 4.5: The silhouette contour (view dependent) of the human body often contains self occlusions

#### 4.3.1.1 Contour from Needle Projection

Given that there are an infinite number of needles on a complete disk, an immediate question is raised: which of those needles contributes to the actual silhouette contour? This relation has been analysed in the book [Cipolla and Giblin 2000] by Cipolla and Giblin. For a bounded smooth surface, the silhouette contour (called the *apparent contour*) is the projection of a particular space curve (called the *contour generator*) on the image plane. It turns out there is a simple and powerful property: at every point along the *contour generator*, the surface normal is orthogonal to the viewing ray. This is illustrated in Figure 4.6.

An ideal camera model with a cylinder to simulate the needle based limb is shown in Figure 4.7, where the left diagram shows the perspective view including a camera and a cylinder with a disk centred at  $B$ , and the right diagram shows the perspective view from the camera. The above property is interpreted as  $\overleftarrow{A(\theta)OC}^T \overleftarrow{A(\theta)B} = 0$ .

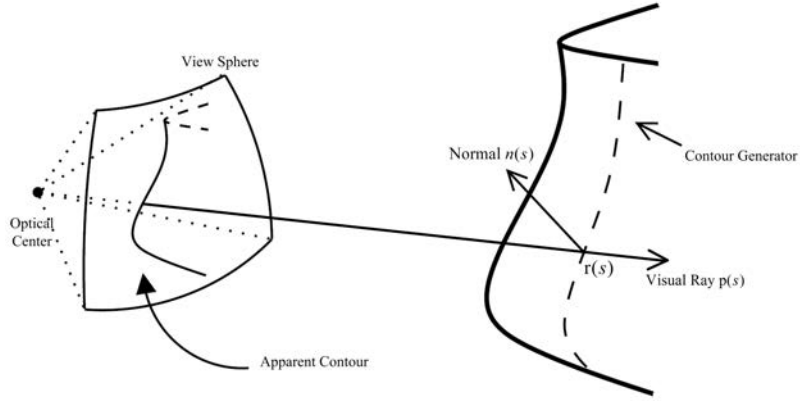


Figure 4.6: The visual ray is perpendicular to the apparent contour and the plane formed by the contour generator and the normal crosses through the corresponding point

However, there are some cases where none of the  $\overleftarrow{A(\theta)B}$  vectors are orthogonal to the visual ray vector  $\overleftarrow{A(\theta)OC}$ , in which case the apparent contour of the disk consists of the projection of all points on the disk circle. To check if there exists  $\theta$  such that  $\overleftarrow{A(\theta)OC}^T \overleftarrow{A(\theta)B} = 0$ , the optical centre  $OC$  is projected onto the plane of the disk centred at  $B$ . The Euclidean distance between the projection  $OC'$  and  $B$  is compared with the radius of the disk. If  $\|B - OC'\|_2$  is smaller than the radius,  $\theta$  does not exist. Otherwise, there are two possible cases. If  $\|B - OC'\|_2$  is equal to the radius, there is a unique  $\theta^*$  which is equal to the angle between  $BOC'$  and the  $y$  axis. If  $\|B - OC'\|_2$  is greater than the radius, there are two solutions  $\theta_1^*$  and  $\theta_2^*$ . One solution can be obtained by solving Equation 4.3.6 or minimising Equation 4.3.7 with the first derivative shown in Equation (4.3.8). Assuming without loss of generality that the first solution found is  $\theta_1^*$ , the second  $\theta_2^*$  can be gained by finding two intersection points between the disk circle and the circle which is centred at  $OC'$  with  $\|B - OC'\|_2$  as the radius.

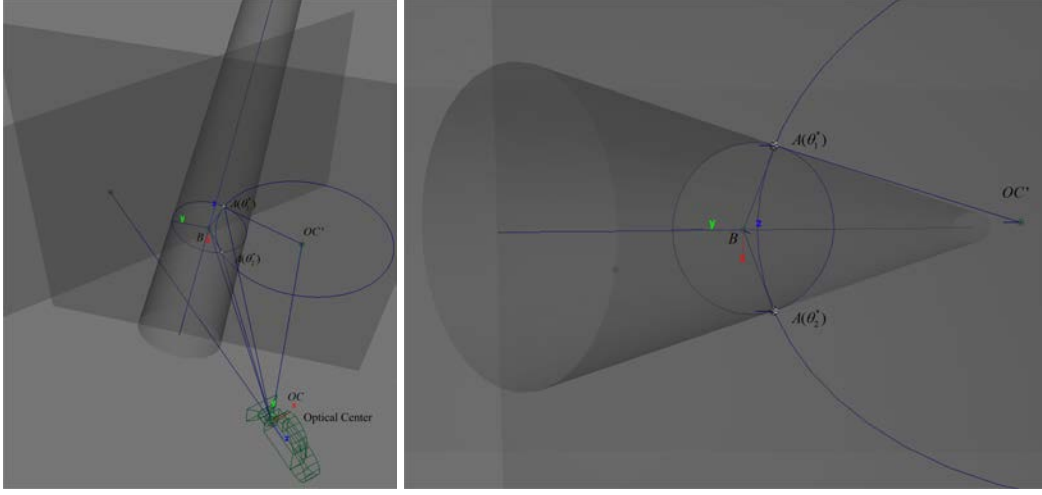


Figure 4.7: The left diagram illustrates the perspective view outside the camera. The right diagram illustrates the perspective view on the image plane. The disk centred at  $B$  may have two possible points  $A(\theta_1^*)$  and  $A(\theta_2^*)$  on the contour depending on the relationship between the distance  $\|B - OC'\|_2$  and the radius of the disk.

$$\overleftarrow{AB} = A - B \quad (4.3.4)$$

$$\overleftarrow{AOC} = A - OC \quad (4.3.5)$$

$$\begin{aligned} \overleftarrow{AB}^T \overleftarrow{AOC} = & (b_x + r \cos(\theta)u_x + r \sin(\theta)v_x - oc_x) (r \cos(\theta)u_x + r \sin(\theta)v_x) \\ & + (b_y + r \cos(\theta)u_y + r \sin(\theta)v_y - oc_y) (r \cos(\theta)u_y + r \sin(\theta)v_y) \\ & + (b_z + r \cos(\theta)u_z + r \sin(\theta)v_z - oc_z) (r \cos(\theta)u_z + r \sin(\theta)v_z) \end{aligned} \quad (4.3.6)$$

The solution of  $\overleftarrow{AB}^T \overleftarrow{AOC} = 0$  is equivalent to minimisation:

$$\arg \min_{\theta} (\overleftarrow{AB}^T \overleftarrow{AOC})^2, (0 \leq \theta < 2\pi) \quad (4.3.7)$$

The first derivative is given as follow:

$$\begin{aligned}
\frac{d(\overleftarrow{AB}^T \overleftarrow{AOC})^2}{d\theta} = & 2 \left( (b_x + r \cos(\theta) u_x + r \sin(\theta) v_x - oc_x) (r \cos(\theta) u_x + r \sin(\theta) v_x) \right. \\
& + (b_y + r \cos(\theta) u_y + r \sin(\theta) v_y - oc_y) (r \cos(\theta) u_y + r \sin(\theta) v_y) \\
& + (b_z + r \cos(\theta) u_z + r \sin(\theta) v_z - oc_z) (r \cos(\theta) u_z + r \sin(\theta) v_z) \\
& \times ((-r \sin(\theta) u_x + r \cos(\theta) v_x) (r \cos(\theta) u_x + r \sin(\theta) v_x) \\
& + (b_x + r \cos(\theta) u_x + r \sin(\theta) v_x - oc_x) (-r \sin(\theta) u_x + r \cos(\theta) v_x) \\
& + (-r \sin(\theta) u_y + r \cos(\theta) v_y) (r \cos(\theta) u_y + r \sin(\theta) v_y) \\
& + (b_y + r \cos(\theta) u_y + r \sin(\theta) v_y - oc_y) (-r \sin(\theta) u_y + r \cos(\theta) v_y) \\
& + (-r \sin(\theta) u_z + r \cos(\theta) v_z) (r \cos(\theta) u_z + r \sin(\theta) v_z) \\
& \left. + (b_z + r \cos(\theta) u_z + r \sin(\theta) v_z - oc_z) (-r \sin(\theta) u_z + r \cos(\theta) v_z) \right) \quad (4.3.8)
\end{aligned}$$

### 4.3.2 Data-Driven Body Shape Parameterisation

One popular data-driven method for characterising the variation between forms is the technique of Principal Component Analysis (PCA). In the field of computer graphics, PCA has been applied to sets of facial features [Blanz and Vetter 1999], human head models [Xi et al. 2007], human body models [Azouz et al. 2006] and more. PCA breaks down each example into a linear combination of orthonormal component vectors. These component vectors are arranged such that the first component explains as much of the variance in the examples as possible. The second vector explains as much of the remaining variance as possible after factoring out the first component, and so on. As a result, PCA serves two main purposes. First of all, it identifies correlated aspects of the examples within each component. Secondly, it gives us a way to reduce the amount of data we need to store. If we were to throw away the components above a certain threshold, we would still be able to reconstruct each example as a linear combination of just the low-numbered components.

The PCA based approach used in this section is similar to the method proposed in [Allen et al. 2003]. The major difference between our approach and [Allen et al. 2003] is that we use software to generate a variety of synthesised human humanoids as the dataset, rather than 3D body scan data. This allow us great flexibility in simulating variations in body shape and registering the mesh with the articulated skeleton. The open source software packages MakeHuman [MakeHuman 2010] and Blender [Blender 2010] are used to create the synthesised dataset. A script is used to generate 6561 models corresponding to different combinations of eight properties: gender, age, muscle tone, height, weight, chest circumference, waist circumference and hip circumference. Each model is also associated with a feature vector  $\mathbf{f}$  that defines height, weight, muscle tone, gender, body shape and a flag component. All models have an identical number of vertices  $n_v$ , and the same posture. For each model, we put all of the vertex positions into a single shape vector  $\mathbf{a}_i$ , which has  $3n_v$  elements in a single column. The index  $i$  ranges from 1 to  $N$ , with  $N$  equal to 6561 in this case. To begin, we calculate the mean  $\bar{\mathbf{a}}$  of the example vectors,  $\bar{\mathbf{a}} = \frac{1}{N} \sum_{i=1}^N \mathbf{a}_i$ . Next, we define a matrix  $\mathbf{A}$  whose  $i$ th column is equal to  $\mathbf{a}_i - \bar{\mathbf{a}}$ . To obtain the principal components, we multiply  $\mathbf{A}$  by the eigenvectors of  $\mathbf{A}^T \mathbf{A}$ . Associated with each principal component is a variance  $\sigma_i^2$ , which is equal to the corresponding eigenvalue. We sort the eigenvectors in order of decreasing  $\sigma_i^2$ . Because we have subtracted the mean, there are at most  $N - 1$  components with variance greater than zero. It turns out that the overall shape of the body can be captured reasonably well with as few as 25 components. Therefore, we use only the 25 most significant components to represent the template model.

Principal component analysis is able to characterise the space of human body variation, but it does not provide a direct way to explore the range of bodies with intuitive controls, which can be easily perceived by a human being (e.g. weight and height). Blanz and Vetter [Blanz and Vetter 1999] devised such controls for single variables using linear regression. Below, several variables are mapped simultaneously by learning linear regression between the controls and the PCA weights. If we have  $l$  such con-



trols, the mapping can be represented as an  $(N - 1) \times (l + 1)$  matrix (where  $l$  is 6 in this case) denoted by  $\mathbf{M}$ :

$$\mathbf{M}\mathbf{f} = \mathbf{p}$$

where  $f_i$  are the feature values of an individual, and  $\mathbf{p}$  are the corresponding PCA weights. The last component 1 of the feature vector  $\mathbf{f}$  enables  $\mathbf{M}$  to include an offset; without this parameter, setting the feature values to 0 would always result in a zero  $\mathbf{p}$ . Assembling all feature vectors for all models into an  $(l + 1) \times N$  feature matrix  $\mathbf{F}$ , we solve for  $\mathbf{M}$  as:

$$\mathbf{M} = \mathbf{P}\mathbf{F}^\dagger$$

where  $\mathbf{F}^\dagger$  is the pseudoinverse of  $\mathbf{F}$ , and  $\mathbf{P}$  is a matrix containing all of the PCA reconstruction weights as a column for each individual. We can then create a new feature vector, with a desired height and weight, and create an average-looking individual with those characteristics. In this way, the user can edit features independently, or together. So far, we have shown how to synthesise generic models according to feature values, but in addition, we can edit existing models by creating delta-feature vectors of the form:  $\Delta = [\Delta f_1, \dots, \Delta f_l, 0]$  where each  $\Delta f_i$  is the difference between a target feature value and the actual feature value for an individual. By adding  $\Delta\mathbf{p} = \mathbf{M}\Delta\mathbf{f}$  to the PCA weights of that individual, we can edit their features, e.g., make them gain or lose weight, and/or become taller or shorter. Some examples of body deformation are shown in Figures 4.8, 4.9, 4.10, 4.11 and 4.12.

#### 4.3.2.1 Dynamic Bone Length and Collision Bounding Box Adjustment

Another advantage of data-driven body shape parameterisation is that bone length and collision bounding boxes can be associated directly to the body shape parameterisation. For instance, when height is increased, the forearm bone length is also increased accordingly. When weight is increased, the breadth of the thigh is also increased and so is the dimension of the bounding box. This is simply done by calculat-

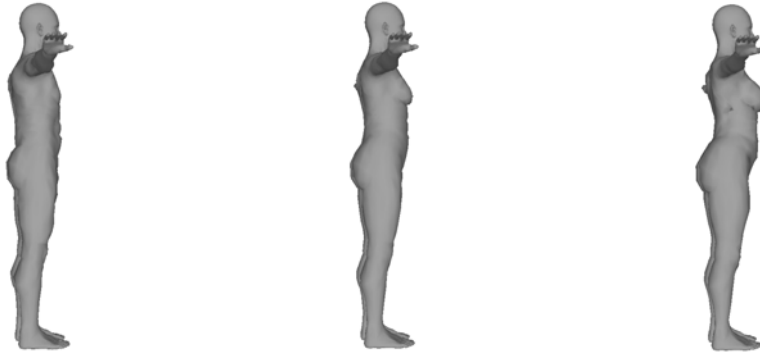


Figure 4.8: Gender Control: from left to right, male body shape is morphed to female body shape.



Figure 4.9: Height Control: from left to right, the height of the body is increased with other body parts changed accordingly.



Figure 4.10: Weight Control: from left to right, the body shape is changed corresponding to weight gain.



Figure 4.11: Shape Control: from left to right, the body shape is transformed from triangular to inverse triangular.



Figure 4.12: Muscle Tone Control, from left to right, the body parts are changed to reflect increasing muscle tone.

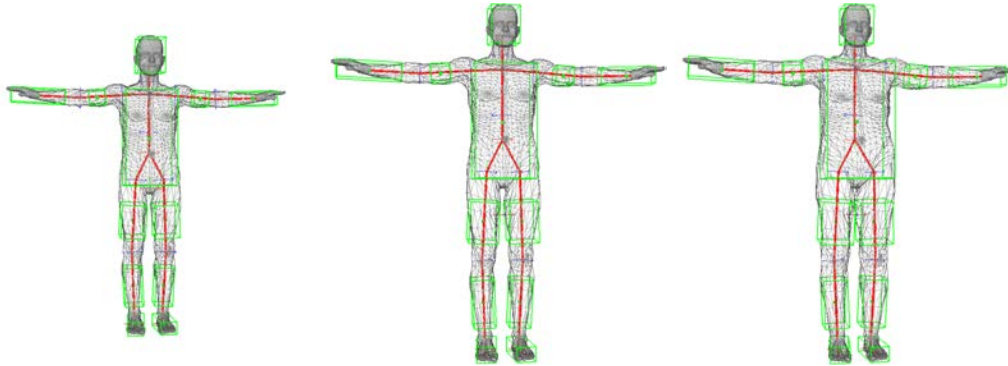


Figure 4.13: Bone lengths and collision bounding boxes are adjusted dynamically reflecting the underlying body morphing and body dimension changes.

ing a joint centre as the mean point of a predefined set of vertices and a bounding box of a set of vertices specifying a body segment.

## 4.4 Automatic Initialisation

Human centred applications like anthropometry, human factors design, ergonomics, virtual reality and performance measurement have played an increasingly important role in daily life. Many studies have shown successful achievements in parameterising human body shape and 3D modelling of the human body. Most of them require dedicated hardware and the instructive posture as premises. In markerless motion capture, the subject may change and clothing may vary frequently. Initialisation at the early stage thus becomes very difficult and requires a great amount of manual intervention. In this dynamic setting, automatically capturing appearance differences and pose becomes very desirable and necessary. Some works [Corazza et al. 2009; de Aguiar et al. 2005; Ahmed et al. 2005] demonstrate automatic subject modelling and pose estimation, which is quite similar to our work.

We employ a template human body model described in Section 4.3 whose shape and proportions can be customised. The articulated skeleton described in Section 4.2 comprises 10 segments, which provides 25 pose parameters in total. The surface ver-

---

tices are registered to the skeleton via the SSD technique described in Section 4.3. A silhouette-based analysis-by-synthesis approach is performed to search the optimal anthropomorphic shape and pose parameters in order to maximise overlap between the silhouette of the re-projected model and the image silhouette in all camera views. The energy function that numerically assesses this overlap sums up the number of pixels in binary XOR between the image and model silhouettes from all camera perspectives. In the following sections, we describe and compare two types of body shape parameterisation models.

#### 4.4.1 Using Needle based Body Parameterisation

An experiment has been conducted on the HumanEval 7-view dataset Subject 3. The silhouettes are manually extracted from the images. The true pose of the skeleton is assumed to be known. Then, we employ Powell’s method to optimise Equation (4.3.1) and the Levenberg Marquardt method to optimise Equation (4.3.2). Both equations have about 150 DOFs. It turns out that the former approach allows for finding a solution ten times faster than the latter approach. The latter approach is more prone to local optimums, appearing as more non-overlapping areas in the bottom panel of Figure 4.15. Results are shown visually in the initial shape configuration at the top and the converged shape configuration at the bottom of Figures 4.14 and 4.15. Even though the human body may be deformed dramatically in various gestures, the local shape of the human body surface still preserves certain geometric structures. However both of the approaches above have a problem with capturing these geometric structures. The data-driven based approach introduced in the next section is superior to the above approaches, and is able to well capture the underlying invariant geometric structures of the human body.

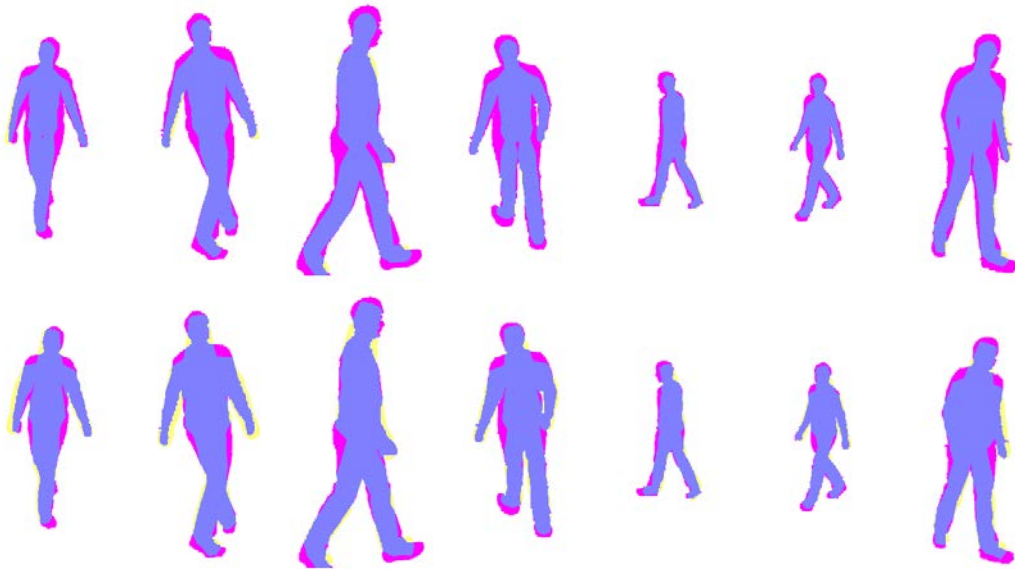


Figure 4.14: Experimental results on the HumanEval dataset with 7 perspective views using discrete evaluation with respect to the bit mask silhouette. The top row shows the overlapping images given the initial pose. The bottom row shows the converged results.

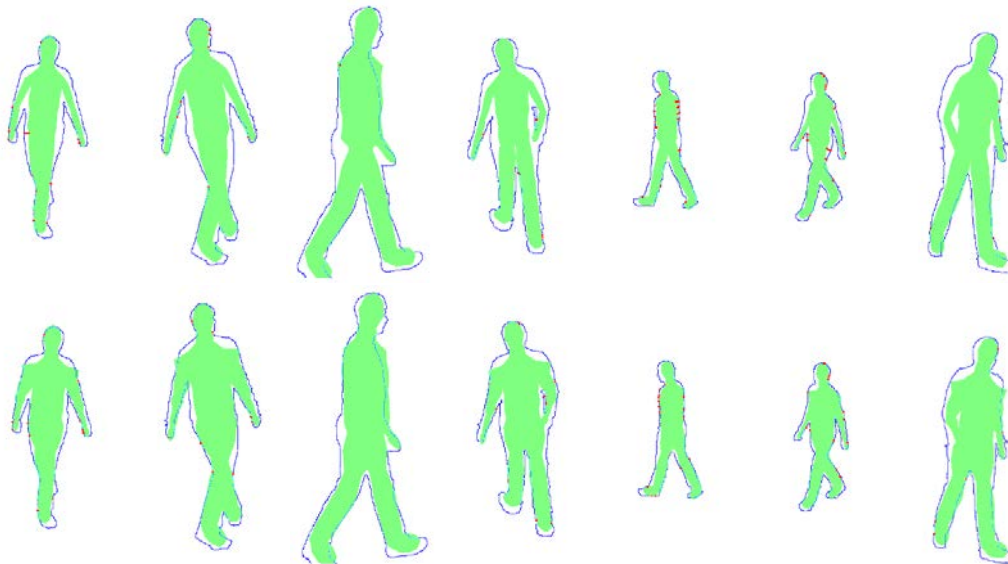


Figure 4.15: Experimental results on the HumanEval dataset with 7 perspective views using continuous evaluation with respect to the contour. The top row shows the template model with the initial pose projected to the contour images. The bottom row shows the converged results after applying the Levenberg Marquardt method. The primary non-overlapping region is caused by the detail differences between the real subject and the template model, since this needle based shape parameterisation method can not capture small-scaled invariant geometric structures of the human body

---

### 4.4.2 Using Data-Driven Shape Parameterisation

Given a pose and shape configuration, we can model a corresponding data-driven model (e.g. shown in Figure 4.16). The human pose space conforms to the hierarchical skeleton structure, with optimisation implicitly constrained to this hierarchical order. For instance, before the torso position is reasonably estimated, estimating the other pose parameters and shape parameters are meaningless efforts. To avoid the curse of dimensionality and utilise these hierarchical constraints, we perform CMA-Annealing optimisation (described in Section 5.4) to the articulated skeleton structure. The hierarchical procedure begins with estimating the torso position. The silhouettes are substituted with distance transformed silhouettes as in Figure 4.17 in order to make the energy function better behaved. Consequently, the descent direction more clearly points to the original silhouette, and optimisation has a lower chance of becoming trapped in local optimums. After the torso position has converged to a reasonable degree, the position and rotation of the torso are optimised simultaneously. The torso position is constrained to a smaller search range due to the fact that it is already close to the optimal position. Once the torso has been posed into a reasonable position and orientation, the optimisation procedure is applied to the body segments in hierarchical order. The body segments used for re-projection also obey the hierarchical structure as shown in Figure 4.18. The head, left/right upper arms, left/right thighs, left/right lower arms, and left/right calves can then be estimated in order. The whole body model should then already be posed reasonably close to the optimal posture. In the next stage, the shape configuration is estimated while the pose configuration is also adjusted but constrained to small search ranges. Finally, the shape and pose configurations of each limb are re-estimated within small search ranges to refine the converged result. The progress of convergence for the automatic initialisation are demonstrated in Figure 4.19. The initial and converged poses in the 3D environment are demonstrated in Figure 4.20.

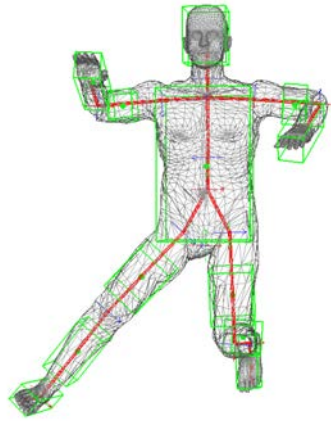


Figure 4.16: A typical customised model with a given pose and shape configuration

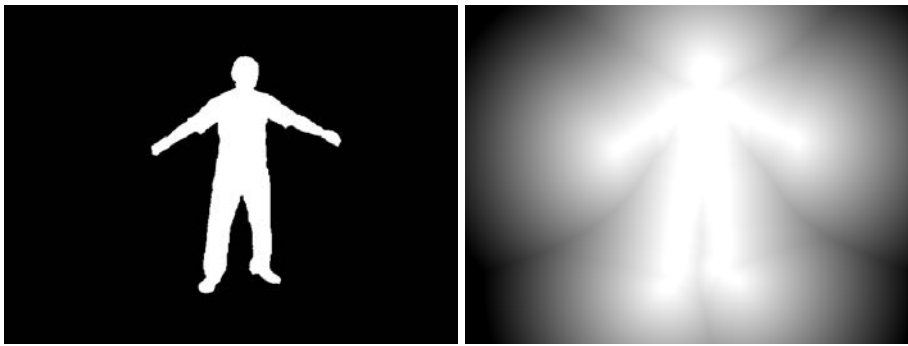


Figure 4.17: The left diagram is the silhouette. The right diagram is the distance transformed silhouette, which helps the energy function be well behaved, making errors progressively concentrate around the area of the silhouette.

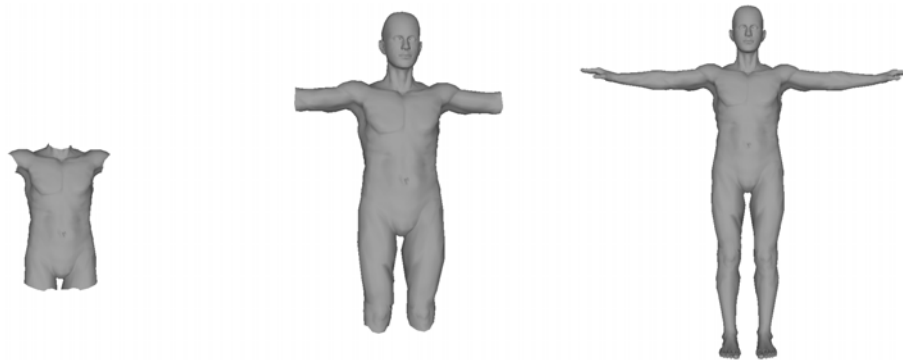


Figure 4.18: Hierarchical optimisation: Optimisation begins with estimating the translation and rotation of the torso; only the torso segment is projected onto the image. Subsequently the head, left/right upper arms, and left/right thighs are estimated one by one. Finally the left/right lower arms and left/right calves are estimated one by one. This approach partitions the original search space into several lower dimensional subspaces so that we only need to solve easier optimisation problems.





Figure 4.19: The sequence of images from four camera views illustrates progressive convergence of results. The first column is the initial pose rendering. The second and third columns show the results after estimating the torso's translation and rotation respectively. The fourth column gives results after positioning the head, left/right upper arms, and left/right thighs. The fifth column shows estimates of height, weight, gender, shape and muscular tone. The sixth column is after estimating left/right lower arms, left/right calves and shape parameters. The last column shows the original images.

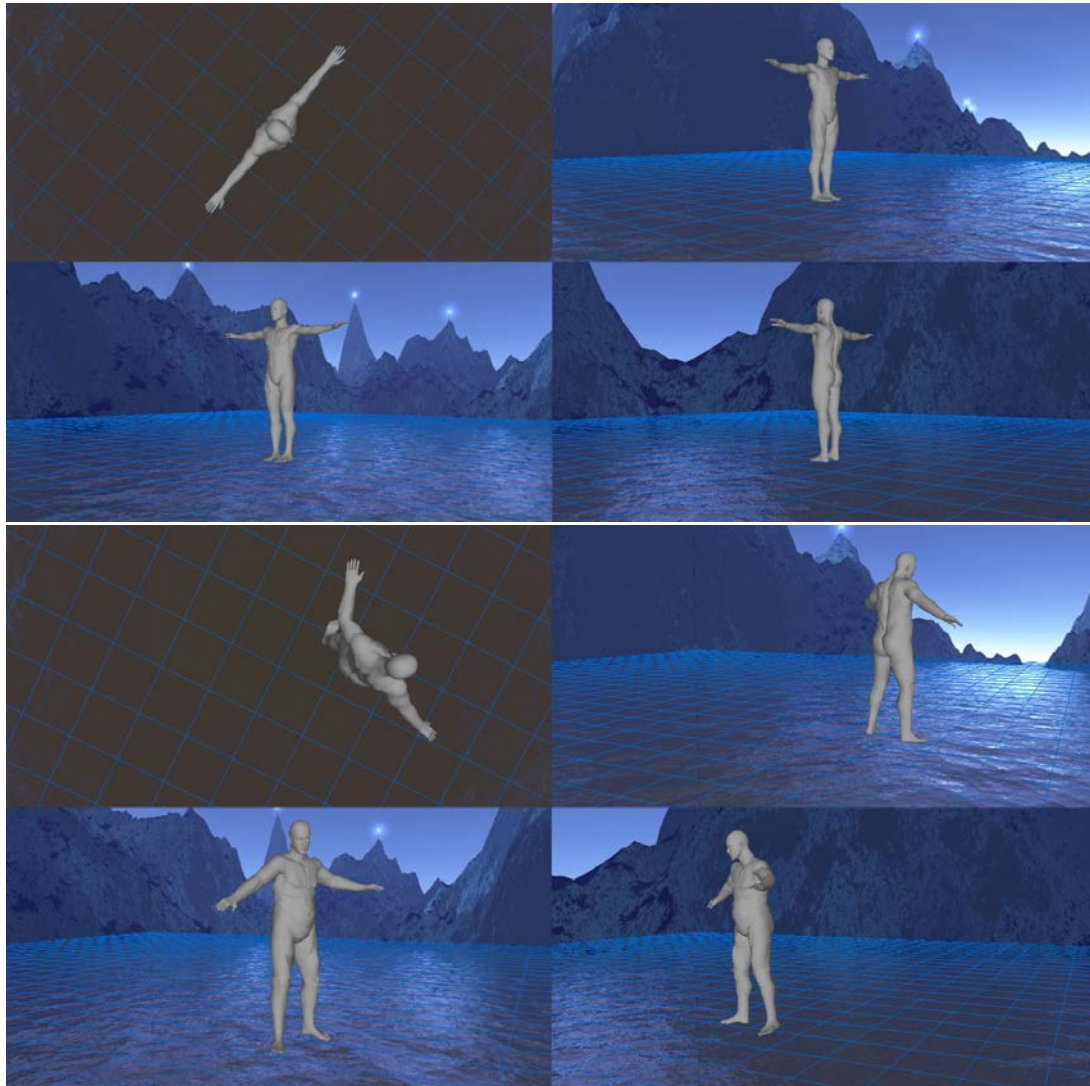


Figure 4.20: Top diagram: The initial body shape and “T” pose; Bottom diagram: the morphed body shape and estimated pose

# Nature Inspired Global Optimisation

---

Due to the diversity of nature, many practical problems reside in complex circumstances in which the problems are impacted by multiple factors. These competing factors can result in multiple solutions to practical problems – so-called multimodal problems. To find the “best” and “unique” solution in a multimodal problem requires global optimisation. This is a process of exploring and exploiting global information to find the optimal solution. Depending on the circumstances, global optimisation may appear straightforward for some problems, but it has been very challenging and even practically impossible for many others. For instance, if the problem has a convex property – local shape is consistent with global shape – this can be utilised to reveal global information, and global optimisation can be solved by relatively easy local optimisation. However, most examples of scientific and engineering problems are not convex, but multimodal, and gathering global information is almost impossible due to the complexity, which is multiplied by the number of factors. Fortunately, there are plenty of similar problems which have occurred and been resolved very well in nature. Notably, biological evolution has shed some light on this topic. Evolution itself is a process of moving towards some kind of optimised state, where biological species are favoured based on the how well they adapt to the environment. Variation (a somewhat random process) of successive generations is introduced by reproduc-

tion (e.g. trait inheritance, mutation, and genetic recombination). Progressively, the environment and ecology as whole are improved and move towards a more adapted biological state. This has inspired a great number of stochastic methodologies for solving many practical problems in real life.

In the context of the markerless motion capture multimodal problem, global information is often unknown beforehand and there is no efficient way<sup>1</sup> to explore and exploit properties of global information. This problem is often regarded as a black box, and solved by meta-heuristic stochastic optimisation. Stochastic optimisation not only shows excellent scalability<sup>2</sup>, but it also overcomes the limitations of the learning based approach which is only able to generalise activities similar to the training actions. Until better ways emerge to efficiently represent and exploit multimodal global information in markerless motion capture, stochastic optimisation has many advantages over other methods in terms of accuracy, robustness and generality to different activities. In this chapter, we review three stochastic optimisation algorithms – Simulated Annealing, Particle Swarm Optimisation and Covariance Matrix Adaptation Evolution Strategy – to solve global optimisation problems. Then we propose a novel hybrid method, Covariance Matrix Adaptation Annealing, to attack a specific class of problems. In this class of problems, our method is able to maintain high convergence speed as well as robustness to multimodality. Its characteristics and properties are validated through a series of benchmark multimodal functions. Its behaviour and performance are compared with other stochastic methods.

## 5.1 Simulated Annealing

Simulated Annealing has an analogy with thermodynamics, specifically with the way that metals, and some liquids, cool and crystallise. Initially, when the temperature is

---

<sup>1</sup>To the date, to the best of our knowledge, learning-based techniques still can not fully utilise global information in markerless motion capture

<sup>2</sup>Stochastic optimisation is able to solve a multimodal problem in high dimensional search space even on a personal computer.

---

very high, particles are free to move and explore a large area, and as the temperature decreases according to a cooling schedule, thermal mobility is gradually lost. Slowly lowering the temperature allows thermal equilibrium to be attained at each stage. Eventually, the particles tend to line themselves up in a stable crystalline structure, which often is the state with the minimum energy.

Simulated Annealing as proposed by Kirkpatrick et al [Kirkpatrick et al. 1983] opened a new perspective on combinatorial optimisation incorporating statistical mechanics. To quote: “There is a deep and useful connection between statistical mechanics and multivariate or combinatorial optimisation.” [Kirkpatrick et al. 1983].

Statistical mechanics is the discipline of analysing the relationships between microscopic properties of individual particles and the macroscopic or aggregate properties of the entire system that can be observed in daily life. A central principle of statistical mechanics is the Boltzmann theorem, which can be stated as follows: Consider an isolated<sup>3</sup> system that contains a large number of particles at temperature  $T$ . At any point in time, each particle is identified by its microstate (i.e., position and velocity). Furthermore, an aggregation of all particles’ microstates at a point in time identifies a unique macrostate  $s_i$  of the system. Each of these states is associated with an energy  $E(s_i)$ . It turns out that the probability the system is in the state  $s_i$ , with energy  $E(s_i)$ , is given by the Boltzmann distribution:

$$P(S = s_i) = \frac{\exp\{-E(s_i)/(K_B T)\}}{Z(T)}$$

where  $Z(T)$  is a partition function and  $K_B$  is known as the Boltzmann constant. The probability of the system having energy  $e$  is given by:

$$P(E = e) = \frac{g(e) \exp\{-e/(K_B T)\}}{Z(T)}$$

---

<sup>3</sup>The conservation law holds

where  $g(e)$  is the number of states having the energy  $e$ . In the above equations, the exponential form of the Boltzmann factor reflects that the number of states with a similar energy exponentially decays when the energy is approaching a minima or maxima. Intuitively, this is due to the exponential relationship of the entropy of the system. The consequence of this relationship has been implicitly stated by the second law of thermodynamics – an isolated system will tend toward equilibrium in which the number of possible states is largest, and equivalently, the entropy of the system is maximised. At a high temperature, the probability mass is distributed roughly evenly among all macrostates. As the temperature decreases, the probability mass of the Boltzmann distribution concentrates on the low energy states, and eventually the lowest energy state, since high energy states become increasingly impossible.

Simulated Annealing is a generalisation of Metropolis et al's algorithm [Metropolis et al. 1953] which employs only a single fixed temperature. The Metropolis algorithm is described as follows: Given a current state of the system with energy  $E_0$ , a new state is randomly chosen but accepted according to the Metropolis criterion. The criterion is that if the new state has energy  $E$ , and  $E < E_0$ , the system then moves to the new state. However, if  $E \geq E_0$ , the new state is accepted with a probability equal to  $\min\{1, \exp(-(E - E_0)/(K_B T))\}$ . When the procedure is repeated, the system will reach a state of equilibrium in which it has a Boltzmann distribution.

Instead of fixing temperature, Simulated Annealing introduces an annealing schedule to lower the temperature in phases. Beginning with very high temperature, where every state has nearly the same probability,  $\exp\{-E/T\} \approx \exp\{0\}$ , the Metropolis algorithm is followed until equilibrium is achieved. The temperature is then lowered and further iterations of the Metropolis algorithm followed according to the schedule. This procedure is iterated until the system freezes. If the annealing schedule is sufficiently slow, then the system will eventually freeze in the state with minimum energy. The Metropolis criterion is crucial to the success of Simulated Annealing, because it

allows probabilistic escape from local minima.

### 5.1.1 Simulated Annealing Particle Filter

Recalling Particle Filter, priori knowledge and the temporal model often are unknown and difficult to acquire, many studies eliminate the temporal term  $p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i)$  in the update equation A.2.3 by equating  $\pi(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i)$  to  $p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i)$  in order to simplify the calculation. This type of the Importance Sampling Re-sampling algorithm is commonly known as the bootstrap filter and condensation algorithm [Isard and Blake 1998b]. In this form, the posterior probability is solely dependent upon likelihood, and the maximum a posteriori solution is equivalent to the maximum likelihood solution. As a result, the problem is transferred from solving the maximum a posteriori to solving the maximum likelihood, eventually estimating the true state via optimisation.

Simulated Annealing as proposed by Kirkpatrick et al. [Kirkpatrick et al. 1983] serves as a general-purpose optimisation algorithm. Later, Deutscher et al [Deutscher and Reid 2005] introduced it for estimating the maximising likelihood of particle filtering in human motion tracking. Usually the likelihood probability  $p(\mathbf{y}_t | \mathbf{x}_t)$  is formulated as an exponential function  $f(\mathbf{y}_t, \mathbf{x}_t)$  with respect to a metric of  $E(\mathbf{y}_t, \mathbf{x}_t)$  between  $\mathbf{y}_t$  and  $\mathbf{x}_t$ .

$$f(\mathbf{y}_t, \mathbf{x}_t) = \exp\{-E(\mathbf{y}_t, \mathbf{x}_t)\}$$

Adding annealing variable  $\lambda$ :

$$p(\mathbf{y}_t | \mathbf{x}_t) = f(\mathbf{y}_t, \mathbf{x}_t)^\lambda = \exp\{-\lambda E(\mathbf{y}_t, \mathbf{x}_t)\} \quad (5.1.1)$$

When  $\lambda \rightarrow \infty$ , the probability mass will concentrate on the minimum of  $E(\mathbf{y}_t, \mathbf{x}_t)$ , or equivalently, the maximum of  $f(\mathbf{y}_t, \mathbf{x}_t)$ . To avoid being trapped in local minima,  $\lambda$  is initially assigned a small value, and is gradually increased according to a predefined set of values  $\{\lambda = \lambda_m, \dots, \lambda_1\}$ , where  $\lambda_m < \lambda_{m-1} < \dots < \lambda_1$ , which is known

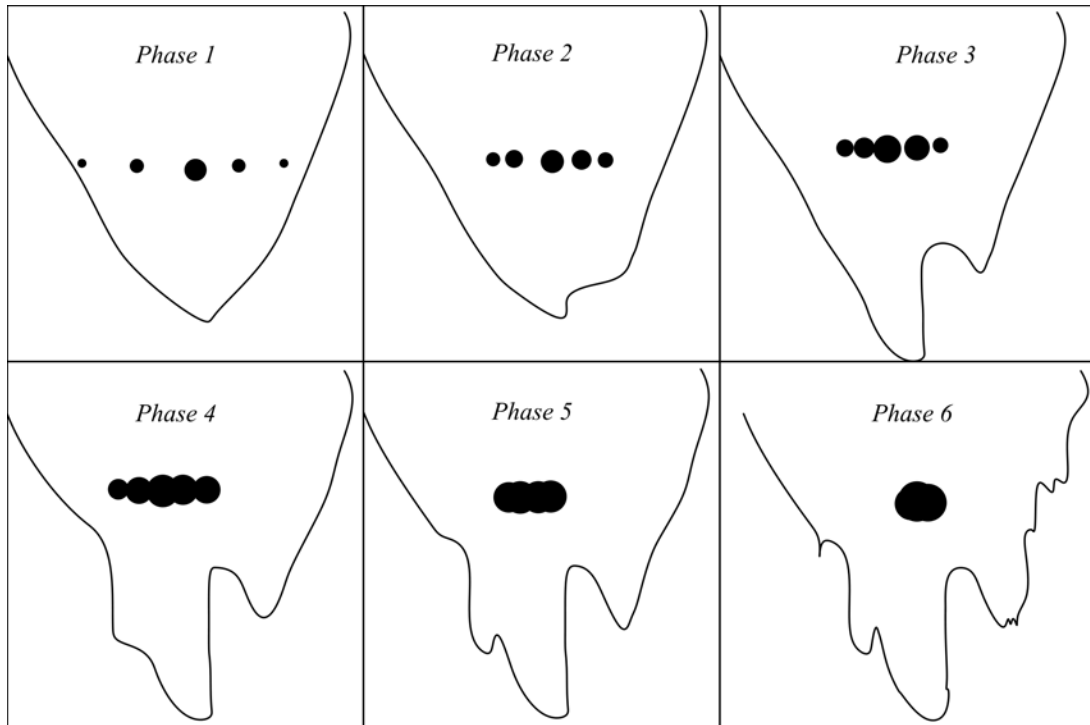


Figure 5.1: As  $\lambda$  gradually increases, the observation likelihood distribution evolves from flat (probably single mode) to peaked (multiple modes), and its probability mass slowly concentrates on the maxima, equivalently, the minima of  $E(\mathbf{y}_t, \mathbf{x}_t)$ . Particles gradually move from low to high likelihood areas while their coverage shrinks from large to small. Provided the annealing schedule is sufficiently slow and long enough, particles will eventually concentrate on the global maximum mode.

as the annealing scheduling. Gradually increasing of  $\lambda$  introduces the evolution of  $f(\mathbf{y}_t, \mathbf{x}_t)$  from a flat distribution (probably single mode) to a peaked distribution (multiple modes). In a typical process, the samples of the state  $\mathbf{x}_t$  are weighted by  $f(\mathbf{y}_t, \mathbf{x}_t)$ , re-sampled to concentrate on a better minimum and finally perturbed with Gaussian noise. Theoretically, the Simulated Annealing Particle Filter should not be misled by local minima, so it can converge to the global minimum within the search space. Figure 5.1 illustrates this evolving procedure.

Besides  $\lambda_m$ , another two important parameters are survival rate  $\alpha_m$  and a perturbation covariance matrix  $\mathbf{P}_m$ , which control and tune the pace at which samples are superseded and perturbed to concentrate on the minimum of the energy function. The



survival rate is given by:

$$\alpha_m = \frac{N_{eff}(m)}{N}$$

$$N_{eff}(m) = \frac{1}{\sum_{i=1}^N (w_{t,m}^i)^2}$$

where,  $N_{eff}(m)$  shares the same sense as equation 3.3.6, but has a different weight definition  $w_{t,m}^i = \exp\{-\lambda_m E(\mathbf{y}_t^i, \mathbf{x}_t^i)\}$ . A high survival rate corresponds to a flat importance distribution whose probability mass is uniformly distributed. Resampling from this distribution ensures that good and bad particles are roughly equally likely to be sampled, enabling a broad range of exploration. Conversely, a low survival rate with a peaked importance distribution ensures good particles are more likely to be sampled so that the exploration concentrates on highly likely areas in search space. Instead of simply increasing annealing variables, the annealing schedule should be determined by accounting for the shape of the importance distribution. This leads to more effective resampling in the probabilistically important directions. Given the survival rate  $\alpha_m$  of particles at the current phase,  $\lambda_m$  can be determined as suggested in [Deutscher and Reid 2005] by:

$$\alpha_m N \sum_{i=1}^N (w_{t,m}^i)^2 = \left( \sum_{i=1}^N w_{t,m}^i \right)^2 \quad (5.1.2)$$

where,  $N$  is the number of particles, and  $w_{t,m}^i = \exp\{-\lambda_m E(\mathbf{y}_t^i, \mathbf{x}_{t,m}^i)\}$ .

The survival rate at any phase can be assigned the desired value  $\alpha_{desired}$  by adjusting the annealing variable  $\lambda_m$ . Note survival rates are fixed  $\alpha_m = \dots = \alpha_1 = \alpha_{desired}$  and  $\lambda_m$  is monotonically increasing. This implies that the perturbation covariance matrix  $\mathbf{P}_m$  contributes to increasing uniformness of the probability mass between phases. As  $\lambda_m$  gradually increases, particles become closer and closer to the global minimum.  $\mathbf{P}_m$  is gradually scaled so that particles do not waste time exploring fruitless areas which are far away from the global minimum. The current perturbation covariance matrix

$\mathbf{P}_m$  can be scaled by assigning it to be proportional to the product of the survival rate and the previous covariance matrix. Thus  $\mathbf{P}_m$  can be given by:

$$\mathbf{P}_m = \alpha_m \times \dots \times \alpha_1 \times \mathbf{P}_0 = (\alpha_{diresed})^m \times \mathbf{P}_m$$

This is analogous to the situation where as the temperature falls, the energy of particles decreases and therefore the range of movement of the particles is squeezed.

The Simulated Annealing Particle Filter can be regarded as a particle filter with  $p(\mathbf{x}_t|\mathbf{x}_{t-1})$  as the importance distribution, and extra optimisation steps. The optimisation process takes place after the weights have been updated, and completed before the number of effective samples is evaluated. The Annealing Particle Filter is shown in Algorithm 5.

---

**Algorithm 5** Annealing Particle Filter for a typical frame at time  $t$

---

**Require:** appropriate  $\alpha_m$  is defined, previous particles  $\mathbf{x}_{t-1}$ , observation  $\mathbf{y}_t$ , the number of phases  $M$  and the initial covariance matrix  $\mathbf{P}_0$  are given

**for**  $m = 1$  to  $M$  **do**

- 1: Initialise  $N$  particles  $\mathbf{x}_t^i$  from the previous phase or the temporal model  $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$ .
- 2: Calculate the energy  $E(\mathbf{y}_t, \mathbf{x}_t)$  for all particles.
- 3: Find  $\lambda_m$  by solving an equation (5.1.2).
- 4: Update weights for all particles using equation (5.1.1).
- 5: Resample  $N$  particles from the importance distribution.
- 6: Perturb particles by Gaussian noise with covariance  $\mathbf{P}_m = \alpha_m \mathbf{P}_{m-1}$  and mean  $\mu = 0$ .

**end for**

---

## 5.2 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) was introduced by Kennedy and Eberhart [Kennedy and Eberhart 1995] in 1995 as a concept for the optimisation of continuous nonlinear functions. It was discovered through simulation of simplified social models and has roots in both artificial life and evolutionary computation. Because

---

**Algorithm 6** Basic Particle Swarm Optimisation

---

```

for each particle  $i = 1$  to  $N$  do
  Initialize the particle's position with a uniformly distributed random vector:  $\mathbf{x}_i \sim U(blo, bup)$ , where  $blo$  and  $bup$  are the lower and upper boundaries of the search-space.
  Initialize the particle's best known position to its initial position:  $\mathbf{x}_i^{(lbest)} = \mathbf{x}_i$ 
  if  $f(\mathbf{x}_i^{(lbest)}) < f(\mathbf{x}^{(gbest)})$  then
    update the swarm's best known position:  $\mathbf{x}^{(gbest)} = \mathbf{x}_i^{(lbest)}$ 
  end if
  Initialize the particle's velocity:  $\mathbf{v}_i \sim U(-|bup - blo|, |bup - blo|)$ 
end for
repeat
  for each particle  $i = 1$  to  $N$  do
    Generate random numbers:  $r_l, r_g \sim U(0, 1)$ 
    Update the particle's velocity:  $\mathbf{v}_i = w\mathbf{v}_i + c_l r_l (\mathbf{x}_i^{(lbest)} - \mathbf{x}_i) + c_g r_g (\mathbf{x}^{(gbest)} - \mathbf{x}_i)$ 
    Update the particle's position:  $\mathbf{x}_i = \mathbf{x}_i + \mathbf{v}_i$ 
    if  $f(\mathbf{x}_i) < f(\mathbf{x}_i^{(lbest)})$  then
      Update the particle's best known position:  $\mathbf{x}_i^{(lbest)} = \mathbf{x}_i$ 
      if  $f(\mathbf{x}_i^{(lbest)}) < f(\mathbf{x}^{(gbest)})$  then
        update the swarm's best known position:  $\mathbf{x}^{(gbest)} = \mathbf{x}_i^{(lbest)}$ 
      end if
    end if
  end for
until Termination criterion fulfilled (e.g. number of iterations performed, or adequate fitness reached).

```

---

of its simplicity and computational cheapness, PSO became an attractive population-based optimisation algorithm. In recent years it has been applied to many kinds of optimisation problems [Poli et al. 2007], [Poli 2008] and much research has been done in this field, resulting in many different PSO strategies. The fundamental hypothesis of PSO, however, has remained unchanged [Kennedy and Eberhart 1995]:

...social sharing of information among conspecifics offers an evolutionary advantage.

To a great extent PSO is inspired by nature and can be nicely described by an analogy. Consider a shoal looking for food. Each fish randomly swims around, hoping to get something to eat. It will remember places where it has previously encountered food, so it is drawn towards places with a high probability of success. But somehow it also knows about the place with the highest food density found so far by any of the other swarm members. So the fish is drawn towards that direction as well. Occasionally, a fish finds a location with an even bigger food density. Quickly, the whole shoal will be drawn to that location. With such a behaviour, the shoal explores the territory. Fishes fly over locations with high food density, constantly checking if they have found a better place before. Generally speaking, individuals gather and communicate. By sharing information, each individual increases its chance of survival. From a technical point of view, PSO is based on the communication among swarm members (particles) in order to find an optimal solution for a given optimisation problem. On a meta-level, information flow between individuals contributes to a paradigm referred to as Swarm Intelligence. Following the definition of Millonas [Millonas 1994], Swarm Intelligence is based on five principles. The Proximity Principle dictates that the population should be able to carry out simple space and time computations. The Quality Principle says the population should be able to respond to quality factors in the environment. The Principle of Diverse Response is that the population should not commit its activities along excessively narrow channels. The Principle of Stability says

the population should not change its mode of behaviour every time the environment changes. And finally, the Principle of Adaptability is that the population must be able to change behaviour mode when it is worth the computational price.

### 5.2.1 Algorithm Description

A particle can be described by its position  $\mathbf{x}$  and velocity  $\mathbf{v}$  in a multi-dimensional, problem-dependent search space ( $R^n$ ). While moving through the problem space, particles evaluate the given fitness (cost) function and update their velocity via an update rule that not only follows local, but also incorporates global information. Each particle keeps track of the best solution it has found so far (the position with the best fitness function value), called local best and in the following denoted as  $\mathbf{x}^{(lbest)}$ . And it is also aware of the current global best position ( $\mathbf{x}_{(gbest)}$ ), detected by any swarm member. If  $\mathbf{x}_t$  denotes the particle position at time  $t$ , the velocity and position update can be described by the following equations:

$$\begin{aligned}\mathbf{v}_{t+1} &= w\mathbf{v}_t + c_l r_l (\mathbf{x}^{(lbest)} - \mathbf{x}_t) + c_g r_g (\mathbf{x}^{(gbest)} - \mathbf{x}_t) \\ \mathbf{x}_{t+1} &= \mathbf{x}_t + \mathbf{v}_{t+1}\end{aligned}$$

The above equations have two main components: one is the attraction towards the particle's local best position, and the other the attraction towards the global best position.  $r_l$  and  $r_g$  are uniform random numbers in the interval  $[0, 1]$ , introducing a stochastic factor to the algorithm to simulate the unpredictable component of natural swarm behaviour.  $c_l$  and  $c_g$  are constants, defining how much local or global information influences the particles' movement, and are usually chosen as  $c_l = c_g = 2$ . This makes the mean of the stochastic factor equal to 1. In other words, the particles "over-fly" the target about half the time [Kennedy and Eberhart 1995].  $w$  denotes an inertial weight that controls the influence of the previous velocity. Because  $c_l$  pulls the particle toward its local best position, this factor is often called the "cognitive rate", and  $c_g$ , as

it draws the particle towards the global best position found by any swarm member, is called the “social rate”. The basic Particle Swarm Optimisation is summarised in Algorithm 6.

The choice of PSO parameters can have a large impact on optimisation performance. Selecting PSO parameters that yield good performance has therefore been the subject of much research [Shi and Eberhart 1998; Ebe 2000]. PSO parameters can also be tuned by using another overlaying optimiser, a concept known as meta-optimisation, in which parameters are simultaneously tuned for various optimisation scenarios [Pedersen and Chipperfield 2010].

### **5.3 Covariance Matrix Adaptation Evolution Strategy**

Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is a stochastic, iterative method for real-valued-parameter optimisation of non-linear, non-convex functions. It is a powerful optimisation procedure and performs especially well in rugged search-landscapes with discontinuities, noise, local optima, etc. CMA-ES employs Gaussian adaptation to learn a second order model of the underlying objective function, analogous to the approximation of the inverse Hessian matrix in a Quasi-Newton method. In some sense, CMA-ES is a second order approach to estimating a positive definite matrix, but unlike most second order methods, CMA-ES makes fewer assumptions about the underlying objective function. Only the ranking between candidate solutions is exploited for learning the sample distribution and neither derivatives nor even the function values themselves are required. Additionally, a so-called evolution path, containing information on the correlation between consecutive steps, is recorded and used for the covariance matrix adaptation mechanism as well as for an auxiliary step-size control.

CMA-ES was first introduced by Hansen and Ostermeier in 1996 [Hansen and Ostermeier 1996]. Major improvements on the initial idea made CMA-ES a highly

elaborate optimisation algorithm. In 2001, weighted recombination was introduced to CMA-ES [Hansen and Ostermeier 2001]. Two years later, the so-called rank- $\mu$ -update greatly reduced time complexity [Hansen et al. 2003]. Ongoing adjustments and modifications improved performance. It was found that global search characteristics can be enhanced if the population size is increased [Hansen and Kern 2004]. A very recent modification further reduces time and space complexity [Ros and Hansen 2008].

### 5.3.1 Evolution Strategy

The Evolution Strategy (ES) dates back to the mid 1960s when Bienert, Rechenberg, and Schwefel [Rechenberg 1973; Rechenberg 1994], at the Technical University of Berlin, Germany, developed the first bionics-inspired schemes for optimising the shape of minimum drag bodies in a wind tunnel using Darwinian principles of evolution. First of all, a population with a predefined number of individuals is initialised within the given problem space (mostly at random). A selection mechanism chooses individuals that will be considered parents of the next generation. Based on a recombination mechanism, offspring from the selected parent population will be created. In analogy to the concept of variation inheritance, in evolutionary computation mutations ensure that offspring resemble their parents, but are not identical. After creating the offspring, the selection operator will again choose individuals to be the parents of the next generation of offspring.

The canonical versions of ES, comma-selection and plus-selection, are denoted respectively by:

$$(\mu/\rho, \lambda)\text{-ES} \quad \text{and} \quad (\mu/\rho + \lambda)\text{-ES}$$

Here  $\mu$  denotes the number of parents,  $\rho \leq \mu$  the mixing number (i.e., the number of parents involved in the procreation of offspring), and  $\lambda$  the number of offspring. The parents are deterministically selected (i.e., deterministic survivor selection) from the (multi-)set of either the offspring, referred to as comma-selection ( $\mu < \lambda$  must

hold), or both the parents and offspring, referred to as plus-selection. Selection is based on the ranking of the individuals' fitness  $f(\mathbf{x})$  taking the  $\lambda$  best individuals (also referred to as truncation selection). In general, an ES individual  $\alpha = (\mathbf{x}, \mathbf{s}, f(\mathbf{x}))$  comprises the object parameter vector  $\mathbf{x}$  to be optimized, a set of strategy parameters  $\mathbf{s}$ , needed especially in self-adaptive ESs, and the individual's observed fitness  $f(\mathbf{x})$  being equivalent to the objective function. The simplest evolution strategy operates on a population of size two: the current point (parent) and the result of its mutation. Only when the offspring's fitness is at least as good as the parent one does it become a parent of the next generation, otherwise the offspring is disregarded. This is a  $(1 + 1)$ -ES. More generally,  $\lambda$  offspring can be generated and compete with the parent, called  $(1 + \lambda)$ -ES. In  $(1, \lambda)$ -ES the best offspring becomes the parent of the next generation while the current parent is always disregarded. The meta algorithm of the Evolution Strategy is given in Algorithm 7.

---

**Algorithm 7** Self-Adaptation-Evolution-Strategy

---

1. Initialize parent population  $\mathbf{P}_\mu = \{\alpha_1, \dots, \alpha_\mu\}$ .
  2. Generate  $\lambda$  offspring  $\hat{\alpha}$  forming the offspring population  $\hat{\mathbf{P}}_\lambda = \{\hat{\alpha}_1, \dots, \hat{\alpha}_\lambda\}$  where each offspring  $\hat{\alpha}$  is generated thusly:
    - 1) Select (randomly)  $\rho$  parents from  $\mathbf{P}_\mu$  (if  $\rho = \mu$  take all parental individuals instead).
    - 2) Recombine the  $\rho$  selected parents  $\alpha$  to form a recombinant individual  $\mathbf{r}$ .
    - 3) Mutate the strategy parameter set  $\mathbf{s}$  of the recombinant  $\mathbf{r}$ .
    - 4) Mutate the objective parameter set  $\mathbf{x}$  of the recombinant  $\mathbf{r}$  using the mutated strategy parameter set to control the statistical properties of the object parameter mutation.
  3. Select new parent population (using deterministic truncation selection) from
    - # either the offspring population  $\hat{\mathbf{P}}_\lambda$  (this is referred to as comma-selection, usually denoted as " $(\mu, \lambda)$ -selection"),
    - # or the offspring  $\hat{\mathbf{P}}_\lambda$  and parent  $\mathbf{P}_\mu$  population (this is referred to as plus-selection, usually denoted as " $(\mu + \lambda)$ -selection")
  4. Goto 2. until termination criterion fulfilled.
- 

Recombination and mutation play important roles in ES, and in the next section we introduce the state-of-the-art evolutionary optimisation CMA-ES and describe how



the recombination and mutation parameters for the search distribution are modelled.

### 5.3.2 Covariance Matrix Adaptation

As stated by the principle of maximum entropy, “the probability distribution which best represents the current state of knowledge is the one with the largest remaining uncertainty (maximum entropy) consistent with given constraints”. This principle avoids additional assumptions and provides the least biased estimate possible for any given information. Given a finite mean and variance (covariance), the (multivariate) Gaussian distribution has maximum entropy relative to all probability distributions covering the entire real line  $(-\infty, \infty)$ . The Gaussian Adaptation based CMA-ES is motivated by two principles: first, maximising entropy minimises the amount of prior information built into the distribution, preserving generality; second, many physical systems tend to move towards maximum entropy configurations over time.

In CMA-ES [Hansen and Ostermeier 1996; Hansen and Ostermeier 2001; Hansen and Kern 2004; Ros and Hansen 2008], the population of offspring for the next generation  $(g + 1)$  is generated by sampling a multivariate normal distribution with mean  $\mathbf{m} \in R^n$  and covariance  $\mathbf{C} \in R^{n \times n}$ . In addition, sampling is controlled by an overall standard deviation  $\sigma$ . Where  $\mathbf{x}_k^{(g)}$  denotes the  $k$ th individual at generation  $g$ , the sampling is given as:

$$\mathbf{x}_k^{(g)} = \mathbf{m}^{(g)} + \sigma^{(g)} N(\mathbf{0}, \mathbf{C}^{(g)}) \quad k = 1, \dots, \lambda \quad (5.3.1)$$

The obvious problems are how to calculate  $\mathbf{m}$ ,  $\mathbf{C}$  and  $\lambda$  for the next generation  $(g + 1)$ .

#### 5.3.2.1 Selection and Recombination

From the  $k$  sampled points,  $\mu$  points are selected and ranked in ascending order, according to their fitness values. The new mean value can be calculated as a weighted

intermediate recombination of the selected points:

$$\mathbf{m}^{(g+1)} = \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g+1)} \quad \sum_{i=1}^{\mu} w_i = 1 \quad w_1 \geq w_2 \geq \dots \geq w_{\mu} > 0 \quad (5.3.2)$$

where  $w_i$  are positive weight coefficients for recombination and  $x_{i:\lambda}^{(g+1)}$  denotes the  $i$ th ranked individual of the  $\lambda$  sampling points  $x_k^{(g+1)}$ . For example, selected sample points could be equally weighted ( $w_i = 1/\mu$ ). A measure, the variance effective selection mass, is defined as:

$$\mu_{eff} = \left( \frac{\|\mathbf{w}\|_1^2}{\|\mathbf{w}\|_2^2} \right) = \left( \sum_{i=1}^{\mu} w_i^2 \right)^{-1}. \quad (5.3.3)$$

Variance effective selection mass is in the range of  $1 \leq \mu_{eff} \leq \mu$ , and  $\mu_{eff} = \mu$  for equal recombination weights, i.e.  $w_i = 1/\mu$  for all  $i = 1, \dots, \mu$ . Usually,  $\mu_{eff} \approx \lambda/4$  indicates a reasonable setting of  $w_i$ . Typical settings could be  $w_i \propto \mu - i + 1$ , and  $\mu \approx \lambda/2$ .

### 5.3.2.2 Adapting the Covariance Matrix

In this section, the update of the covariance matrix,  $\mathbf{C}$ , is derived. We will start out by reviewing some properties of the covariance matrix, followed by constructing the covariance matrix from a single population of one generation. For small populations, the estimation is unreliable and an adaptation procedure rank- $\mu$ -update is introduced. In the limit case where only a single point can be used to update (adapt) the covariance matrix at each generation, the rank-one-update is employed. Finally the adaptation can be enhanced by exploiting dependencies between successive steps applied cumulatively via evolutionary paths.

**Covariance Matrix** Covariance Matrix is a symmetric, positive-definite matrix,  $\mathbf{C} \in \mathbb{R}^{n \times n}$ . All eigenvalues of  $\mathbf{C}$  are positive, and eigenvectors of  $\mathbf{C}$  are orthonormal

and forming a basis. Particularly,  $\mathbf{C}^{-1}$ ,  $\mathbf{C}^{\frac{1}{2}}$  and  $\mathbf{C}^{-\frac{1}{2}}$  can be derived as:

$$\begin{aligned}\mathbf{C}^{-1} &= (\mathbf{B}\mathbf{D}^2\mathbf{B}^T)^{-1} \\ &= \mathbf{B}\mathbf{diag}(1/d_1^2, \dots, 1/d_n^2)\mathbf{B}^T \\ \mathbf{C}^{\frac{1}{2}} &= \mathbf{B}\mathbf{D}\mathbf{B}^T \\ \mathbf{C}^{-\frac{1}{2}} &= \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^T \\ &= \mathbf{B}\mathbf{diag}(1/d_1, \dots, 1/d_n)\mathbf{B}^T\end{aligned}$$

where the column vectors of  $\mathbf{B}$  are eigenvectors and the diagonal elements of  $\mathbf{D}^2$  are eigenvalues. A multivariate normal distribution,  $N(\mathbf{m}, \mathbf{C})$  can be sampled in different ways:

$$\begin{aligned}N(\mathbf{m}, \mathbf{C}) &= \mathbf{m} + N(\mathbf{0}, \mathbf{C}) \\ &= \mathbf{m} + \mathbf{C}^{\frac{1}{2}}N(\mathbf{0}, \mathbf{I}) \\ &= \mathbf{m} + \mathbf{B}\mathbf{D}\mathbf{B}^TN(\mathbf{0}, \mathbf{I}) \\ &= \mathbf{m} + \mathbf{B}\mathbf{D}N(\mathbf{0}, \mathbf{I})\end{aligned}$$

Hence, sampling from  $N(\mathbf{m}, \mathbf{C})$  becomes easy, as  $N(\mathbf{0}, \mathbf{I})$  is a vector of independent  $(0, 1)$ -normally distributed numbers that can easily be determined computationally.

Considering a multivariate Gaussian random variable  $\mathbf{x}$  with mean  $\mathbf{m}$  and covariance matrix  $\mathbf{C}$ , the joint probability density function is given as:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\mathbf{C}|^{1/2}} \exp -\frac{1}{2}(\mathbf{x} - \mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x} - \mathbf{m})$$

The objective function can be defined as its negative logarithm:

$$f(\mathbf{x}) = -\ln(p(\mathbf{x})) = \frac{n}{2} \ln 2\pi + \frac{1}{2} \ln |\mathbf{C}| + \frac{1}{2}(\mathbf{x} - \mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x} - \mathbf{m})$$

which is a quadratic function of the components in  $\mathbf{x}$ . By 2nd-order partial derivatives evaluated at  $\mathbf{x}$ , we obtain that the Hessian matrix is equal to the inverse of the covariance matrix,  $\mathbf{H} = \mathbf{C}^{-1}$ . The optimal covariance matrix equals the inverse Hessian matrix with a constant factor. Consequently, the objective of covariance matrix adaptation is to approximate the inverse Hessian matrix, similar to a quasi-Newton method. More generally, the objective is to fit the search distribution to the contour lines of the objective function  $f$ .

**Covariance Matrix From One Generation** We assume that the population contains enough information to reliably estimate a covariance matrix from the population and  $\sigma = 1$ . The empirical covariance matrix is given as:

$$\mathbf{C}_{emp}^{(g+1)} = \frac{1}{\lambda - 1} \sum_{i=1}^{\lambda} \left( \mathbf{x}_i^{(g+1)} - \frac{1}{\lambda} \sum_{j=1}^{\lambda} \mathbf{x}_j^{(g+1)} \right) \left( \mathbf{x}_i^{(g+1)} - \frac{1}{\lambda} \sum_{j=1}^{\lambda} \mathbf{x}_j^{(g+1)} \right)^T$$

The empirical covariance matrix  $\mathbf{C}_{emp}^{(g+1)}$  is an unbiased estimator of  $\mathbf{C}^{(g)}$ . Instead of taking the mean value of the actual samples for the covariance calculation, the true mean value  $\mathbf{m}^{(g)}$  of the sample distribution serves as the reference point:

$$\mathbf{C}_{\lambda}^{(g+1)} = \frac{1}{\lambda} \sum_{i=1}^{\lambda} \left( \mathbf{x}_i^{(g+1)} - \mathbf{m}^{(g)} \right) \left( \mathbf{x}_i^{(g+1)} - \mathbf{m}^{(g)} \right)^T$$

Also the matrix  $\mathbf{C}_{\lambda}^{(g+1)}$  is an unbiased estimator of  $\mathbf{C}^{(g)}$ . Using only selected samples and introducing weighting just as in Equation 5.3.3, the covariance matrix is calculated as:

$$\mathbf{C}_{\mu}^{(g+1)} = \sum_{i=1}^{\mu} w_i \left( \mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g)} \right) \left( \mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g)} \right)^T \quad (5.3.4)$$

where  $\mathbf{x}_{i:\lambda}^{(g+1)}$  is  $i$ th best individual out of  $\mathbf{x}_1^{(g+1)}, \dots, \mathbf{x}_{\lambda}^{(g+1)}$  and  $f(\mathbf{x}_{1:\lambda}^{(g+1)}) \leq f(\mathbf{x}_{2:\lambda}^{(g+1)}) \leq \dots \leq f(\mathbf{x}_{\lambda:\lambda}^{(g+1)})$ .  $\mathbf{C}_{\mu}^{(g+1)}$  can be interpreted as an estimator for the distribution of selected steps. This means, that sampling from  $\mathbf{C}_{\mu}^{(g+1)}$  tends to reproduce selected steps. Comparing (5.3.4) with the Estimation of Multivariate Normal Algorithm (EMNA)

[Lar 2001], the covariance matrix in EMNA is given as:

$$\mathbf{C}_{EMNA}^{(g+1)} = \frac{1}{\mu} \sum_{i=1}^{\mu} \left( \mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g+1)} \right) \left( \mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g+1)} \right)^T \quad (5.3.5)$$

where  $\mathbf{m}^{(g+1)} = \frac{1}{\mu} \sum_{i=1}^{\mu} \mathbf{x}_{i:\lambda}^{(g+1)}$ . Equation (5.3.5) estimates the variance within the selected population while (5.3.4) estimates selected steps. Equation (5.3.5) always reveals smaller variances than (5.3.4), because its reference mean value is the minimiser for the variances. Moreover, in most conceivable selection situations (5.3.5) decreases the variances compared to  $\mathbf{C}^{(g)}$ . Equation (5.3.4) geometrically increases the expected variance in the direction of the gradient where the selection takes place, while Equation (5.3.5) always decreases the variance in the gradient direction geometrically. Therefore, (5.3.5) is highly susceptible to premature convergence<sup>4</sup>, in particular with small parent populations, where the population cannot be expected to bracket the optimum at any time. With Equation (5.3.4), in order to ensure that  $\mathbf{C}_{\mu}^{(g+1)}$  is a reliable estimator, the variance effective selection mass  $\mu_{eff}$  must be large enough. Getting condition numbers smaller than ten for  $\mathbf{C}_{\mu}^{(g)}$  in  $f_{sphere}(\mathbf{x}) = \sum_{i=1}^n \mathbf{x}_i^2$ , requires  $\mu_{eff} \approx 10n$ .

**Rank- $\mu$ -Update** To achieve fast search (as opposed to more robust or more global search), the population size  $\lambda$  must be small.  $\mu_{eff}$  also must be small (e.g.  $\mu_{eff} \approx \lambda/4$ ). In this case it is not possible to get a reliable estimator for a good covariance matrix from Equation (5.3.4). One possible remedy could be to use additional information from previous generations. For example, after a sufficient number of generations, the mean of the estimated covariance matrices from all generations,

$$\mathbf{C}^{(g+1)} = \frac{1}{g+1} \sum_{i=0}^g \frac{1}{\sigma^{(i)^2}} \mathbf{C}_{\mu}^{(i+1)}$$

becomes a reliable estimator for the selected steps. To allow recent generations to

<sup>4</sup>However, for large values of  $\mu$  in large populations with large initial variances, the impact of the different reference mean value can become marginal.

have stronger influences on the covariance matrix adaptation, exponential smoothing is introduced to put more weight on recent generations. Given  $\mathbf{C}^{(0)} = \mathbf{I}$  and a learning rate  $c_\mu \in [0, 1]$ , the iterative update is:

$$\begin{aligned} \mathbf{C}^{(g+1)} &= (1 - c_\mu)\mathbf{C}^{(g)} + c_\mu \frac{1}{\sigma^{(g)^2}} \mathbf{C}_\mu^{(g+1)} \\ &= (1 - c_\mu)\mathbf{C}^{(g)} + c_\mu \sum_{i=1}^{\mu} w_i \mathbf{y}_{i:\lambda}^{(g+1)} \mathbf{y}_{i:\lambda}^{(g+1)T} \end{aligned} \quad (5.3.6)$$

$$= (1 - c_\mu)^{g+1} \mathbf{C}^{(0)} + c_\mu \sum_{i=0}^g (1 - c_\mu)^{g-i} \frac{1}{\sigma^{(i)^2}} \mathbf{C}_\mu^{(i+1)} \quad (5.3.7)$$

where  $\mathbf{y}_{i:\lambda}^{(g+1)} = (\mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{m}^{(g)})/\sigma^{(g)}$ . The backward time horizon is defined by  $\Delta g$  where about 63% of the overall weight is summed up and

$$c_\mu \sum_{i=g+1-\Delta g}^g (1 - c_\mu)^{g-i} \approx 0.63 \approx 1 - e^{-1}$$

Resolving the sum yields:

$$(1 - c_\mu)^{\Delta g} \approx e^{-1}$$

Using the Taylor approximation for the natural logarithm yields:

$$\Delta g \approx c_\mu^{-1}$$

That is, approximately 37% of the information in  $\mathbf{C}^{(g+1)}$  is older than  $c_\mu^{-1}$  generations, and the original weight is reduced by a factor of 0.37 after approximately  $c_\mu^{-1}$  generations. It turns out that  $c_\mu$  can be calculated by the first order approximation  $c_\mu \approx \mu_{eff}/n^2$ . For a fixed number of function evaluations, a small population size  $\lambda$  allows a larger number of generations and therefore usually leads to a faster adaptation of the covariance matrix.

### Rank-One-Update

The rank-one-update uses only a single selected step for the covariance matrix

adaptation. Together with the concept of cumulation, correlations between consecutive steps are exploited to give the so-called evolution path. Conceptually, the evolution path is the path that the strategy takes over a number of generations. The single rank one covariance matrix update can be derived from Equation (5.3.6) as:

$$\mathbf{C}^{(g+1)} = (1 - c_1)\mathbf{C}^{(g)} + c_1\mathbf{y}_{g+1}\mathbf{y}_{g+1}^T$$

where  $\mathbf{y}_{g+1} = \frac{\mathbf{x}_{1:\lambda}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}}$ . The second term of rank one  $c_1\mathbf{y}_{g+1}\mathbf{y}_{g+1}^T$  adds the maximum likelihood term for  $\mathbf{y}_{g+1}$  into the covariance matrix  $\mathbf{C}^{(g)}$ . Ideally,  $\mathbf{y}_{g+1}$  is a fit individual. Therefore the likelihood of generating the sample  $\mathbf{y}_{g+1}$  in the next generation increases. However, since  $\mathbf{y}\mathbf{y}^T = -\mathbf{y}(-\mathbf{y})^T$ , the sign of the step is irrelevant for the update of the covariance matrix. To exploit the sign information, the evolution path is introduced. This can be expressed as the sum over consecutive steps of the mean value  $\mathbf{m}$ . The recursive construction of the evolution path,  $\mathbf{p}_c^{(g+1)} \in R^n$ , with exponential smoothing, with  $\mathbf{p}_c^{(0)} = 0$ , is referred to as cumulation:

$$\mathbf{p}_c^{(g+1)} = (1 - c_c)\mathbf{p}_c^{(g)} + \sqrt{c_c(2 - c_c)\mu_{eff}} \frac{\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} \quad (5.3.8)$$

where  $\sqrt{c_c(2 - c_c)\mu_{eff}}$  is a normalisation constant for  $\mathbf{p}_c$ , such that  $\mathbf{p}_c^{(g+1)} \sim N(\mathbf{0}, \mathbf{C})$ . A backward time horizon  $1/c_c$  between  $\sqrt{n}$  and  $n$  is reasonable. Then rank one update of the covariance matrix  $\mathbf{C}^{(g)}$  via the evolution path  $\mathbf{p}_c^{(g+1)}$  is:

$$\mathbf{C}^{(g+1)} = (1 - c_1)\mathbf{C}^{(g)} + c_1\mathbf{p}_c^{(g+1)}\mathbf{p}_c^{(g+1)T} \quad (5.3.9)$$

**Combining Rank- $\mu$ -Update and Rank-One-Update** Combining the rank- $\mu$ -update (5.3.6) and rank-one-update (5.3.9), the final covariance matrix update rule can be expressed as:

$$\mathbf{C}^{(g+1)} = (1 - c_1 - c_\mu)\mathbf{C}^{(g)} + c_1\mathbf{p}_c^{(g+1)}\mathbf{p}_c^{(g+1)T} + c_\mu \sum_{i=1}^{\mu} w_i \mathbf{y}_{i:\lambda}^{(g+1)} \mathbf{y}_{i:\lambda}^{(g+1)T} \quad (5.3.10)$$

where  $c_1 \approx 2/n^2$  and  $c_\mu \approx \min(\mu_{eff}/n^2, 1 - c_1)$ . On the one hand, the information within the population of one generation is used efficiently by the rank- $\mu$ -update to deal with large populations. On the other hand, information on correlations between generations is exploited by using the evolution path for the rank-one-update to small populations. The former is important in large populations, the latter is particularly important in small populations.

### 5.3.2.3 Step-Size Control

In addition to the covariance matrix adaptation rule, a step-size control is introduced, that adapts the overall scale of the distribution based on information obtained by the evolution path. The following rationale is applied: If the evolution path is long and single steps are pointing more or less to the same direction, the step-size should be increased. On the other hand, if the evolution path is short and single steps cancel each other out, the step-size should be decreased. Similar to Equation 5.3.9, the step-size evolution path  $p_\sigma$  is initialized with  $p_\sigma^{(0)} = 0$  in generation 0 and in the subsequent generations calculated as:

$$\mathbf{p}_\sigma^{(g+1)} = (1 - c_\sigma)\mathbf{p}_\sigma^{(g)} + \sqrt{c_\sigma(2 - c_\sigma)\mu_{eff}}\mathbf{C}^{(g)-\frac{1}{2}} \frac{\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} \quad (5.3.11)$$

with  $c_\sigma < 1$  again being the backward time horizon of the evolution path. When  $c_\sigma = 1$ , only the most recent step contributes to the cumulation.  $\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}$  gives the current step and  $\frac{\sqrt{c_\sigma(2-c_\sigma)\mu_{eff}}}{\sigma^{(g)}}$  is a normalisation constant. The main difference from Equation (5.3.9) is the term  $\mathbf{C}^{(g)-\frac{1}{2}}$ , representing a transformation, which makes the expected length of  $\mathbf{p}_\sigma^{(g+1)}$  independent of its direction.

Now a step-size adaptation rule can be formulated. Hansen reflects that selection ideally does not bias the length of the evolution path,  $\|\mathbf{p}_\sigma^{(g+1)}\|$ , and that the length is equal to its expected length under random selection, which is simply equal to the expected length of a random normal vector,  $E\|N(\mathbf{0}, \mathbf{I})\|$ . Comparing  $\|\mathbf{p}_\sigma^{(g+1)}\|$  and



$E\|N(\mathbf{0}, \mathbf{I})\|$  results in the final step-size adaptation rule:

$$\sigma^{(g+1)} = \sigma^{(g)} \exp \left( \frac{C_\sigma}{d_\sigma} \left( \frac{\|\mathbf{p}_\sigma^{(g+1)}\|}{E\|N(\mathbf{0}, \mathbf{I})\|} - 1 \right) \right) \quad (5.3.12)$$

where  $d_\sigma \approx 1$ , a damping parameter, scales the change magnitude of  $\ln \sigma^{(g)}$ . Overall, CMA-ES is outlined in Algorithm 8.

---

**Algorithm 8** Covariance Matrix Adaptation-Evolution Strategy

---

**Require:**

Selection and Recombination:  $\lambda = 4 + \lfloor 3 \ln n \rfloor$ ,  $\mu = \lfloor \mu' \rfloor$ ,  $\mu' = \lambda/2$ ,  $w_i = \frac{w'_i}{\sum_{j=1}^\mu w'_j}$ ,  
 $w'_i = \ln(\mu' + 0.5) - \ln i$  for  $i = 1, \dots, \mu$   
Step-size control:  $c_\sigma = \frac{\mu_{eff} + 2}{n + \mu_{eff} + 5}$ ,  $d_\sigma = 1 + 2 \max \left( 0, \sqrt{\frac{\mu_{eff} - 1}{n + 1}} - 1 \right) + c_\sigma$   
Covariance matrix adaptation:  $c_c = \frac{4 + \mu_{eff}/n}{n + 4 + 2\mu_{eff}/n}$ ,  $c_1 = \frac{2}{(n + 1.3)^2 + \mu_{eff}}$ ,  $c_\mu = \min \left( 1 - c_1, \alpha_\mu \frac{\mu_{eff} - 2 + 1/\mu_{eff}}{(n + 2)^2 + \alpha_\mu \mu_{eff}/2} \right)$  with  $\alpha_\mu = 2$

**Initialisation:**

Set evolution paths  $\mathbf{p}_\sigma = \mathbf{0}$ ,  $\mathbf{p}_c = \mathbf{0}$ , covariance matrix  $\mathbf{C} = \mathbf{I}$ , and  $g = 0$ . Choose distribution mean  $\mathbf{m} \in R^n$  and step-size  $\sigma \in R_+$  (problem dependent)

**repeat**

$g = g + 1$

Sample new population of search points, for  $k = 1, \dots, \lambda$

$$\mathbf{z}_k = N(\mathbf{0}, \mathbf{I})$$

$$\mathbf{y}_k = \mathbf{B} \mathbf{D} \mathbf{z}_k$$

$$\mathbf{x}_k = \mathbf{m} + \sigma \mathbf{y}_k$$

Selection and recombination

$$\tilde{\mathbf{y}}_w = \sum_{i=1}^\mu w_i \mathbf{y}_{i:\lambda} \text{ where } \sum_{i=1}^\mu w_i = 1, w_i > 0$$

$$\mathbf{m} = \mathbf{m} + \sigma \tilde{\mathbf{y}}_w = \sum_{i=1}^\mu w_i \mathbf{y}_{i:\lambda}$$

Step size control

$$\mathbf{p}_\sigma = (1 - c_\sigma) \mathbf{p}_\sigma + \sqrt{c_\sigma(2 - c_\sigma)\mu_{eff}} \mathbf{C}^{-1/2} \tilde{\mathbf{y}}_w$$

$$\sigma = \sigma \exp \left( \frac{C_\sigma}{d_\sigma} \left( \frac{\|\mathbf{p}_\sigma\|}{E\|N(\mathbf{0}, \mathbf{I})\|} - 1 \right) \right)$$

Covariance matrix adaptation

$$\mathbf{p}_c = (1 - c_c) \mathbf{p}_c + \sqrt{c_c(2 - c_c)\mu_{eff}} \tilde{\mathbf{y}}_w$$

$$\mathbf{C} = (1 - c_1 - c_\mu) \mathbf{C} + c_1 \mathbf{p}_c \mathbf{p}_c^T + c_\mu \sum_{i=1}^\mu w_i \mathbf{y}_{i:\lambda} \mathbf{y}_{i:\lambda}^T$$

**until** termination criterion fulfilled.

---

## 5.4 Covariance Matrix Adaptation Annealing

We propose a new global optimisation method, Covariance Matrix Adaptation Annealing, which is a hybrid of CMA-ES, Simulated Annealing and Particle Swarm Optimisation. Generally, it is designated a generic black box optimisation, with no analytical objective function required. Specifically, it is aimed at solving the multimodal problem in high dimensional space. It assumes that the initial position is close enough to the global optimum that the small search space around the global optimum can be reasonably densely sampled<sup>5</sup>. Comparing with Simulated Annealing, it has a faster convergence speed, but a higher chance of missing the global optimum. On the other hand, compared with CMA-ES, it has relatively slow convergence speed on a locally consistent landscape, but is more robust to multimodal landscapes. Therefore, it suits a well-defined specific category of problems to be solved. Following this section, we describe the category of problems can be solved efficiently by Covariance Matrix Adaptation Annealing.

### 5.4.1 Problems in Dynamic Settings

Covariance Matrix Adaptation Annealing is specifically suited to solving global optimisation in a dynamic setting. In a dynamic setting, optimisation at each point in time has very strong temporal coherence. On the other hand, for many practical problems, the global optimum has major influence on the entire landscape of the energy function. To avoid overcomplicated energy functions with golf-course-like landscapes, we impose smoothness constraints near the global optimum, allowing faster and more robust convergence. These together reveal three characteristics of problems in this category:

1. The optimisation problems for consecutive times have similar landscapes. In fact, the landscape from the current time is some evolution of the landscape

---

<sup>5</sup>The density of sampling and the size of the search space are determined by the distance between the initial position and the global optimum.

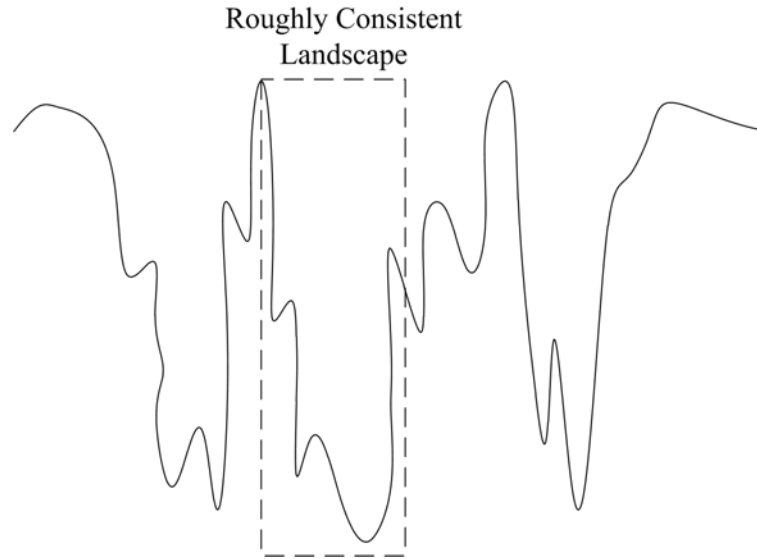


Figure 5.2: In the multimodal landscape situation, if a good initial position can be obtained from the previous time and there is a roughly consistent landscape with respect to the global optimum, the global optimum can be efficiently found by Covariance Matrix Adaptation Annealing.

from the previous time.

2. The optimal solutions from consecutive times also appear relatively close to each other. The previous solution can be used as a good initial position for current optimisation.
3. Within the small space (with respect to the search space) around the global optimum, the landscape is roughly consistent with the shape of the global optimum. This means that on the whole this small space forms a rough convex shape that leads to the global optimum, as illustrated in Figure 5.2.

#### 5.4.2 Covariance Matrix Adaptation Annealing Algorithm

The proposed algorithm has the several advantages: 1) It drives the landscape transition of the energy function from roughly convex to the original peaked shape. This allows particles to gradually move towards to the global mode; 2) The perturbation is not isotropic in all directions, but tends to perturb the samples along the weighted

directions conforming to the observation likelihood; 3) The accumulative path similar to the evolution path of CMA-ES is incorporated into the perturbation matrix to maintain the primary trend. This prevents the perturbation directions from changing dramatically and reduces the chance of oscillating successive perturbations which cancel out each other; and 4) It is inspired by human social behaviour, where individuals are attracted by the combined behaviour of both successful individuals and majority interests. This extra influence is introduced, allowing each sample to move closer to better fit samples, or the mean sample. All samples are expected to move towards to the centre and gradually concentrate. The algorithm preserves the major features of Simulated Annealing; the major differences only exist in the perturbation matrix update.

Initially, a series of survive rates  $\alpha_1, \dots, \alpha_M$  for  $M$  phases are defined. A large survival rate suggests slow convergence and is often used with a large number of phases. Conversely, a small survive rate creates a sharply shaped importance distribution, resulting in fast convergence, and is suitable for a small number of phases. In our experience, it is recommended that  $\alpha$  be varied from 0.3 to 1. The perturbation matrix  $\mathbf{P}_0$  is initialised to account for maximum perturbation according to the specific context. If the perturbation matrix from the previous time is available,  $\mathbf{P}_{0,t}$  is initialised by:

$$\mathbf{P}_{0,t} = \beta \mathbf{P}_{0,t-1} + (1 - \beta) \mathbf{P}_0$$

where  $\beta$  controls the balance between the dynamic part  $\mathbf{P}_{0,t-1}$  from the previous time and the stable part  $\mathbf{P}_0$  as a hard constraint. Otherwise,  $\mathbf{P}_{0,t} = \mathbf{P}_0$ . If the previous  $N$  samples/particles  $\mathbf{x}_{t-1}$  are available, the  $N$  particles  $\mathbf{x}_t^i$  can be initialised with the temporal model. Otherwise, particles  $\mathbf{x}_t^i$  are initialised by applying Gaussian perturbation with covariance  $\mathbf{P}_{0,t}$  and mean  $\mu = 0$  to the given initial position.

Then, for all particles  $\mathbf{x}_t^i$ , we evaluate the energy function  $E(\mathbf{y}_t, \mathbf{x}_t^i)$  to obtain the corresponding energy  $e^i$ . The global optimum is obtained when this energy function

is minimised. It is assumed to be positive. As the evaluation of the energy function forms the computational bottleneck, it is desirable to design the energy function to be quickly computed. After determining the energy for all particles, we want to form the importance distribution which has the specific survive rate  $\alpha_m$ . This is done by solving Equation (5.1.2) to find  $\lambda_m$  such that the importance distribution has the desired shape. We use the resulting  $\lambda_m$  to update the weights for all particles. Therefore, all particles associated with weights can be regarded as an approximation of the importance distribution. When  $N$  particles are resampled from the importance distribution, the offspring have a higher chance of coming from better parent particles. Roughly speaking,  $100 \cdot \lambda_m$  percent of particles will have offspring, and other particles will be obsoleted. This is how survive rate shapes the importance distribution, influences particle selection and eventually controls the convergence speed.

#### 5.4.2.1 Perturbation Matrix and Particle Velocity Update

The new particles are derived from three major factors: 1) direct resampling from the importance distribution  $\hat{\mathbf{x}}^i$ , 2) Gaussian perturbation with adaptive covariance  $\delta^i$ , and 3) the particle velocity imposed by attraction from superior individuals  $\mathbf{v}^i$ . This is formulated by:

$$\mathbf{x}^i = \hat{\mathbf{x}}^i + (1 - c_v)\delta^i + c_v\mathbf{v}^i$$

$\hat{\mathbf{x}}^i$  is similar to the resampled particle in Simulated Annealing and  $\delta^i$  is Gaussian perturbation generated by  $N(0, (\prod_{i=1}^m \alpha_i) \mathbf{P}_m)$ . The covariance  $\mathbf{P}_m$  is learned adaptively throughout the course of optimisation and progressively scaled by the survive rate to simulate cooling and freezing of the particles' movement. Different from Gaussian perturbation in many aspects, the particle velocity  $\mathbf{v}^i$  drives the particle to move in the direction approaching the global best and mean particles. This is similar to the PSO algorithm.  $c_v \in [0, 1]$  is a parameter used to control the contributions from Gaussian perturbation and particle velocity.  $c_v > 0.5$  favours fast convergence in the roughly

convex situation. Conversely,  $c_v < 0.5$  favours broader exploration of the multimodal landscape.

In the Gaussian perturbation, the covariance matrix  $\mathbf{P}_m$  is calculated adaptively, similar to Rank- $\mu$ -Update and Rank-One-Update in CMA-ES, which can be mathematically represented as:

$$\begin{aligned}\bar{\mathbf{x}} &= \sum_{i=1}^N w_i \mathbf{x}_i \\ \mathbf{p}_c &= (1 - c_c) \mathbf{p}_c + c_c (\bar{\mathbf{x}}_m - \bar{\mathbf{x}}_{m-1}) \\ \mathbf{P}_m &= (1 - c_1 - c_\mu) \mathbf{P}_{m-1}^i + c_1 \mathbf{p}_c \mathbf{p}_c^T + c_\mu \sum_{i=1}^N w_i (\mathbf{x}^i - \bar{\mathbf{x}}_m) (\mathbf{x}^i - \bar{\mathbf{x}}_m)^T\end{aligned}$$

where Rank- $\mu$ -Update  $c_\mu \sum_{i=1}^N w_i (\mathbf{x}^i - \bar{\mathbf{x}}_m) (\mathbf{x}^i - \bar{\mathbf{x}}_m)^T$  is observation likelihood weighted covariance to shape the perturbation direction, expected to help push the next particle perturbations towards the exploration of more favourable regions. Rank-One-Update  $c_1 \mathbf{p}_c \mathbf{p}_c^T$  incorporates historical information to smooth the perturbation directions. This is done by enforcing that particles move along the accumulative path of the mean particle  $\bar{\mathbf{x}}$  without excessive oscillations. The parameters  $c_c$ ,  $c_\mu$  and  $c_1$  are used to control the contributions of Rank- $\mu$ -Update and Rank-One-Update, as well as control exponential smoothing between phases. In our experience, with  $c_c = 0.5$ ,  $c_\mu = 1/3$  and  $c_1 = 1/3$  we obtain reasonable results for general cases.

The particle velocity is perturbed to simulate social behaviour and interaction between particles, through attraction to the best individual and the mean individual, as addressed in the formulation below:

$$\mathbf{v}^i = (1 - r_b) \frac{\bar{\mathbf{x}}_m - \hat{\mathbf{x}}^i}{\|\bar{\mathbf{x}}_m - \hat{\mathbf{x}}^i\|_2} + r_b \frac{\mathbf{x}^{(best)} - \hat{\mathbf{x}}^i}{\|\mathbf{x}^{(best)} - \hat{\mathbf{x}}^i\|_2}$$

where  $r_b$  denotes a uniform random variable in  $[1, 0]$ . Thus,  $\mathbf{v}^i$  is a vector originating at  $\hat{\mathbf{x}}^i$  and pointing to the range between  $\bar{\mathbf{x}}_m$  and  $\mathbf{x}^{(best)}$ . The squared  $L_2$  denominator is designed to simulate weakening influence caused by increasing the distance. This is

also consistent with the influence weakening in a social network where the physical distance between two individuals is increasing. Therefore, if  $\hat{\mathbf{x}}^i$  is far away from both  $\bar{\mathbf{x}}_m$  and  $\mathbf{x}^{(best)}$ , the particle velocity has a very small magnitude and Gaussian perturbation  $\delta^i$  will dominate the new position of the particle. If  $\hat{\mathbf{x}}^i$  is far from  $\bar{\mathbf{x}}_m$  and close to  $\mathbf{x}^{(best)}$ , the particle velocity will be dominated by  $\mathbf{x}^{(best)} - \hat{\mathbf{x}}^i$ . Conversely, if  $\hat{\mathbf{x}}^i$  is far from  $\mathbf{x}^{(best)}$  and close to  $\bar{\mathbf{x}}_m$ , the particle velocity will drive  $\hat{\mathbf{x}}^i$  to move more towards  $\bar{\mathbf{x}}_m$ .

To summarise, Covariance Matrix Adaptation Annealing is given in Algorithm 9.

---

**Algorithm 9** Covariance Matrix Adaptation Annealing at time  $t$

---

**Require:** a sequence of  $\alpha_m$  for every phase is defined, previous particles  $\mathbf{x}_{t-1}$ , observation  $\mathbf{y}_t$ , the number of phases  $M$  and the initial covariance matrix  $\mathbf{P}_0$  and  $\mathbf{P}_{0,t}$  are given

**for**  $m = 1$  to  $M$  **do**

- 1: Initialise  $N$  particles  $\mathbf{x}_t^i$  from the previous phase or the temporal model  $p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i)$ .
- 2: Calculate the energy  $E(\mathbf{y}_t, \mathbf{x}_t^i)$  for all particles.
- 3: Find  $\lambda_m$  by solving the equation  $\alpha_m N \sum_{i=1}^N (w_{t,m}^i)^2 = (\sum_{i=1}^N w_{t,m}^i)^2$ .
- 4: Update weights for all particles using the equation  $w_i = p(\mathbf{y}_t | \mathbf{x}_t^i) = \exp\{-\lambda E(\mathbf{y}_t, \mathbf{x}_t^i)\}$ .
- 5: Resample  $N$  particles  $\hat{\mathbf{x}}^i$  from the importance distribution.
- 6: Update the perturbation matrix.

$$\bar{\mathbf{x}} = \sum_{i=1}^N w_i \hat{\mathbf{x}}^i$$

$$\mathbf{p}_c = (1 - c_c) \mathbf{p}_c + c_c (\bar{\mathbf{x}}_m - \bar{\mathbf{x}}_{m-1})$$

$$\mathbf{P}_{m,t} = (1 - c_1 - c_\mu) \mathbf{P}_{m-1,t} + c_1 \mathbf{p}_c \mathbf{p}_c^T + c_\mu \sum_{i=1}^N w_i (\mathbf{x}^i - \bar{\mathbf{x}}_m)(\mathbf{x}^i - \bar{\mathbf{x}}_m)^T$$

- 7: Generate  $N$  random perturbations  $\delta^i$  by Gaussian noise with covariance  $(\prod_{i=1}^m \alpha_i) \mathbf{P}_{m,t}$  and mean  $\mu = 0$ .
- 8: Compute particle velocities imposed by the majority force from  $\bar{\mathbf{x}}_m$  and attraction to the current global best  $\mathbf{x}^{(best)}$ ,  $\mathbf{v}^i = (1 - r_b) \frac{\bar{\mathbf{x}}_m - \hat{\mathbf{x}}^i}{\|\bar{\mathbf{x}}_m - \hat{\mathbf{x}}^i\|_2} + r_b \frac{\mathbf{x}^{(best)} - \hat{\mathbf{x}}^i}{\|\mathbf{x}^{(best)} - \hat{\mathbf{x}}^i\|_2}$ .
- 9: Compute the final  $N$  particles with  $\mathbf{x}^i = \hat{\mathbf{x}}^i + (1 - c_v) \delta^i + c_v \mathbf{v}^i$

**end for**

---

### 5.4.3 Experiments with Benchmark Optimisation Problems

To compare the efficiency of the Covariance Matrix Adaption Annealing algorithm with the three optimisation methods mentioned in this chapter, we conduct experiments against four benchmark optimisation problems in high dimensional space, including some difficult multimodal and non-separable problems. All methods have been allocated equivalent initial conditions. An equal number of evaluations and the average minimum function value together are used to measure how fast the algorithms converge.

#### 5.4.3.1 Ackley Problem

The original Ackley problem [Ackley 1987] was defined for two dimensions, but the problem has been generalised to  $N$  dimensions [Bäck 1996]. Formally, this generalised problem can be described as finding an  $N$  dimensional vector  $\mathbf{x} = \{x_1, x_2, \dots, x_N\}$ , with  $x_i \in (-32.768, 32.768)$ , that minimises the following equation:

$$f(\mathbf{x}) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + \exp(1) \quad (5.4.1)$$

The Ackley problem is multimodal. There are competitive local minima around the global minimum. However, the attraction from the global minimum is still dominant within the domain  $(-32.768, 32.768)$ , as illustrated in Figure 5.3. The global minimum is located at  $\mathbf{x} = \mathbf{0}$  with  $f(\mathbf{x}) = 0$ .

The four methods—Covariance Matrix Adaptation (CMA), Particle Swarm Optimisation (PSO), Simulated Annealing and Covariance Matrix Adaptation Annealing (CMA-Annealing) have been used to optimise the 30 dimensional Ackley function from the same initial position. The average results from 100 executions have been plotted in Figure 5.4. Simulated Annealing has a very slow convergence compared with the other methods. The average minimum value slowly converges from an initial value above 10 to roughly zero during the course of 4000 evaluations. Since Sim-



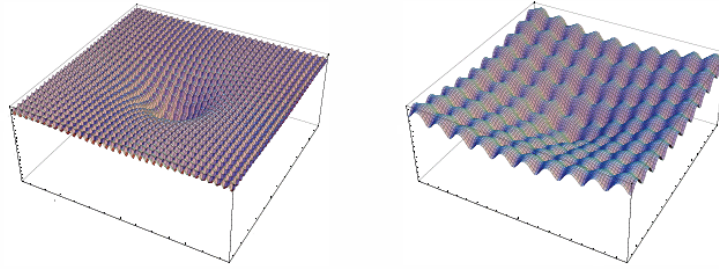


Figure 5.3: The Ackley function at two different zoom ratios. The graph on the left side employs the whole definition area of the function from -30 to 30. The graph on the right side shows the area of the global minimum giving a better impression of the properties of the function.

ulated Annealing uses nearly isotropic Gaussian perturbation for each particle, the overall exploration direction is undetermined or stochastic. Therefore, its convergence speed is slow for a roughly consistent structure like the Ackley function. In comparison, CMA, PSO, and CMA-Annealing have relatively sharper convergence rates. After 1600 evaluations, these three methods have reached the average minimum value close to zero. CMA-Annealing and CMA have very similar performance; the two graphs are effectively intertwined all the way through the end of 4000 evaluations. Due to the consistent overall structure of the Ackley function, PSO has slightly lower performance than the CMA-based methods. This consistent structure favours the Hessian-based approximation used in the CMA-based methods.

#### 5.4.3.2 Rastrigin Problem

The Rastrigin problem, first proposed by Rastrigin [Törn and Zilinskas 1989] as a 2-dimensional function, has been generalised by Mühlenbein et al in [Mühlenbein et al. 1991]. This is a fairly difficult multimodal problem due to its large search space and large number of local minima. The highly multimodal surface of the function is generated by the cosine modulation variables  $A$  and  $\omega$ , which control the amplitude and frequency modulation respectively. It is usually recommended to set  $A = 10$ , and  $\omega = 2\pi$ . The problem is defined as finding an  $N$  dimensional vector  $\mathbf{x}$ , where

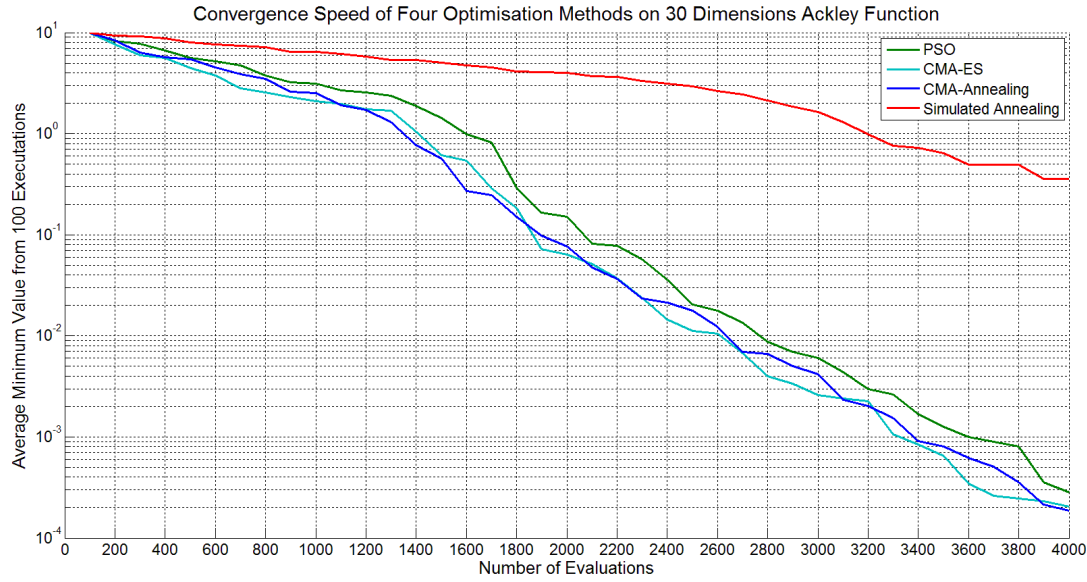


Figure 5.4: Four different methods are applied to Ackley function. Simulated Annealing shows slower convergence than the other methods. The CMA-based methods (CMA and CMA-Annealing) have better convergence than PSO due to the fact that Ackley function has roughly consistent structure.

$x_i \in [-5.12, 5.12]$ , which minimises the equation below:

$$f(\mathbf{x}) = AN + \sum_{i=1}^N x_i^2 - A \cos(\omega x_i) \quad (5.4.2)$$

The Rastrigin function has a complexity of  $O(N \ln(N))$ . Its global minimum is also located at  $\mathbf{x} = \mathbf{0}$ . The two dimensional Rastrigin function is plotted in Figure 5.5.

The 30 dimensional Rastrigin function (Figure 5.6) is difficult to optimise using the four methods mentioned. All of them with 4000 evaluations failed to converge close to the global minimum. The major reason is the initial position is far from the global minimum and local minima easily trap the algorithms. The perturbation range is scaled down before the samples can find the major basin.

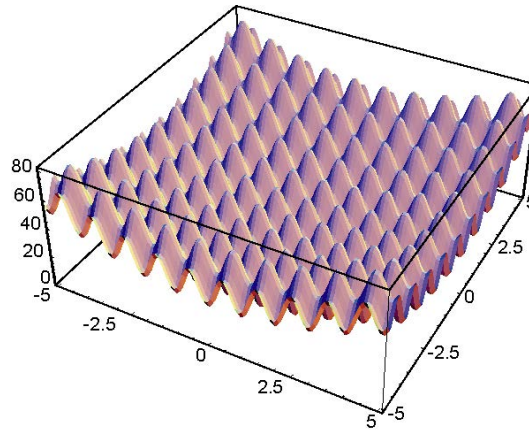


Figure 5.5: The two dimensional Rastrigin function. It has a cosine modulation to produce many local minima. However, the locations of the minima on the highly multimodal surface are regularly distributed.

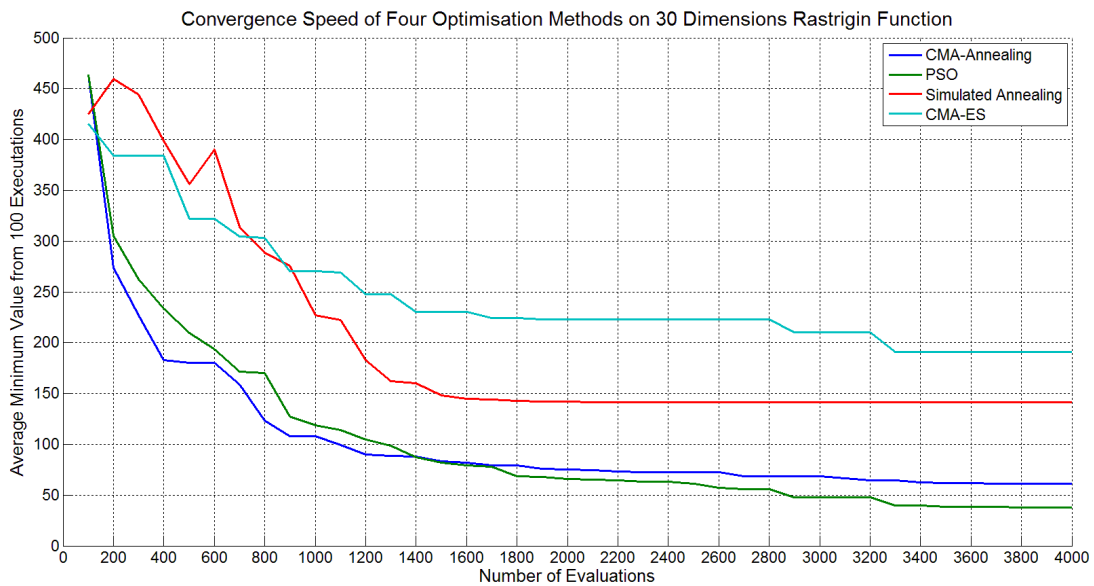


Figure 5.6: Algorithm comparison with 4000 evaluations on the 30 dimensional Rastrigin function show that the four different methods fail to converge close to the global minimum. This is due to the large search space and number of local minima with respect to the limited number of evaluations.

### 5.4.3.3 Griewank Problem

The Griewank problem [Griewank 1981] is similar to the Rastrigin problem. It has many widespread local minima that are regularly distributed. The one-dimension Griewank function has 191 local minima; as its dimension increases, the Griewank problem has an exponentially increasing number of local minima. To find the global minimum is a very challenging task, especially in high dimensional space. The problem can be described as finding an  $N$  dimensional vector  $\mathbf{x}$ , with  $x_i \in (-600, 600)$ , that minimises the following equation:

$$f(\mathbf{x}) = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^N \cos\left(\frac{x_i}{\sqrt{i}}\right) \quad (5.4.3)$$

Its global minimum is located at  $\mathbf{x} = \mathbf{0}$  with  $f(\mathbf{x}) = 0$ . Figure 5.7 shows the 2D Griewank function.

The experimental results on the 30 dimensional Griewank function with 4000 evaluations using the four different methods are shown in Figure 5.8. Although the Griewank problem has an exponentially increasing number of local minima with respect to dimensionality, it still has a roughly consistent local structure, similar to the Ackley function. CMA based methods are able to converge very quickly. In particular, CMA-Annealing performs better than the original CMA, reaching  $10^{-7}$  accuracy compared with  $10^{-6}$  for CMA after 4000 evaluations. PSO has an encouraging convergence rate up to 1800 evaluations, where it converges more quickly than CMA. However, it becomes trapped in a local minimum after 1800 evaluations. Simulated Annealing has the slowest convergence speed, but it obtains a better average minimum value than PSO after 3600 evaluations. CMA-Annealing demonstrates appealing performance overall, achieving reasonable accuracy after only 1400 evaluations.

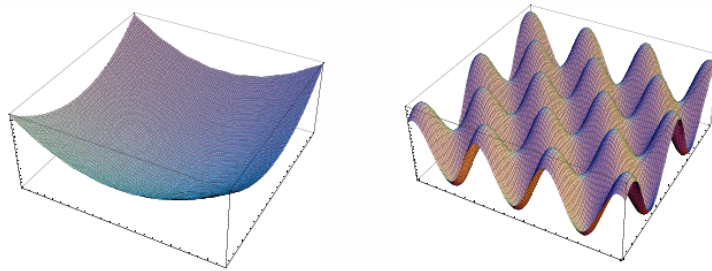


Figure 5.7: The interpretation of the Griewank function changes with scale. A general overview suggests a convex function; a medium-scale view suggests the existence of local extrema, and the fine details indicate a complex structure with numerous local extrema.

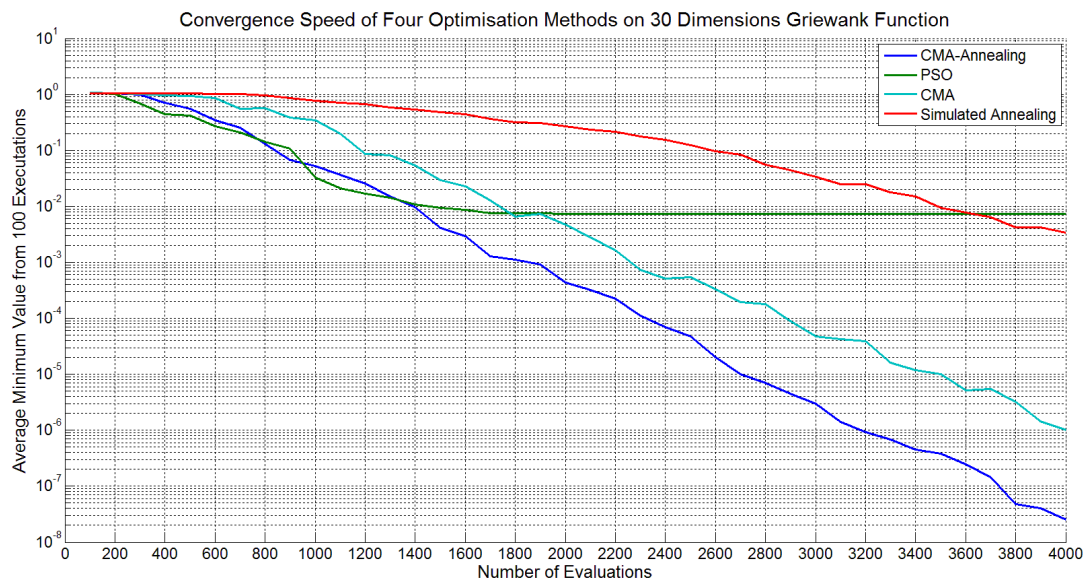


Figure 5.8: Algorithm comparison with 4000 evaluations on the 30 dimensional Griewank function shows CMA-Annealing achieves the fastest convergence, followed by PSO and CMA. Although Simulated Annealing converges slower than the other methods, it avoids being trapped in local minima unlike PSO.

#### 5.4.3.4 Rosenbrock Problem

The Rosenbrock problem [Rosenbrock 1960] is a classic optimisation problem, also known as the banana function. It is non-separable problem. The global optimum lays inside a long, narrow, relatively flat parabolic valley. To find the valley is trivial, however convergence to the global optimum is difficult. The  $N$  dimensional problem can be defined as finding a vector  $\mathbf{x}$  that minimises the equation:

$$f(\mathbf{x}) = \sum_{i=1}^{N-1} \left( 100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2 \right) \quad (5.4.4)$$

Many researchers take the high-dimensional Rosenbrock function as a unimodal function by instinct. However, the Rosenbrock function has been shown [Shang and Qiu 2006] to have exactly one minimum for  $N = 3$  (at  $(1, 1, 1)$ ) and exactly two minima for  $4 \leq N \leq 30$ —the global minimum of all ones and a local minimum nearby. Figure 5.9 illustrates the Rosenbrock valley in two dimensions.

The four methods have the expected result of optimising the 30 dimensional Rosenbrock function as shown in Figure 5.10. Simulated Annealing converges relatively slowly, but it is more capable of escaping from the attraction of local minima. The other three methods have better convergence rates, however, they are trapped after 2000 evaluations. While CMA-Anealing has better convergence than CMA and PSO, its improvements in accuracy after 2000 evaluations are minor. It is difficult for the CMA based method to move along the Rosenbrock valley. Thus it is easy for CMA-Annealing to be trapped in local minima.

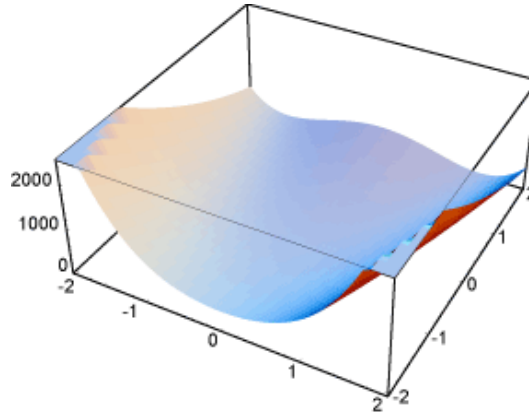


Figure 5.9: The two dimensional Rosenbrock function. Although it is a unimodal problem, finding the global minimum is very difficult. For instance, if an optimisation method starts at the initial point located at  $(-1.2, 1)$ , it has to find its way to the other side of a flat, curved valley to find the optimal point.

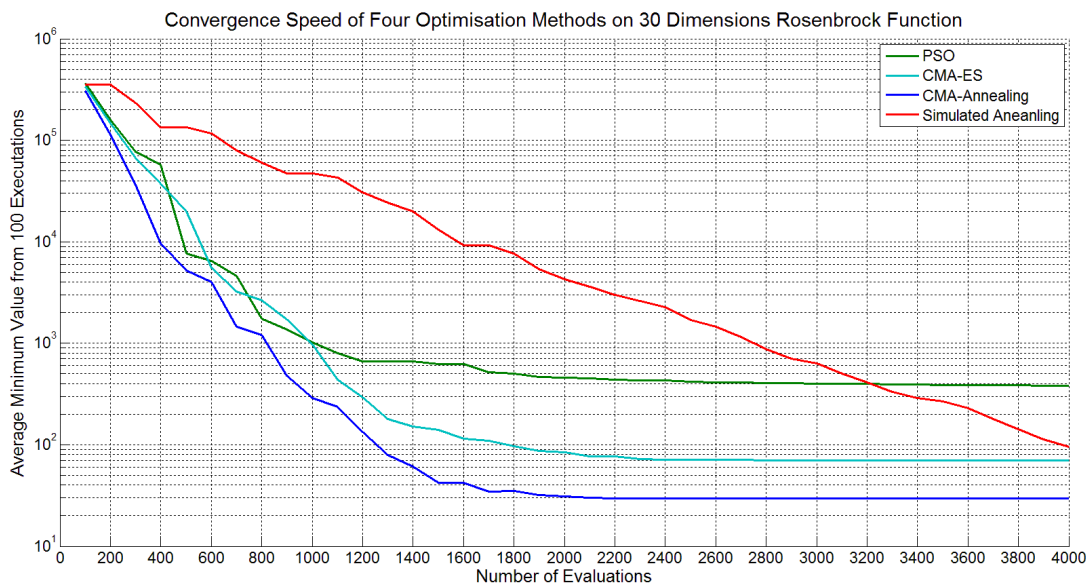


Figure 5.10: With the exception of Simulated Annealing, the methods become trapped in local minima. However, Simulated Annealing has the slowest convergence rate. Overall, none of the four methods are able to converge reasonably close to the global minimum after 4000 evaluations.





## Robust Evaluation Model

---

A general failure of tracking using the Annealing Particle Filter like optimisation method may indicate two possibilities: 1) The number of particles or layers is inadequate to cover large enough search space to guarantee finding reasonably good solutions. This often requires increasing the computational time or effectively reducing the computational complexity to expand coverage in search space. 2) The optimisation process is converged to the global minimum, but it does not correspond to the true gesture. This is the consequence of the evaluation being modelled inaccurately. For instance, the converged estimate may correspond to the global minimum of corrupted data rather than the true posture. In this chapter, we propose several improvements on the evaluation model. Incremental Relaxation by Fast March Method aims to avoid premature convergence; Colour and Texture Incorporation attempts to introduce extra information to resolve ambiguities and improve robustness to light variations. Maximisation of Mutual Information and Gradual Sampling together allow the evaluations to concentrate on large errors (caused by misalignments of the tracking subject), and therefore be robust to other noise errors as well as being computationally efficient.

## 6.1 Incremental Relaxation by Fast March Method

The silhouette<sup>1</sup> is often used in markerless human motion capture to describe the shape of the human body. However, when lacking colour and texture information, the silhouette essentially describes an image as no more than a contour line which only contains partial information from the original image. Shape ambiguities can occur along the depth direction. This can result in multiple solutions when one attempts to fit the original human body to the visual hull computed by the shape-from-silhouette technique [Laurentini 1994]. The parameterised human pose usually resides in high dimensional space, and it turns out that the solution of human motion capture is subject to non-convex and multi-modal optimisation in high dimensional space.

Many different approaches have attempted to breach this non-convex and multi-modal high dimensional problem. One of these ideas that is particularly intriguing, Graduated Optimisation, assumes that we can convert the original problem to a sequence of designed problems which are ordered from simple to complex (equivalent to the original problem). Since those solutions can be progressively obtained and used as the initialisation for each successive problem, solving this sequence of problems is much simplified compared with solving the original problem. Take the example of a continuous multi-modal problem shown in Figure 6.1. Here the original optimisation problem is transformed into a sequence of optimisation problems, such that the first problem in the sequence is convex (or nearly convex), the solution to each problem gives a good starting point for the next problem in the sequence, and the last problem in the sequence is the difficult optimisation problem that it ultimately seeks to solve. If each problem in the sequence is locally convex around the optimal value and the solutions are good enough inside the local convex region, then, it can be guaranteed that the optimal solution to the final problem in the sequence will be found.

Simulated Annealing [Kirkpatrick et al. 1983] has similar behaviour when the

---

<sup>1</sup>The silhouette is more robust to illumination variation and easier to match than colour and texture features.

---

annealling variable is gradually evaluated, causing the energy function to be transformed from a single peak to the multiple peaks in the landscape of the global optimum. In Chapter 7 of [Blake and Zisserman 1987], the authors introduce the Graduated Non-Convexity Algorithm. The algorithm first constructs a convex approximation to the non-convex energy function, and then proceeds to find its minimum. In order to achieve the objective of a convex approximation, several constraints on line process interactions in the form of penalties levied on broken contours, etc., are added. The energy function becomes a function of the penalty intensities and a control parameter. In successive steps a sequence of energy functions generated by varying this control parameter are minimized, starting from the initial convex stage. Such a procedure is certainly intuitively appealing. It is unlikely that very general statements can be made about its effectiveness for an arbitrary non-convex cost function. But in the case of the energy functions that describe the weak string and membrane models<sup>2</sup>, the algorithm can be shown to be correct for significant class of signals [Blake and Zisserman 1987].

An incremental method introduced in this section shares a similar spirit, but it works on the data domain. Although this approach is designed for our particular problem, its insight is generally applicable to any contexts and applications. The coarse-to-fine operation is performed incrementally on the data. The coarse and fine data correspond to a simple energy function (ideally with roughly convex shape) and a complex energy function (the original energy function). The proposed method seamlessly incorporates a control parameter for an isotropic distance map to APF, allowing incremental data relaxation to work consistently with the annealling schedule.

The basic idea is that the fitness test criteria are relaxed to allow a large number of particles to survive at an early stage and encourage broad exploration. The complete fitness test is delayed until adequate information is gathered. This method enables

---

<sup>2</sup>The weak string, however, preserves discontinuities without any prior information about their existence or location.

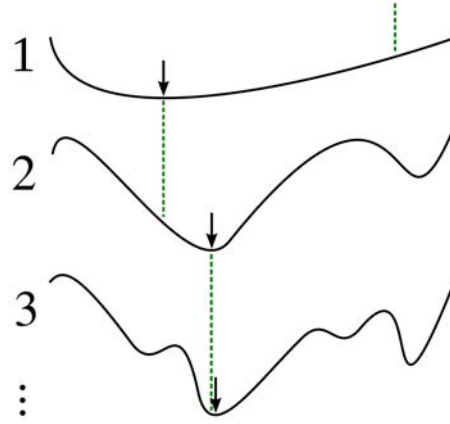


Figure 6.1: Graduated Optimisation for a continuous multi-modal problem: the multi-modal problem is reshaped and simplified as a sequence of easier subproblems. The solution of the current problem provides a good initial position to the successive problem, which simplifies optimisation dramatically. (courtesy of [Wikipedia 2011])

a better chance than APF to escape the attraction of local minima and converge to the global minimum. As shown in Figure 6.2, the contour of the distance map is relaxed (thickened), allowing a larger number of particles to survive and explore a larger search space. As it proceeds, the contour is gradually contracted to approach the original silhouette shape. Probabilistically, it selects better fit particles, and gradually concentrates them on the region which contains the global minimum.

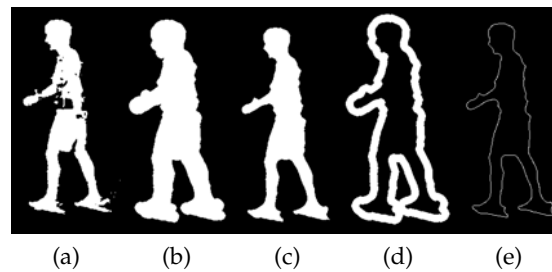


Figure 6.2: Level Set and Silhouette Images: (a) the original silhouette, (b) outwards relaxed silhouette, (c) outwards and inwards fixed silhouette, (d) outwards/inwards relaxed contour, and (e) the original contour

On the basis of this idea, an extra term  $D_c(rel(\mathbf{y}_t, m), \mathbf{x}_t)$  is introduced for the purpose of relaxing the fitness test criteria. The energy function  $E(\mathbf{y}_t, \mathbf{x}_t)$  is then given by:

$$\frac{1}{N_v} \sum_{i=1}^{N_v} D_s(\mathbf{y}_t, \mathbf{x}_t) + \alpha D_c(\text{rel}(\mathbf{y}_t, m), \mathbf{x}_t) \quad (6.1.1)$$

where  $N_v$  is the number of views.  $D_s(\mathbf{y}_t, \mathbf{x}_t)$  measures differences between the observed silhouette  $\mathbf{y}_t$  and the silhouette generated by the particle  $\mathbf{x}_t$ .  $D_c(\text{rel}(\mathbf{y}_t, m), \mathbf{x}_t)$  measures how well the contour (generated by the particle  $\mathbf{x}_t$ ) fits the relaxed contour of the isotropic distance map.  $\alpha$  is a factor<sup>3</sup> to adjust the influence of the relaxing term on the energy function.  $m$  controls the degree of relaxation. As the annealing schedule proceeds, the contour of the isotropic distance map is changed from the thick curve (corresponding to the loosest criteria) at the beginning to the original thin curve (corresponding to the original criteria) at the end shown in Figures 6.2d and 6.2e, respectively. The relaxing operation  $\text{rel}(\mathbf{y}_t, m)$  is computed by Fast March Method, whose details are described in the next section. Furthermore, noisy silhouette images can be smoothed by marching Fast March Method outwards and then inwards as shown in Figures 6.2a, 6.2b and 6.2c, respectively. Overall, the relaxed APF algorithm in a typical annealing phase is outlined in Algorithm 10.

### 6.1.1 Fast March Method

The Fast March Method [Sethian 1999] is a technique for tracking the evolution of an expanding front. In this context, a front is a closed surface in 3D (or a closed curve in 2D) which separates an interior and an exterior region. The Fast March Method is simply a technique for computing the arrival time of a front at the points of a discrete lattice. If the front is simply a closed curve in 2D, and the lattice is a pixel raster, the Fast March Method assigns to each pixel the time at which the expanding curve hits the pixel. The method applies only to cases where the front is uniformly expanding since the arrival time of the front is uniquely defined only in these cases. The front evolves by motion in the normal direction of the curve. The speed does not have to be

<sup>3</sup>It can be learned from the cross-validation method at the training stage.

the same everywhere, but the speed must always be non-negative. At a given point, the motion of the front is described by the equation known as the Eikonal equation:

$$\|\nabla T(\mathbf{x})\|F(\mathbf{x}) = 1$$

where  $T$  is the arrival time of the front at point  $\mathbf{x}$  and  $F \geq 0$  is the speed of the front at point  $\mathbf{x}$ . Because the front can only expand, the arrival time  $T$  is single valued. Although the Fast March Method is more general, for simplicity, we will restrict our attention to 2D lattices of the usual sort, e.g. isotropic, rectangular 2D lattices. We will generally assume that  $F = 1$  everywhere. In this case, the fast march method simply propagates the shortest distance to the boundary to all other points in the lattice.

---

**Algorithm 10** Relaxed Annealed Particle Filter

---

Initialise  $N$  particles  $\mathbf{x}_t^i$  from the previous phase or the temporal model  $p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i, \mathbf{y}_t^i)$ .  
 Calculate the energy  $E(\mathbf{y}_t, \mathbf{x}_t)$  for all particles using equation (6.1.1).  
 Set an appropriate  $\alpha_m$ , find  $\lambda_m$  by solving the equation  $\alpha_m N \sum_{i=1}^N (w_{t,m}^i)^2 = (\sum_{i=1}^N w_{t,m}^i)^2$   
 Update particle weights using the equation  $p(\mathbf{y}_t | \mathbf{x}_t) = \exp\{-\lambda E(\mathbf{y}_t, \mathbf{x}_t)\}$   
 Resample  $N$  particles from the importance distribution.  
 Perturb particles by Gaussian noise with covariance  $\mathbf{P}_m = \alpha_m \mathbf{P}_{m-1}$  and mean 0.

---

The Fast March Method works outwards from an initial condition. To define the initial condition, a set of pixels in the image are labelled as frozen, and we compute distances to their neighbours. The vertices that have computed distances but are not yet frozen are said to be narrow band pixels. For each iteration of the central loop of the algorithm, the narrow band pixel having the smallest distance value is frozen, and distances are computed from its neighbours. Frozen pixels are used to compute the values of other pixels but are never computed again. Thus we can see the method itself as a front of narrow band pixels that propagates from the initial condition, freezing pixels as it moves along.

Distances are computed by solving the Eikonal equation. In this way we find a dis-

---

**Algorithm 11** Fast March Method

---

**Require:** A list of pixels  $L$  containing pixels whose distances are known and thus form the initial condition, and a binary heap  $H$  that is initially empty.

```

for each pixel  $p$  in  $L$  do
  Freeze  $p$ 
  for each neighbour  $p_n$  of  $p$  do
    Compute distance  $d$  at  $p_n$ 
    if  $p_n$  is not in narrow band then
      Label  $p_n$  as narrow band
      Insert  $(d, p_n)$  into  $H$ 
    else
      Decrease key of  $p_n$  in  $H$  to  $d$ 
    end if
  end for
end for
while  $H$  is not empty do
  Extract  $p$  from top of  $H$ 
  Freeze  $p$ 
  for each neighbour  $p_n$  of  $p$  do
    if  $p_n$  is not frozen then
      Compute distance  $d$  at  $p_n$ 
      if  $p_n$  is not in narrow band then
        Label  $p_n$  as narrow band
        Insert  $(d, p_n)$  into  $H$ 
      else
        Decrease key of  $p_n$  in  $H$  to  $d$ 
      end if
    end if
  end for
end while

```

---

tance value for the narrow band pixel so that the estimated magnitude of the gradient  $\|\nabla T\|$  is equal to  $= 1/F$ . Sethian [Sethian 1999] suggested a gradient approximation borrowed from the field of hyperbolic conservation laws:

$$\max(d_{ij}^{-x}T, -d_{ij}^{+x}T, 0)^2 + \max(d_{ij}^{-y}T, -d_{ij}^{+y}T, 0)^2 = 1/F_{ij}^2 \quad (6.1.2)$$

where  $d_{ij}^{-x}$  stands for the difference between the distances of the points  $(i, j)$  and  $(i - 1, j)$ . The Fast March Method is summarised in Algorithm 11. Figure 6.3 shows the results from the Fast March Method applied to the human hand contour. Considering in reverse order, the shape details of the contour are transformed from blurred to highly detailed. This is in agreement with the coarse-to-fine strategy in the anneal schedule.

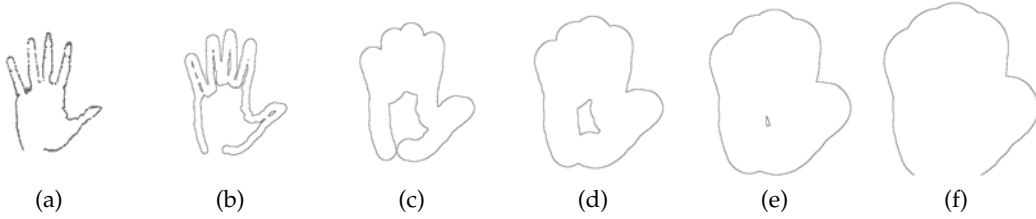


Figure 6.3: Progressive Results of the Fast March Method on Hand Contour

### 6.1.2 Experiments

The dataset from [Balan et al. 2005], which contains multi-view images, calibrated cameras and ground truth data from motion capture, was used. The first experiment compared the accuracy of the proposed method (relaxed AFP) with standard AFP using 200 particles and a 10-phase schedule. The results in Figure 1 show the average joint angle error (absolute values over 28 joint angles) and the position error (measured in Euclidean distance). Despite the fact that fewer particles were used in this experiment than in [Deutscher and Reid 2005], the outcome of relaxed AFP still appears encouraging. However, we make the important observation that the global



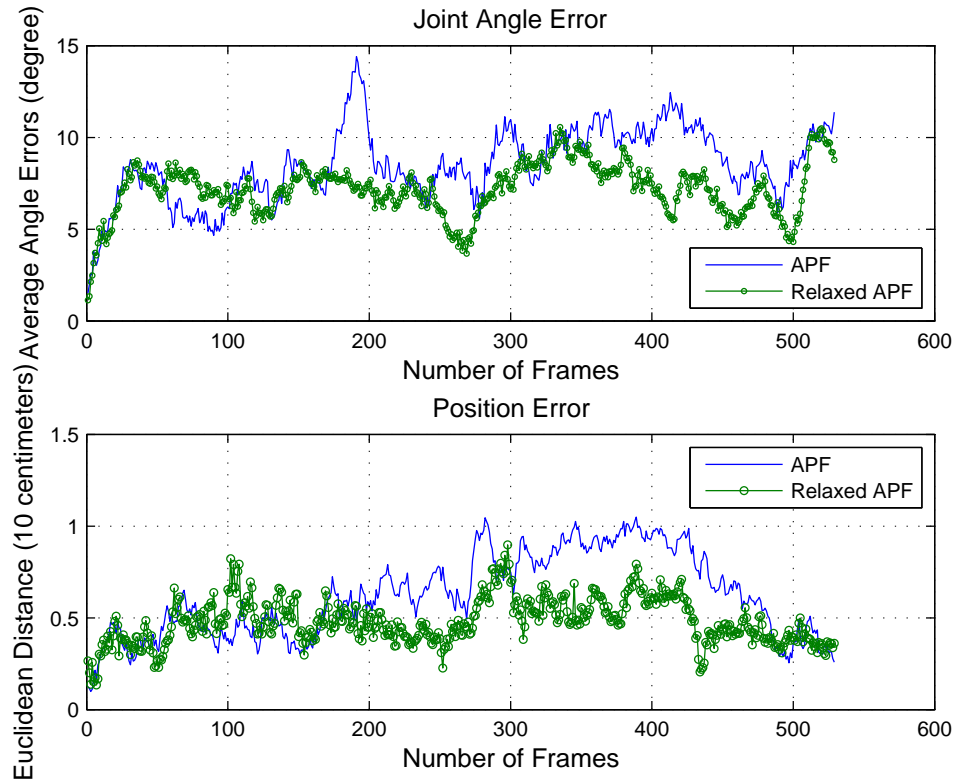


Figure 6.4: Comparison of APF and Relaxed APF: Relaxed APF has slightly better performance in terms of the joint angle errors, showing less than 10 degrees on average. The torso position errors for Relaxed APF appear more stable at about 5cm on average after frame 200, whereas APF suffers from the relatively greater fluctuation.

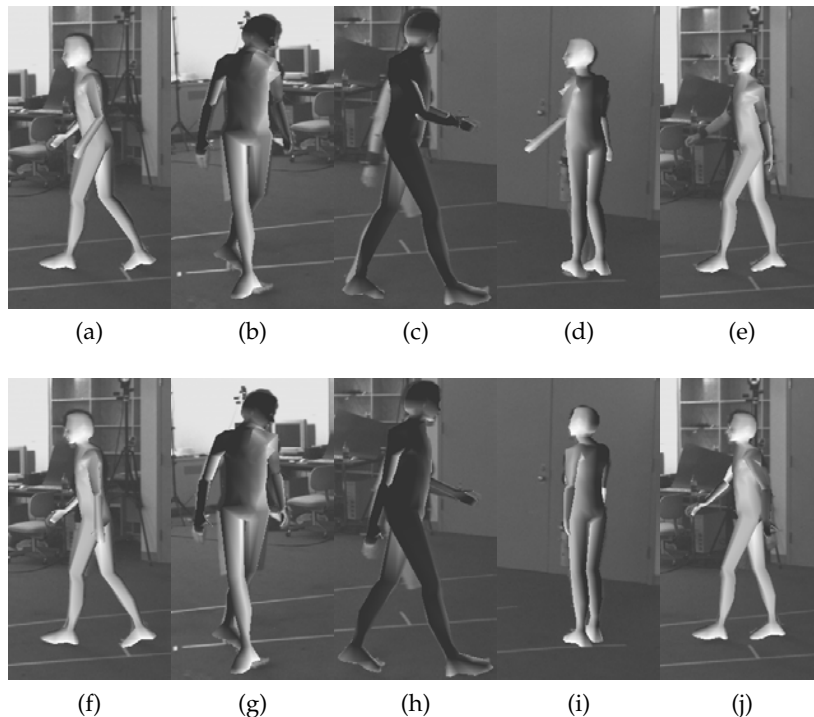


Figure 6.5: Figures (c), (d) and (e) show mistracking of the arm occurs in APF after 300 frames. In contrast, Relaxed APF still maintains accurate tracking results.

minimum of the energy function is often a mismatch for the true pose since the silhouette images are contaminated by noise. In this case, even though an optimisation method is able to find the global minimum, it may still miss the true pose and introduce a large error. The second experiment was conducted by using a different human body model which has a higher resemblance between the body shape and the real performer. The outcomes of APF and relaxed APF are listed in the second column and the third column of Figure 6.5, respectively. Figures 6.5c, 6.5d and 6.5e show that the arm of the performer is sometimes mistracked in APF after self-occlusions appear in certain views. In contrast, relaxed APF can still perform normal tracking.

## 6.2 Colour and Texture Incorporation

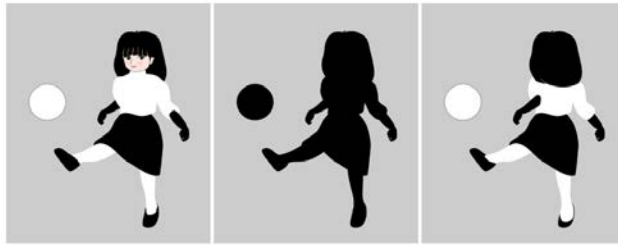


Figure 6.6: Silhouette Ambiguities (courtesy of [Kitaoka 2007])

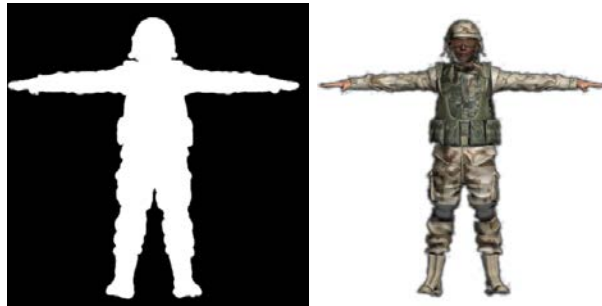


Figure 6.7: Texture and colour encode a considerable amount of information due to the absorption and scattering properties of the surface and material being different from that of the incoming wavelengths of light that illuminate it.

There is a simple phenomenon whereby the human eye can not infer the posture of a human body from one single silhouette image, but can easily recognise the posture from one colour image. This is related to the Silhouette Illusion [Kitaoka 2007].

---

Some examples of ambiguities are illustrated in Figure 6.6 and 6.7. In fact, a silhouette is a less informative, weak descriptor which is very likely to lead to ambiguities. In contrast, a textured silhouette is a much richer and stronger descriptor. Colour and texture is a consequence of the different configuration of atoms and electrons that objects that have, and of the incoming light. These differences eventually cause reflection, scattering, absorption, transmission and refraction of the light. Colour and texture encodes very rich information, including information directly related to what we are interested in, for instance, the distance between the object and viewer, and normals on the surface. Current ray tracing techniques are already able to simulate realistic light travel and generate high-colour and fine-textured models such as seen in Figure 6.7.

To recover distances and normals from colour and texture is sophisticated and beyond of this work. Nevertheless, colour and texture contains much richer information than silhouettes. Incorporating colour and texture information into the evaluation model undoubtedly improves robustness and distinction. Incorporating colour and texture information for markerless motion capture requires a surface model of the human body that describes the colour and texture information of clothes or skin. Real deformation of this surface model is quite a complex process because it involves various factors (e.g. gravity, bones' transformation, skin, clothes). Excellent works from Starck et al's [Starck and Hilton 2007b; Starck and Hilton 2006; Starck et al. 2006; Starck and Hilton 2005a; Starck et al. 2005; Starck and Hilton 2005b; Starck and Hilton 2003a] and Theobalt et al's [Ahmed et al. 2008; Carranza et al. 2003; Ahmed et al. 2007; Theobalt et al. 2005; Eisemann et al. 2008; de Aguiar et al. 2008; de Aguiar et al. 2007a] studies have demonstrated that texture incorporation is able to improve tracking accuracy. Compared with their works, the method proposed below is rather lightweight and can be efficiently implemented. To simplify the problem, we adopt a mature technology of texture mapping from the computer graphics community. We emphasise how to generate and incorporate a mesh texture of the body model in markerless mo-

tion capture.

Silhouettes and the body meshes can be roughly registered to input multiview images (the method described in Section 4.4 is one of possible solutions). We can incorporate the appearance of material by parameterising a surface mesh (termed texture mapping), and blending multiview images into the mesh texture. Texture mapping is a well-studied problem in computer graphics. In general, it is often represented as the mapping from 3D vertex coordinates on a mesh to 2D  $uv$  texture coordinates in a texture image. Borrowing terminology from mathematics, this is often referred to as creating an atlas of charts for a given surface. Modern graphics hardware is capable of rendering a highly realistic model with a large mesh texture in real time. The automatic generation of high quality texture parameterisations has become a widely supported functionality in many graphics software packages. We use Blender [Blender 2010] to generate patch-based texture parameterisation for the body mesh.

After creating the patch-based texture parameterisation for the body mesh, we re-sample the texture mesh from the input images that have been registered with the mesh. We show how to blend multiview images using the body model. First, we perform a vertex-to-image binding for all vertices of the body mesh. Each mesh vertex  $v$  is assigned a set of valid images, which is defined as the subset of the input images where  $v$  is visible in each image and  $v$  is a non-silhouette vertex. A vertex  $v$  is visible in an image, if the projection of  $v$  on the image plane is contained in the image, the normal vector of  $v$  is directed towards the viewpoint and there are no other intersections of the face mesh with the line that connects  $v$  and the viewpoint. A vertex  $v$  is called a silhouette vertex, if at least one of the triangles in the triangle fan around  $v$  are oriented opposite to the viewpoint. Note, the set of valid images for a vertex may be empty.

Theoretically, this is enough for classifying vertices. However, there may be some error because of registration or numerical errors, especially in the neighbourhood of

silhouettes. Some of the vertices can be bound to background pixels of the input images. Such vertices should be classified as unbound vertices. We detect this registration error by comparing the colour value with the background colour of the input image. First we calculate the projection coordinate of a vertex to its bound image. Then, we sample the pixel value and calculate the distance between this pixel value and the background colour. To avoid mis-detection, we reduce the noise in the input images by applying a normal median filter and do not just compare with a single sample pixel, but perform Gaussian convolution with a 3x3 subimage mask. If the distance is larger than a given threshold value, then the vertex is re-classified as an unbound vertex.

Let  $\triangle = \{v_1, v_2, v_3\}$  denote a triangle of the body mesh and  $\tilde{\triangle} = \{\tilde{v}_1, \tilde{v}_2, \tilde{v}_3\}$  be the corresponding triangle in the texture mesh. For each triangle  $\triangle$ , exactly one of the following situations might occur:

1. There exists at least one common image in the sets of valid images of the three vertices  $\{v_1, v_2, v_3\}$  of  $\triangle$ .
2. All of the vertices of  $\triangle$  are bound to at least one image, but no common image can be found for all three vertices.
3. At least one vertex of  $\triangle$  is not bound to any image.

In the first case, we rasterise  $\tilde{\triangle}$  in texture space. For each texel  $T$ , we determine its barycentric coordinates  $\rho, \sigma, \tau$  with respect to  $\tilde{\triangle}$  and compute the corresponding normal  $N$  by interpolating the vertex normals of  $\triangle$ :  $N = \rho N(v_1) + \sigma N(v_2) + \tau N(v_3)$ . For each common image  $i$  in the sets of valid images of all vertices of  $\triangle$ , we compute the dot product between  $N$  and the viewing direction  $V_i$  for the pixel  $P_i$  that corresponds to  $T$ . Finally, we colour  $T$  with the colour obtained by the weighted sum of pixel colours  $\sum_i \langle N, V_i \rangle \text{Colour}(P_i) / \sum_i \langle N, V_i \rangle$ .

In the second case, we colour each vertex  $\tilde{v}_j$  of  $\tilde{\triangle}$  individually by summing up the

weighted pixel colours of the corresponding pixels in all valid images  $i$  of  $\tilde{v}_j$  similar to the first case:  $Colour(\tilde{v}_j) = \sum_i \langle N(v_j), V_i \rangle Colour(P_i) / \sum_i \langle N(v_j), V_i \rangle$ . The texels of the rasterisation of  $\tilde{\Delta}$  are then coloured by barycentric interpolation of the colours of the vertices  $\tilde{v}_1, \tilde{v}_2, \tilde{v}_3$ . Alternatively, we tried to use as much information as possible from the input images if, for instance, the vertices  $v_1, v_2$  of  $\Delta$  share an image and the vertices  $v_2, v_3$  share another image. However, this situation always happens near the silhouette of an object and the extrapolation of a missing vertex on the image will be unstable. Colour interpolation from reliable vertices is used to cope with this issue, producing reasonable results.

Since the set of valid images for a vertex may be empty, there might exist some vertices that cannot be coloured by any of the previously described schemes. We address this problem in a two-stage process: First, we iteratively assign an interpolated colour to each unbound vertex. Next, we perform the colour interpolation scheme from the second case for the remaining triangles of  $\Delta$  that have not yet been coloured. The first step iteratively loops over all unbound and uncoloured vertices of the face mesh. For each unbound vertex  $v$ , we check if at least  $p = 80\%$  of the vertices in the one-ring around  $v$  are coloured (either by being bound to an image or by having an interpolated colour). If this is true, we assign to  $v$  the average colour of all the coloured vertices around  $v$ , otherwise we continue with the next unbound vertex. We repeat this procedure until there are no further vertex updates. Next, we start the same procedure again, but this time we only require  $p = 60\%$  of the vertices in the one-ring around  $v$  to be coloured. As soon as there are no more updates, we repeat this step twice again with  $p = 40\%$  and  $p = 20\%$ . Finally, we update each unbound vertex that has at least one coloured neighbour. Upon termination of this last step, all vertices of the face mesh are either bound or coloured and the remaining triangles of  $\Delta$  can be coloured.

This colour interpolation method is fast and easy to implement, and it can fill all

missing pixels. But the texture details can not be reconstructed by this scheme. More sophisticated pixel filling methods are able to more accurately reconstruct the texture, for example, image inpainting [Bertalmío et al. 2000] and texture synthesis [Igehy and Pereira 1997]. The output model is imported in graphics software for minor manual refinement. Nevertheless, the texture quality reconstructed by our method is good enough and suitable for tracking purposes.



Figure 6.8: Reconstructed mesh texture for three different subjects

### 6.2.1 Illumination Invariant Colour Difference

It is easy for the human eye to pick up differences in the colour of clothing, because the colour of clothing conforms comfortably to human perception. The CIELab colour space is ideally designed to approximate perceptual uniformity of human perception

and capture colour differences uniformly. So, the difference computed between two arbitrary colour values is consistent with the differences perceived by the human eye. Intuitively, the likelihood evaluation involving the colour of clothing should be performed in the CIELab colour space. However, it turns out that perceptual colour uniformity in the CIELab colour space still has room to improve. The numerical values of the CIELab colour heavily depend on both the region of colour space and the direction of the colour differences. Several standards have been designed to address this issue. The CIE94 colour difference formula [McDonald and Smith 1995] proposes weight coefficients to an ellipsoid equation for compensating visual tolerances in lightness  $L$ , chroma  $C$  and hue  $h$  differences so that there is always a uniform single-number tolerance, regardless of the colour centre and direction of differences from it.

$$\begin{aligned}
 C &= \sqrt{a^2 + b^2} \quad h = \arctan \frac{b}{a} \\
 \Delta E_{94} &= \sqrt{\left(\frac{L_2 - L_1}{K_L}\right)^2 + \left(\frac{C_2 - C_1}{1 + K_1 C_1}\right)^2 + \left(\frac{h_2 - h_1}{1 + K_2 C_1}\right)^2} \quad (6.2.1)
 \end{aligned}$$

where  $K_L = 1$ ,  $K_1 = 0.045$  and  $K_2 = 0.015$  for graphic arts. More recently, the latest proposal of CIEDE2000 [Luo et al. 2001] includes not only lightness, chroma, and hue weighting functions, but also an interactive term between chroma and hue differences for improving the performance for blue colours, and a scaling factor for the CIELab  $a$  scale for improving the performance for gray colours. It can be given by:



$$\begin{aligned}
\Delta E_{00}^* &= \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2} + R_T \frac{\Delta C'}{k_C S_C} \frac{\Delta H'}{k_H S_H} \\
\Delta L' &= L_2 - L_1 \quad \bar{L} = \frac{L_1 + L_2}{2} \quad \bar{C} = \frac{C_1 + C_2}{2} \\
a'_1 &= a_1 + \frac{a_1}{2} \left(1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\right) \quad a'_2 = a_2 + \frac{a_2}{2} \left(1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\right) \\
\bar{C}' &= \frac{C'_1 + C'_2}{2} \text{ and } \Delta C' = C'_1 - C'_2 \quad \text{where } C'_1 = \sqrt{a_1'^2 + b_1'^2} \quad C'_2 = \sqrt{a_2'^2 + b_2'^2} \\
h'_1 &= \tan^{-1}(b_1/a'_1) \pmod{2\pi}, \quad h'_2 = \tan^{-1}(b_2/a'_2) \pmod{2\pi} \\
\Delta h' &= \begin{cases} h'_2 - h'_1 & |h'_1 - h'_2| \leq \pi \\ h'_2 - h'_1 + 2\pi & |h'_1 - h'_2| > \pi, h'_2 \leq h'_1 \\ h'_2 - h'_1 - 2\pi & |h'_1 - h'_2| > \pi, h'_2 > h'_1 \end{cases} \\
\Delta \bar{H}' &= 2\sqrt{C'_1 C'_2} \sin(\Delta h'/2), \quad \bar{H}' = \begin{cases} (h'_1 + h'_2 + 2\pi)/2 & |h'_1 - h'_2| > \pi \\ (h'_1 + h'_2)/2 & |h'_1 - h'_2| \leq \pi \end{cases} \\
T &= 1 - 0.17 \cos(\bar{H}' - \pi/6) + 0.24 \cos(2\bar{H}') + 0.32 \cos(3\bar{H}' + \pi/30) - 0.20 \cos(4\bar{H}' - 21\pi/60) \\
S_L &= 1 + \frac{1 + 0.015 (\bar{L} - 50)^2}{\sqrt{20 + (\bar{L} - 50)^2}} \quad S_C = 1 + 0.045 \bar{C}' \quad S_H = 1 + 0.015 \bar{C}' T \\
R_T &= -2\sqrt{\frac{\bar{C}'^7}{\bar{C}'^7 + 25^7}} \sin \left[ \frac{\pi}{6} \exp \left( - \left[ \frac{\bar{H}' - 275^\circ}{25} \right]^2 \right) \right]
\end{aligned}$$

where  $R_T$  denotes a hue rotation term to deal with the problematic blue region (hue angles in the neighbourhood of 275),  $R_T \frac{\Delta C'}{S_C} \frac{\Delta H'}{S_H}$  is the compensation for neutral colours (the primed values in the  $L C h$  differences), and  $S_L$ ,  $S_C$  and  $S_H$  are weighting functions for lightness, chroma and hue, respectively. The  $k_L$ ,  $k_C$ , and  $k_H$  values are the parametric factors for the lightness, chroma, and hue components, respectively. They should be adjusted according to different viewing parameters such as textures, backgrounds, separations, etc. For more details, please refer to APPENDIX 1 in [Luo et al. 2001]. The conversion from RGB to CIELab is given by:

$$k \in \{R/255, G/255, B/255\} \quad (6.2.2)$$

$$R', G', B' = \begin{cases} k/4.5 & 0 \leq k < 0.081 \\ (\frac{k+0.099}{1.099})^{2.4} & 0.081 \leq k \leq 1 \end{cases} \quad (6.2.3)$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412391 & 0.357584 & 0.180481 \\ 0.212639 & 0.715169 & 0.072192 \\ 0.019331 & 0.119195 & 0.950532 \end{bmatrix} \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} \quad (6.2.4)$$

$$L = 116f(Y) - 16 \quad (6.2.5)$$

$$a = 500[f(X/0.950456) - f(Y)] \quad (6.2.6)$$

$$b = 200[f(Y) - f(Z/1.089058)] \quad (6.2.7)$$

$$f(t) = \begin{cases} t^{1/3} & t > 0.008856 \\ 7.78t + 16/116 & t \leq 0.008856 \end{cases} \quad (6.2.8)$$

## 6.2.2 Experiments

Experiments are performed on the HumanEvaI dataset [Sigal and Black 2006a] that contains 4 grayscale and 3 colour calibrated video streams synchronised with Mocap data at 60Hz. There are 4 subjects performing 6 common actions (e.g. walking, jogging, gesturing, etc.) in HumanEvaI. Our experiments are conducted on the subject 3 validate-walking sequence (trail 1) which has 443 frames. The tracking results are evaluated against the Mocap groundtruth data to obtain mean Euclidean joint position errors and standard deviations. Note that the ground truth data is corrupted at frames 91-108 and 163-176.

In the first experiment, we compare the proposed method with the SIR method and the silhouette based APF method. The proposed method uses only 3 colour videos, 50 particles and 10 temperature phases. The Sampling Importance Resam-

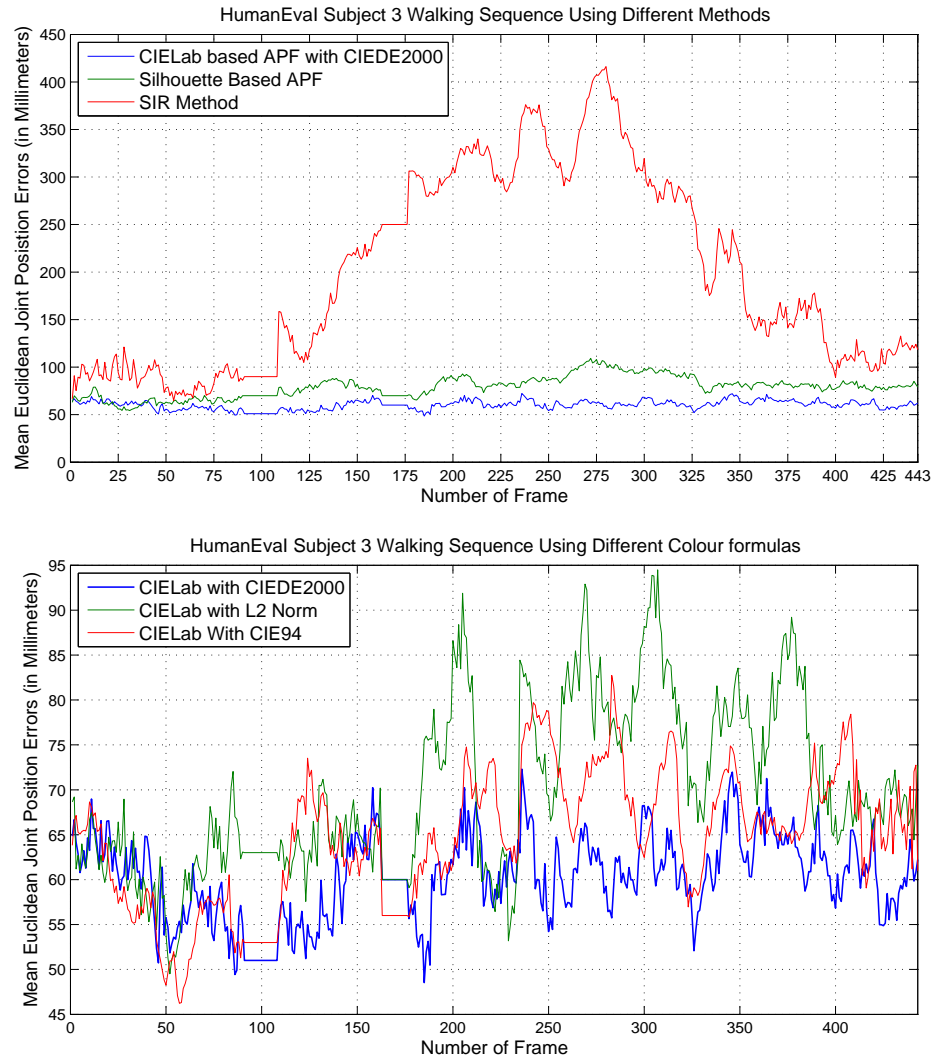


Figure 6.9: Experimental Results from HumanEval Subject 3 using different methods and different colour formulas: With different methods, the CIELab based APF with CIEDE2000 has superior accuracy over the other two methods. The SIR method suffers severe mistracking. Using the same APF framework but with different colour formulas, CIEDE2000 appears more robust and accurate over CIE94 and L2 Norm.

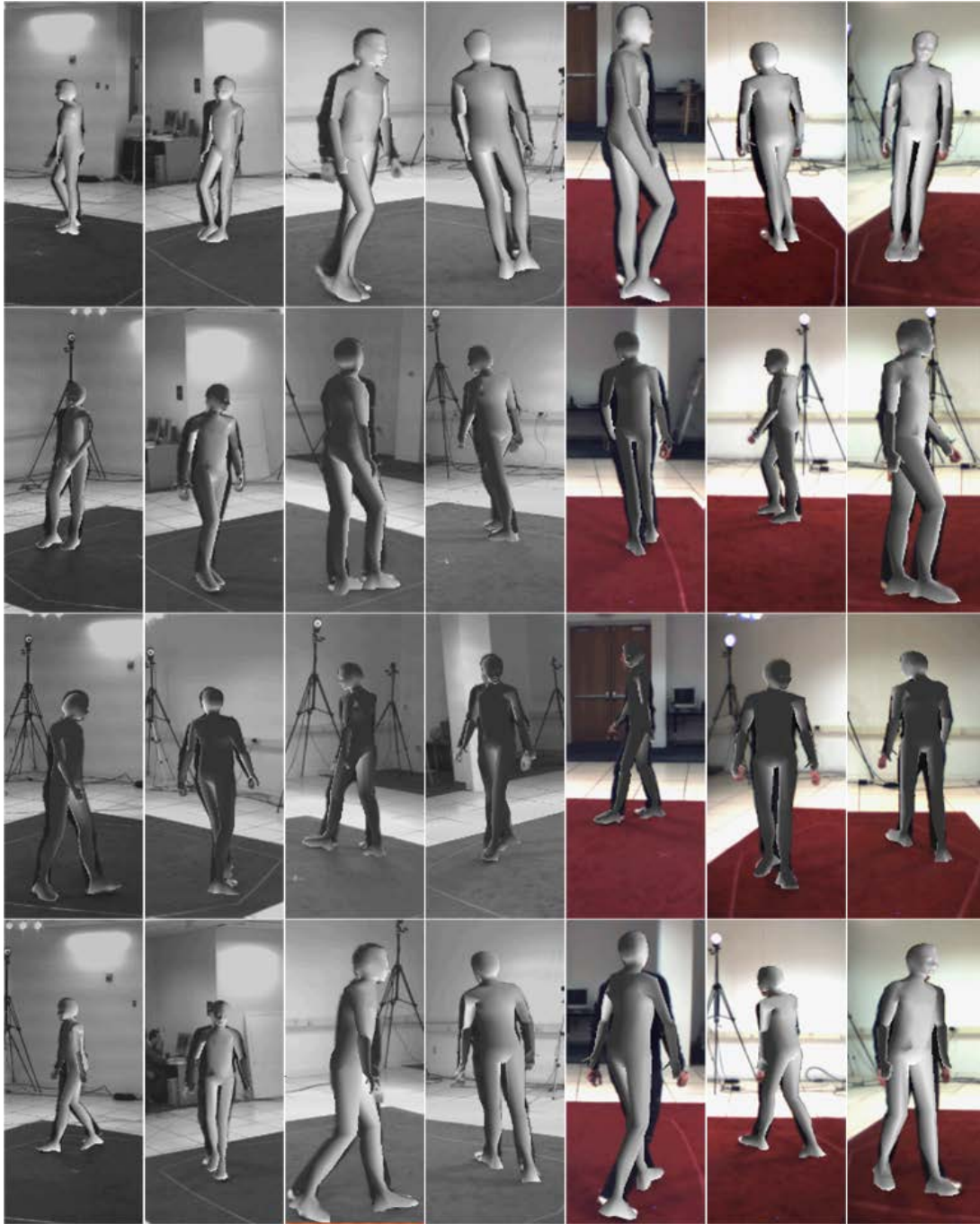


Figure 6.10: Visual tracking results from HumanEval show the visual human model overlapped with the real tracking subject

---

pling (SIR) method uses 7 videos with 500 particles<sup>4</sup>. The silhouette based APF method uses 7 videos, 50 particles and 10 temperature phases. As illustrated in the top of Figure 6.9, the results show the SIR method does not perform well and errors are over  $100mm$ . On the other hand, the silhouette based APF is able to track relatively well and maintain errors within  $79.1 \pm 11.3mm$ . The proposed CIELab based method with CIEDE2000 outperforms both methods, achieving  $60.2 \pm 4.7mm$ . In the second experiment, we compare the CIELab based method using different colour formulas, the  $L_2$  norm, CIE94 and CIEDE2000 formula. All of them employ 50 particles and 10 temperature phases. In the bottom of Figure 6.9, the experimental results show CIELab CIEDE2000 has the most stable performance at  $60.2 \pm 4.7mm$ , while CIELab CIE94 maintains  $64.6 \pm 7.3mm$  which is better than the  $69.9 \pm 9.3mm$  achieved by CIELab with  $L_2$  norm. The method using CIELab  $L_2$  norm occasionally fails to track the right arm when the right arm is occluded with the body. Thus it appears to have a more fluctuating trend. Overall, the CIELab based APF methods are generally superior to the SIR and silhouette-based method in terms of robustness and accuracy.

In summary, the SIR method is unable to scale well in high dimensional space. The silhouette based APF method has much better behaviour in high dimensional space owing to the fact that it focuses on approximating the global mode of the posterior distribution, but it can suffer from noise from silhouette segmentation. The proposed method overcomes both issues and achieves robust performance. Particularly, with the CIEDE2000 formula our method demonstrates more robust behaviour in the moderately illumination varied environment. More visual tracking results are shown in Figure 6.10.

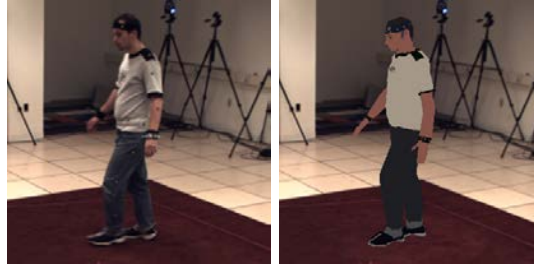


Figure 6.11: From left to right: an observed image and a synthesised image

### 6.3 Maximisation of Mutual Information

Given multi-view image observations, the estimated pose, and the pre-built template human model, the likelihood evaluation is performed by comparing the observed images against the synthesised images. A synthesised image is obtained by projecting the template human model onto a mean static background image based on camera calibration parameters and a given pose configuration. Figure 6.11 shows an observed image and a synthesised image from a tracked example in the HumanEvaII dataset [Sigal and Black 2006a]. A direct comparison of the two images in the commonly used RGB colour space will not be robust and is often affected by lighting conditions and appearance differences between the template model and the real subject. In this work, we employ robust image similarity metrics developed in the literature to overcome this problem. Particularly, the work by Viola et al [Viola and Wells 1997] suggests that Mutual Information (MI) metrics are reliable for evaluating models with substantially different appearances and are even robust with respect to variations in illumination. Mutual Information relies on the entropy of the images' underlying probability densities. This reliance on the probability densities of the two images enables the metric to transcend many constraints that bind other systems. For example, MI handles matching sample  $A$  with the negative of sample  $B$  as easily as simply matching  $A$  and  $B$ . Also, MI can often handle non-functional relationships between  $A$  and  $B$ , as occurs when the same object is viewed under two different spectra.

<sup>4</sup>The equivalent number of evaluations as the APF method

Mutual Information depends upon the entropy and joint entropy of two random variables. In the stereo case, the random variables are the image pixels we take from each image in a stereo pair. If we assume the pixel values  $\Psi$  are discrete random variables with density  $p(\Psi)$ , then we can define the entropy  $H$ :

$$H(X) = -E [\log(p(\Psi))] \quad (6.3.1)$$

An intuitive description of entropy is that it measures the randomness of a random variable. A low entropy means that the average probability over the support set for a given random variable is low. For example, a constant region in an image has a lower entropy than highly textured region. The joint entropy is defined similarly for two random variables  $\Psi$  and  $\Omega$ , replacing the univariate  $p$  with the joint probability function  $p(\Psi, \Omega)$ . Joint entropy can be used to measure alignment, or similarity, because it describes the “crispness” of a joint probability function. Two identical samples will have a lower joint entropy when aligned than when they are misaligned. However, two constant regions will have a low joint entropy as well. To avoid such spurious matches, we want to maximise the entropy in the individual samples that we are comparing. For this reason, we use Mutual Information. Mutual Information is a quantity that measures the mutual dependence of two variables. Considering two images  $\Psi, \Omega$ , and their pixels  $\psi, \omega$  as random variables, Mutual Information can be expressed in terms of entropy as:

$$\begin{aligned} I(\Psi; \Omega) &= H(\Psi) + H(\Omega) - H(\Omega, \Psi) \\ &= - \sum_{\psi \in \Psi} p(\psi) \log p(\psi) - \sum_{\omega \in \Omega} p(\omega) \log p(\omega) + \sum_{\omega \in \Omega} \sum_{\psi \in \Psi} p(\psi, \omega) \log p(\psi, \omega) \\ &= \sum_{\omega \in \Omega} \sum_{\psi \in \Psi} p(\psi, \omega) \log \left( \frac{p(\psi, \omega)}{p(\psi) p(\omega)} \right) \end{aligned}$$

where  $H(\Psi)$  and  $H(\Omega)$  denote the marginal entropies,  $H(\Psi, \Omega)$  the joint entropy, and  $p()$  the probability density function. Mutual Information is bounded, so  $0 \leq$

$MI(\Psi, \Omega) \leq \min(MI(\Psi, \Psi), MI(\Omega, \Omega))$ . The minimum value occurs when  $\Psi$  and  $\Omega$  are identical or there is a one-to-one mapping  $g$  between the two, since  $MI(\Psi, g(\Psi)) = MI(\Psi, \Psi)$ . These last points deserve special attention: they help to justify the performance of mutual information as a similarity metric. Mutual Information measures similarity, but it is also invariant to one-to-one transformations of the data. This invariance enables MI to measure similarity in more situations than many traditional similarity metrics. It also explains MI's limitations: when a transformation is not one-to-one, MI has difficulty measuring similarity. In these cases, a large sample size often alleviates the problem. In our case,  $p()$  is approximated by using the Parzen Window method with the Gaussian functions:

$$p(\psi) \approx \frac{1}{N} \sum_{\psi_i \in W} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\psi - \psi_i)^2}{2\sigma^2}\right)$$

$$p(\psi, \omega) \approx \frac{1}{N} \sum_{\psi_i \in W_\psi, \omega_i \in W_\omega} \frac{1}{2\pi\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2} \begin{bmatrix} \psi - \psi_i \\ \omega - \omega_i \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} \psi - \psi_i \\ \omega - \omega_i \end{bmatrix}\right)$$

where  $N$  denotes the number of samples in the window  $W$  or  $W_\psi, W_\omega$ .  $\Sigma$  is assumed as a diagonal covariance matrix. To be more robust to lighting conditions, MI is computed in the CIELab colour space in our work. Overall, the energy function  $E(\mathbf{y}_t, \mathbf{x}_t)$  can be summarised as:

$$E(\mathbf{y}_t, \mathbf{x}_t) = \frac{1}{N_{view}} \sum_{i=1}^{N_{view}} \frac{1}{k_L I_L(IM_{\mathbf{y}_t}^i; IM_{\mathbf{x}_t}^i) + k_a I_a(IM_{\mathbf{y}_t}^i; IM_{\mathbf{x}_t}^i) + k_b I_b(IM_{\mathbf{y}_t}^i; IM_{\mathbf{x}_t}^i)} \quad (6.3.2)$$

where  $IM_{\mathbf{y}_t}^i$  denotes the  $i$ th view observed image  $\mathbf{y}_t$  at time  $t$ ,  $IM_{\mathbf{x}_t}^i$  the  $i$ th view synthesised image produced by projecting the estimate state  $\mathbf{x}_t$  at time  $t$ , and  $I_L()$ ,  $I_a()$  and  $I_b()$  the MI criterion values calculated in the channels  $L$ ,  $a$  and  $b$ , respectively. Also,  $k_L, k_a$  and  $k_b$  denote the coefficients that control the weights of the  $L$ ,  $a$  and  $b$  channels. Usually,  $k_L$  is set to be small in order to suppress the illumination influence.



## 6.4 Gradual Sampling for Annealed Particle Filter

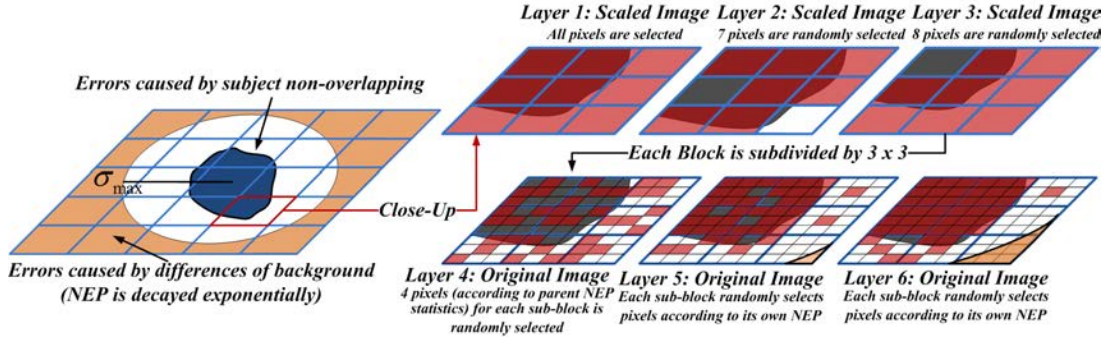


Figure 6.12: Error oriented pixel selection with 6 layers

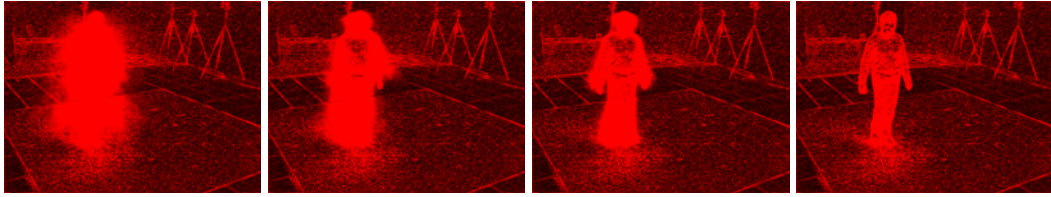


Figure 6.13: Large errors (shaded in red) among 200 particles gradually concentrate on the tracking subject during the course of optimisation

APF works in an iterative process in which the current exploration is built upon the exploitation of historical information. However, at the early stage, explorations are prone to being misled by strong local minima because of a lack of information about the global shape of the energy function. A sufficiently slow annealing schedule is usually taken as a measure to rescue particles from the attraction of local minima. With such an annealing schedule, the initial energy function is shaped in a way that local minima are flattened out whereas the global minimum becomes relatively more pronounced. Also, particles will have more evaluations and random perturbations to move into the neighbourhood of the global mode. However, a sufficiently slow annealing schedule is usually computationally intractable in practice.

We address the above issues by proposing a gradual sampling scheme. The key ideas are that 1) The precision of the energy function evaluations changes adaptively with the process of annealing. We blur the observed images and the synthesized im-

ages at the early stage and use these blurred versions to evaluate the energy function. We observe that blurred images are able to flatten the shape of the energy function, allowing a large number of particles to survive and encouraging a broader exploration. Then we gradually increase the resolution of the observed images and the synthesized images and use them to evaluate the energy function at the later stages. This will provide more precise information for the algorithms to better discriminate the surviving particles and correctly identify those close to the global optimum of the energy function; 2) With the increase of layers in the APF, the majority of large errors will progressively concentrate on the tracking subject and only on some body areas where the observed and synthesised images have not been well aligned with each other. Taking this situation into account, we propose a "smart" selection of the image areas (or extremely, the pixels) that will be used in the evaluation of the energy function.

This special selection has a very strong connection with the weighted mutual information in [Guiasu 1977]. It addresses the fact that in some situations certain objects or events are more significant than others, or that certain patterns of association are more semantically important than others. [Rodrguez-Carranza et al. 1999] proposed a weighted mutual information method to attack the image registration problem. They found that normalized mutual information (for 2D image registration) provides a larger capture range and is more robust, with respect to the optimisation parameters, than the non-normalized measure. In this work, weighted mutual information is intuitively incorporated to the APF frame by Gradual Sampling. In detail, different image areas are weighted based on the magnitude of the error from them in the last layer. Two criteria are used to weight each image area: i) if the magnitude of the error in an area is small (meaning that the observed and the synthesized have been well aligned in this region), then its weight will be lowered; ii) if the image area is far from the centre of the projection of the human body, its weight will be lowered. This is because, as mentioned above, the majority of large errors will progressively concentrate on the tracking subject with the increase of layers in APF. This case is shown in Fig-

ure 6.13 which plots the error distribution evolving through optimisation. For an area with lower weight, fewer pixels will be sampled for inclusion in the energy function evaluation. This is where the name “Gradual Sampling” comes from.

In this work, we partition the synthesized and observed images into a number of small-sized non-overlapping blocks, and compute the distribution of error over these blocks. Then the error-oriented block selection and pixel sampling use the block-based error distribution from the last exploration to guide the current evaluation. This allows the evaluation to focus more on the large-error area and at the same time reduce the number of pixels involved in function evaluation. The whole process relies on a measurement called Number of Effective Pixels <sup>5</sup> (NEP) defined for the  $i$ th block. For the layer  $m$  and each view, it is expressed as:

$$NEP_{m,i} = \begin{cases} N_i \cdot \eta_{m,i} & \|c_i - C_{bo}\|_2 < \sigma \\ N_i \cdot \eta_{m,i} \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\|c_i - C_{bo}\|_2^2}{2\sigma^2}\right) & \|c_i - C_{bo}\|_2 \geq \sigma. \end{cases} \quad (6.4.1)$$

This definition is explained as follows.  $\eta_{m,i}$  is one factor controlling the percentage of pixels sampled from the  $i$ th block. It is proportional to the ratio of the average error from the  $i$ th block (denoted by  $err_{m,i}$ ) to the maximal average error recorded over all of the blocks (denoted by  $\max_i[err_{m,i}]$ ). In doing so,  $NEP_i$  for the block with smaller errors will decrease more quickly, whereas  $NEP_i$  for the block with greater errors will maintain relatively larger and decrease at a slower pace. This helps to gradually concentrate misalignments. Also, during the course of the annealing schedule, considering the fact that misalignments between the template model and real subject are gradually decreasing, we will progressively decrease the total number of pixels involved in function evaluation. This speeds up function evaluation and the tracking speed in turn. To realize this,  $\eta_{m,i}$  is proportional to  $\beta_m$ , which decreases with the

<sup>5</sup>Pixels with large errors often correspond to misalignments of the subject, and they contribute to “effective” measurements.

layer number,  $m$ . Therefore,  $\eta_{m,i}$  is mathematically expressed as

$$\eta_{m,i} = \beta_m \cdot \frac{\cdot err_{m,i}}{\max_i(err_{m,i})}. \quad (6.4.2)$$

Equation (6.4.1) also takes the distance of the  $i$ th block from the tracking subject into account. We use a Gaussian distribution based weighting scheme to exponentially decay the importance of the  $i$ th block. The farther the centroid of the  $i$  block (denoted by  $c_i$ ) from the centroid of the human body template projected onto the image plane (denoted by  $C_{bo}$ ), the less important this block is and the lower the number of sampled pixels. This makes the energy function evaluation concentrate on the error around the tracking subject.  $\sigma$  denotes the standard deviation of this Gaussian distribution. It will be empirically set at the beginning of the tracking based on the scattering radius of the projection of the human body template in a “T” gesture.

We illustrate a typical procedure of the error-oriented pixel sampling in Figure 6.12. At the first layer, the blurred images are used, and all pixels of the blurred images are selected for evaluating the energy function. At the second layer,  $NEP_i$  is calculated for each block based on Equation (6.4.1) and  $NEP_i$  pixels are randomly selected from the  $i$ th block for function evaluations, and the obtained error distribution is used to infer the  $NEP_i$  values for the third layer. At the fourth layer, the blurred images are replaced with the original images to achieve more accurate evaluation. Each block in the first three layers is now subdivided into many smaller-sized blocks and the  $NEP_i$  value for each sub-block is computed according to its parent block’s  $NEP$  statistics. Then the same procedure is repeated. In summary, our gradual sampling scheme shapes the energy function by carefully manipulating the image data and can achieve a similar effect to a sufficiently slow annealing schedule without incurring intractable computational load. It provides a better chance for the particles in the APF-based tracking framework to escape from local minima and thus increases the possibility of convergence to the global optimum. Our gradual sampling procedure at time  $t$  is

outlined in Algorithm 12.

---

**Algorithm 12** Gradual Sampling for a typical frame at time  $t$ 


---

**Require:** The survive rate  $\alpha_m$  in APF [Deutscher et al. 2000], a set of predefined  $\beta_m$ , observation  $\mathbf{y}_t$ , the total number of layers  $M$ , and the initial covariance matrix  $\mathbf{P}_0$ .

**for**  $m = 1$  to  $M$  **do**

- 1: Initialise  $N$  particles  $\mathbf{x}_t^1, \dots, \mathbf{x}_t^N$  from the last layer or the temporal model;
- 2: Evaluate  $E(\mathbf{y}_t, \mathbf{x}_t)$  for each particle with the  $NEP_i$  pixels sampled from each block of blurred/original images. Average the error statistics over all particles;
- 3: Compute  $NEP_i$  for each block with the error statistics and equation (6.4.1)
- 4: Calculate  $\lambda_m$  by solving  $\alpha_m N \sum_{i=1}^N (w_{t,m}^i)^2 = (\sum_{i=1}^N w_{t,m}^i)^2$  where  $w_{t,m}^i = \exp\{-\lambda_m E(\mathbf{y}_t, \mathbf{x}_t^i)\}$  and  $N$  is the number of particles;
- 5: Update weights for all particles using  $\exp\{-\lambda_m E(\mathbf{y}_t, \mathbf{x}_t)\}$ .
- 6: Resample  $N$  particles from the updated importance weight distribution.
- 7: Perturb particles by Gaussian noise with covariance  $\mathbf{P}_m = \mathbf{P}_{m-1} \alpha_m$ .

**end for**

---

### 6.4.1 Connection between Gradual Sampling and Annealing Variable

Image scaling and the error oriented pixel selection used in gradual sampling are not arbitrary choices. They actually have a strong connection to the annealing variable of Simulated Annealing. Considering a pixel-wise homogeneity metric  $d(\cdot)$  that satisfies  $d(a\mathbf{x}, a\mathbf{y}) = |a|d(\mathbf{x}, \mathbf{y})$ , the likelihood probability  $p(\mathbf{y}_t|\mathbf{x}_t)$  can be given as:

$$\begin{aligned}
 p(\mathbf{y}_t|\mathbf{x}_t) &= \exp\{-\lambda E(\mathbf{y}_t, \mathbf{x}_t)\} \\
 &= \exp\left\{\frac{-\lambda}{N_{view}} \sum_{i=1}^{N_{view}} \sum_{k \in w} \sum_{l \in h} d(IM_{\mathbf{y}_t}^i(l, k), IM_{\mathbf{x}_t}^i(l, k))\right\} \\
 &= \exp\left\{\frac{-1}{N_{view}} \sum_{i=1}^{N_{view}} \prod_{k \in w} \prod_{l \in h} \exp\{\lambda d(IM_{\mathbf{y}_t}^i(l, k), IM_{\mathbf{x}_t}^i(l, k))\}\right\}
 \end{aligned} \tag{6.4.3}$$

From the above equation, as  $\lambda$  gradually increases, the big error terms are magnified exponentially in the form of the Boltzmann distribution. Therefore, the annealing schedule pays exponentially increasing attention to eliminate large errors which usually correspond to the global mode. Since  $\lambda$  is strictly positive and because of the

homogeneity of  $d()$ , we can have:

$$p(\mathbf{y}_t|\mathbf{x}_t) = \exp\left\{\frac{-1}{N_{view}}\right\} \prod_{i=1}^{N_{view}} \prod_{k \in w} \prod_{l \in h} \exp\{d(\lambda IM_{\mathbf{y}_t}^i(l, k), \lambda IM_{\mathbf{x}_t}^i(l, k))\} \quad (6.4.4)$$

Weighting each pixel by  $\lambda$  (e.g.  $\lambda IM_{\mathbf{y}_t}^i(l, k)$ ) can be theoretically simulated by a uniform image scaling on “continuous” images. In other words,  $d(\lambda IM_{\mathbf{y}_t}^i(l, k), \lambda IM_{\mathbf{x}_t}^i(l, k))$  can be replaced by  $d(IM_{\mathbf{y}_t}^{i(\lambda)}(l, k), IM_{\mathbf{x}_t}^{i(\lambda)}(l, k))$ . Then, let us define a  $\lambda$ -factor scaled image  $IM^{(\lambda)}$  that always has  $\lambda$  weighted pixels. Assuming that the image is continuous<sup>6</sup> and the intensity in the sub-pixel level is uniformly distributed, more general  $\lambda$ -factor pixel scaling can be defined by:

$$\lambda IM(l, k) = IM^{(\lambda)}(l, k) = \begin{cases} IM^{(\lambda)sub}(l, k) & 0 < \lambda < 1 \\ IM^{(\lambda)sup}(l, k) & \lambda \geq 1 \end{cases}$$

where  $IM^{(\lambda)sup}(l, k)$  denotes a super-pixel with respect to the  $\lambda$ -factor equal to  $\int_0^\lambda IM(l, k)dx$ , and  $IM^{(\lambda)sub}(l, k)$  denotes a sub-pixel with respect to the  $\lambda$ -factor satisfying:

$$IM(l, k) = \int_0^{\frac{1}{\lambda}} IM^{(\lambda)sub}(l, k)dx \quad 0 < \lambda < 1$$

An example of the  $\lambda$ -factor image scaling is graphically displayed in Figure 6.14. The large image can be regarded as the  $\lambda = 9$ -factor scaled image of the small image. Assuming the pixel colours are identical for neighbours, each pixel is duplicated 9 times which produces the 9 times upsampled large image. Vice versa, the small image can be regarded as the  $\lambda = 1/9$ -factor scaled image of the large image. In practice, there are limitations to using uniform image scaling to simulate arbitrary  $\lambda$  – when  $\lambda$  is too large or small, the actual image does not exist at all. Since  $\lambda$  is bound to  $(0, 1)$  [Deutscher et al. 2000] and  $\lambda$ 's increasing trend is relatively steady, our gradual sampling technique can produce reasonable results. In short, the effect of increasing

<sup>6</sup>It ideally has infinite resolution

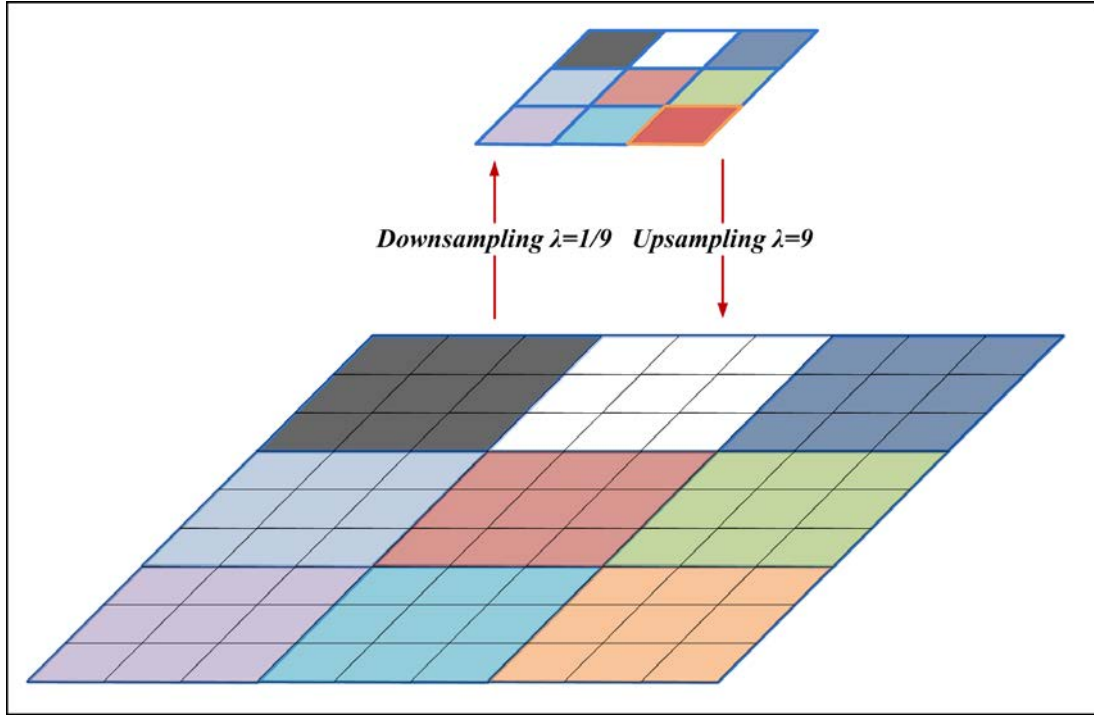


Figure 6.14: The  $\lambda$ -factor uniformly scaled image. From top to bottom, upsampling is equivalent to multiplying by the  $\lambda = 9$ -factor. Vice versa, downsampling is equivalent to multiplying by the  $\lambda = 1/9$ -factor.

$\lambda$  from small to large has two effects: 1) a uniform coarse-to-fine operation, 2) and gradual manifestation of big error terms. These are consistent with the image scaling and error oriented pixel selection used in our gradual sampling.

### 6.4.2 Experiments and Discussion

Experiments are performed on the benchmark dataset HumanEval [Sigal and Black 2006a] that contains 4 grayscale and 3 colour calibrated video streams synchronised

Study	Particles	Layers	Errors( <i>ave</i> $\pm$ <i>std</i> )mm	comments
[Gall et al. 2010]	250	15	$32 \pm 4.5$	two-pass optimisation with smoothing
<b>Ours</b>	<b>200</b>	<b>10</b>	<b><math>54.6 \pm 5.2</math></b>	<b>MI and Gradual sampling</b>
[Bandouch and Beetz 2009]	800	10	50-100	hierarchical approach
[Sigal et al. 2010]	200	5	$80 \pm 5$	Bi-directional silhouette-based
[Cheng and Trivedi 2007]	N/A	N/A	over 170	mixed learning and tracking

Table 6.1: Absolute mean joint position errors on HumanEvalII Subject 4 from different research groups

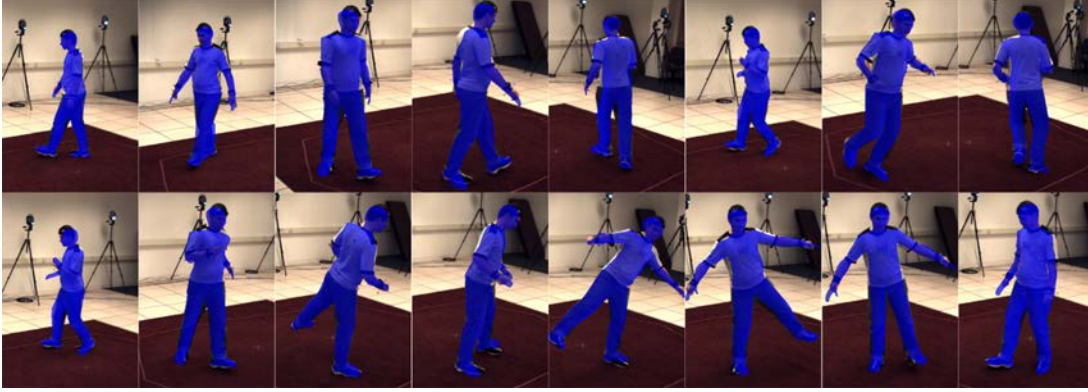


Figure 6.15: Accurate tracking results from HumanEvaII Subject 4. The average Euclidean error of 3D joint positions is less than 55mm.

with Mocap data at 60Hz, and HumanEvaII that contains 4 colour calibrated image sequences synchronised with Mocap data at 60Hz. The tracking results are evaluated against the groundtruth Mocap data to obtain the absolute mean joint centre position errors and standard deviations. The experimental results with 50 particles and 10 layers on the 443-frame trail 1 of the subject 3 walking sequence in HumanEvaI is plotted in the top of Figure 6.16. The proposed method using only 3 colour video streams is able to maintain  $64.5 \pm 8.2mm$  (using both MI and GS),  $69.4 \pm 9.9mm$  (without using MI) and  $68.6 \pm 8.0mm$  (without using GS). The differences among them demonstrate the effects of incorporating MI and GS in human motion tracking. All of the three methods outperform the silhouette-based method<sup>7</sup> which only maintains  $78 \pm 12.8mm$  using 7 video streams.

Another experiment uses 200 particles and 10 layers for the longer 1257-frame combination sequence of subject 4 in HumanEvaII. The ground truth Mocap data is withheld by the data set owner and is only available for online evaluation. We submitted our tracking result and obtained the online evaluation results as follows. As shown, in the middle of Figure 6.16 the proposed method is able to achieve  $54.6 \pm 5.2mm$ . Without the support of MI and GS, errors rise to  $64.3 \pm 12.2mm$  and  $59.38 \pm 5.5mm$ , respectively. In contrast, the silhouette-based method with the same settings

<sup>7</sup>The silhouette based method uses only the silhouette feature.



can only achieve  $90.7 \pm 16.7mm$  and the maximum error reaches about  $170mm$ . This shows the advantage of our appearance-based likelihood evaluation function over a silhouette-based one. Although the jogging from frames 400 to 800 is more difficult to track than the slow movements of walking and balancing, the proposed method using MI can still track stably whereas the other two methods without MI experience drastic fluctuations. As illustrated in Table 6.1, several research groups [Gall et al. 2010; Bandouch and Beetz 2009; Sigal et al. 2010; Cheng and Trivedi 2007] have evaluated their results against subject 4 of HumanEvaII. Our results achieved the second best performance overall. The work in [Cheng and Trivedi 2007] proposed a learning based approach with errors over  $170mm$ , which is relatively inaccurate when compared with APF based approaches. The work in [Sigal et al. 2010] utilised bi-directional silhouette-based evaluation and achieved  $80 \pm 5mm$ . However, it relies on the quality of silhouette segmentation. The work in [Bandouch and Beetz 2009] proposed a hierarchical approach that employs a relatively large number of evaluations to achieve errors within  $50 - 100mm$ . The method in [Gall et al. 2010] can be expected to perform better than ours because they utilise two-pass optimisation. In the second pass, a smoothing process with respect to future frames is used. These are not undertaken in our approach because two-pass optimisation incurs more computational overhead and limits its applicability to real-time tracking. Moreover, as pointed out in [Bandouch and Beetz 2009; Corazza et al. 2010; Cheng and Trivedi 2007], when the error is less than  $50mm$ , the actual tracking error will not be measurable because of the limited precision of the joint centres' positions estimated from the Mocap data, which is considered as ground truth, and the intrinsic error between the human model and the real subject<sup>8</sup>. Therefore, considering this context, our performance of  $54.6 \pm 5.2mm$  is almost the best possible. Also, this context explains why there is approximately  $50mm$  error for our initial pose even though it is accurately set.

---

<sup>8</sup>Note that there are no markers corresponding to actual joint centres in the Mocap data. As a result, the joint centres' positions cannot be recovered very accurately from the Mocap data.

More results are presented in Figure 7.6.

The performance experiment is set up on a dual core Windows system with a 2.8GHz CPU and 4GB RAM. The average computational time per frame is compared with that of the baseline algorithm benchmark<sup>9</sup> [Sigal et al. 2010] which is the only publicly available implementation for the HumanEva dataset. We run both algorithms in different combinations of layers and particles and the results are shown in the bottom of Figure 6.16. Despite the extra computational overhead due to the use of Mutual Information criterion, our method with gradual sampling can still achieve almost 10 times faster calculation than the baseline algorithm in [Sigal et al. 2010].

---

<sup>9</sup>Available online via <http://vision.cs.brown.edu/humaneva/baseline.html>

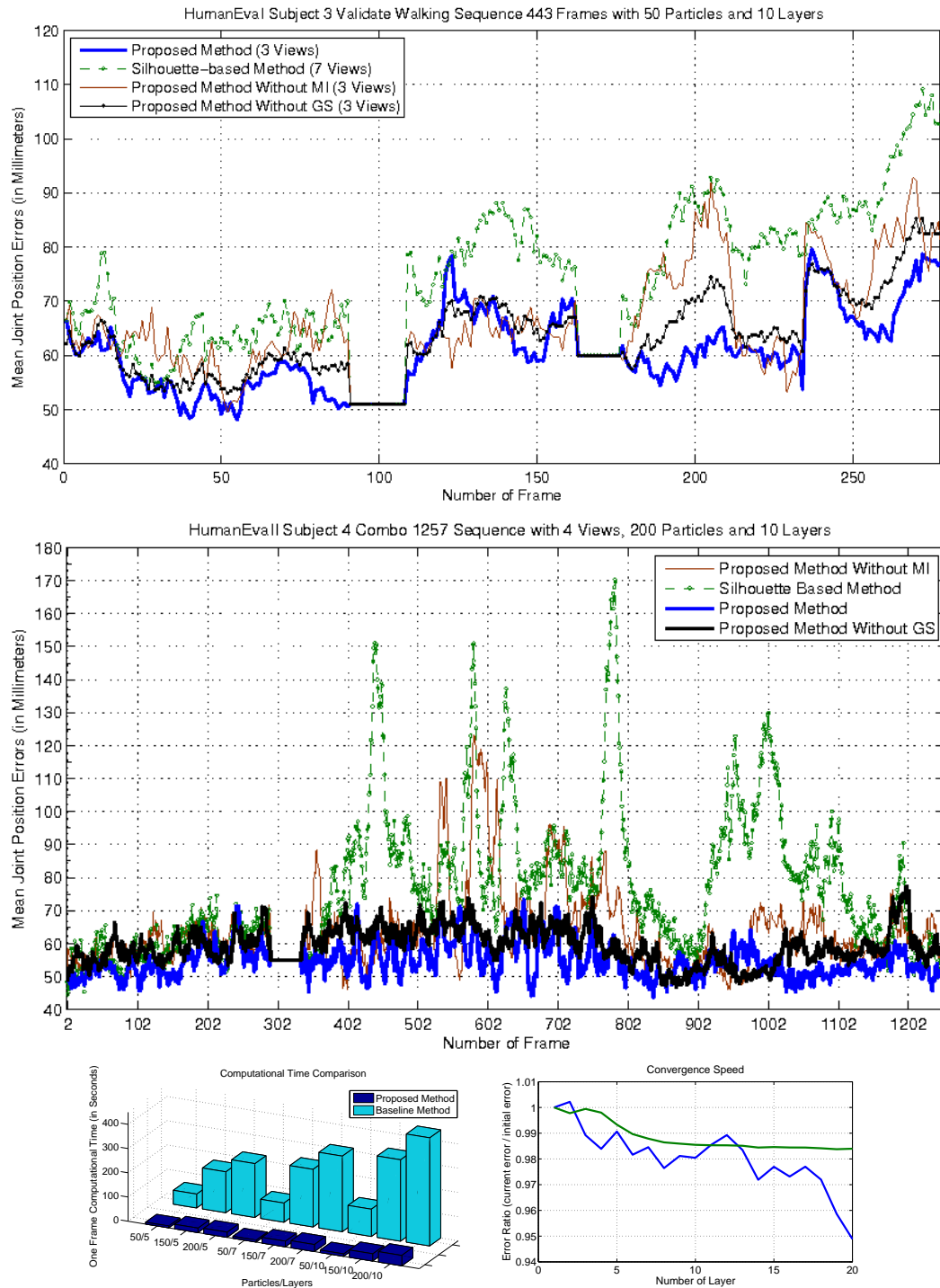


Figure 6.16: From the top to bottom, 1) tracking results on the HumanEval Subject 3 Walking Sequence, 2) the HumanEvalII Subject 4 Combo Sequence, 3) computational time comparison for the proposed and baseline methods with the different number of particles and layers, and 4) the convergence speed of Gradual Sampling versus the Common APF method. Note that the ground truth data is corrupted at frames 91-108 and 163-176 in HumanEval and at frames 298-335 in HumanEvalII.



# Compressive Evaluation

---

Compressive sensing (CS) reconstructs compressible signals from a small number of non-adaptive linear random measurements by combining the steps of sampling and compression [Candes and Tao 2005; Candès et al. 2006; Candès et al. ; Candes and Romberg 2007]. It enables the design of new kinds of compressive imaging systems, including a single pixel camera [Duarte et al. 2008], with some attractive features: simplicity, low power consumption, universality, robustness, and scalability. Recently, there has been a growing interest in compressive sensing in computer vision and it has been successfully applied to face recognition, background subtraction, object tracking and other problems. Wright et al [Wright et al. 2009] represented a test face image as a linear combination of training face images. Their representation is naturally sparse, involving only a small fraction of the overall training database. The problem of classifying multiple linear regression models can be then solved efficiently via  $L1$ -minimisation which seeks the sparsest representation and automatically discriminates between the various classes presented in the training set. Cevher et al [Cevher et al. 2008] cast the background subtraction problem as a sparse signal recovery problem and solved it by greedy methods as well as total variation minimisation of convex objectives to process field data. They showed that it is possible to recover the silhouettes of foreground objects by learning a low-dimensional compressed representation of the background image, without learning the background itself, and to sense changes in the foreground objects. Mei et al [Mei and Ling 2009] formulated the

tracking problem similarly to [Wright et al. 2009]. In order to find the tracking target at a new frame, each target candidate is sparsely represented in the space spanned by target templates and trivial templates. The sparse representation is obtained by solving an  $L1$ -regularised least squares problem to find good target templates. Then the candidate with the smallest projection error is taken as the tracking target. Subsequent tracking is continued using a Bayesian state inference framework in which a particle filter is used for propagating sample distributions over time.

Unlike above works, many data acquisition/processing applications do not require obtaining a precise reconstruction, but rather are only interested in making some kind of evaluation of the objective function. Particularly, human motion tracking essentially attempts to find the optimal value of the observation likelihood function. Therefore, we propose a new framework, called Compressive Annealed Particle Filter, for such a situation that bypasses the reconstruction and performs evaluations solely on compressive measurements. It has been proven [Candes and Tao 2005] that random projections can approximately preserve isometry and pairwise distances, when the number of linear measurements is large enough (still much smaller than the original dimension of the signal). Moreover, noticing the annealing schedule is a coarse-to-fine process, we introduce the staged wavelet decomposition with respect to each annealing layer so that the additional annealing variable is absorbed into the wavelet decomposition. As a result, the number of compressive measurements is progressively increased to gain computational efficiency.

## 7.1 Compressive Sensing

The classic approach of reconstructing signals or images from measured data satisfies the well-known Nyquist-Shannon sampling criterion [Shannon 1949], which states that the sampling rate must be at least twice that of the highest frequency. Similarly, the fundamental theorem of linear algebra suggests that the number of collected sam-

ples (measurements) of a discrete finite-dimensional signal should be at least as large as its length (dimension) in order to ensure reconstruction. This principle underlies most current technological devices such as analog to digital converters, medical imagers and audio/video electronics. The novel theory of Compressive Sensing (CS) provides a fundamentally new approach to data acquisition which provides a stricter sampling condition when the underlying signal is known to be sparse or compressible, yielding a sub-Nyquist sampling criterion.

### 7.1.1 Signal Sparse Representation

Considering a signal  $\mathbf{f} \in \mathbb{R}^N$  that may contain considerable redundant and irrelevant information, a transformation is performed by left multiplying an  $N \times N$  orthogonal matrix  $\Psi^T$  with  $\mathbf{f}$ :

$$\mathbf{f}' = \Psi^T \mathbf{f}$$

The inverse transformation recovers the original signal from  $\mathbf{f}'$ :

$$\mathbf{f} = \Psi \mathbf{f}'$$

The  $\Psi$  is constructed such that the correlation between any entries and the significant entries of  $\mathbf{f}'$  are minimised, so that insignificant entries can be discarded without much perceptual loss. Then,  $\mathbf{f}'$  can be well approximated by  $\mathbf{f}'_K$  that is constructed by selecting the most significant (largest)  $K$  entries of  $\mathbf{f}'$ , keeping only the largest  $K$  entries unchanged and setting all remaining  $N - K$  entries to zero. The signal can be recovered by  $\mathbf{f}'_K$  which has only a small number of non-vanishing entries:

$$\mathbf{f}_K = \Psi \mathbf{f}'_K$$

This is the so called  $K$ -sparse representation. Since  $\Psi$  is an orthogonal matrix,  $\|\mathbf{f} - \mathbf{f}_K\|_2 = \|\mathbf{f}' - \mathbf{f}'_K\|_2$ , and if  $\mathbf{f}'$  is sparse or compressible in the sense that the sorted

magnitudes of the  $x_i$  decay quickly, then  $\mathbf{f}'$  is well approximated by  $\mathbf{f}'_K$ , the error  $\|\mathbf{f}' - \mathbf{f}'_K\|_2$  is small, and the relative error  $\frac{\|\mathbf{f} - \mathbf{f}_K\|_2}{\|\mathbf{f}\|_2}$  is also small. Therefore, the perceptual loss of recovery is hardly noticeable.

This principle reveals what underlies most modern lossy coding schemes: compute  $\mathbf{f}'$  from  $\mathbf{f}$  via  $\mathbf{f} = \Psi\mathbf{f}'$ , then adaptively encode the locations and values of the  $K$  significant entries, and finally in the decoding stage put the locations and values back into  $\mathbf{f}'_K$  and recover  $\mathbf{f}_K$  via  $\mathbf{f}_K = \Psi\mathbf{f}'_K$ . Such a process requires knowledge of all  $N$  entries of  $\mathbf{f}'$ , as the locations of the significant pieces of information may not be known in advance (they are signal dependent). Therefore, this process has to be adaptive.

### 7.1.2 L1 Minimisation and Reconstruction

The above outlined adaptive strategy of compressing a signal  $\mathbf{f}$  by only keeping its largest coefficients is valid when complete information about  $\mathbf{f}$  is available. One may ask whether it is possible to more directly obtain a compressed version of the signal by taking only a small amount of linear and nonadaptive measurements. Compressive sensing surprisingly predicts that reconstruction from vastly undersampled nonadaptive measurements is possible—even by using efficient recovery algorithms. Taking  $M$  linear measurements of a signal  $\mathbf{f}$  corresponds to applying the measurement/sensing  $M \times N$  matrix  $\Phi$ , where,  $M \ll N$ .

$$\mathbf{z} = \Phi\mathbf{f}$$

The main interest is to recover  $\mathbf{f}$  from  $\mathbf{z}$ , called the measurement vector. Since the linear system is highly underdetermined, without further information the recovery is impossible and therefore has infinitely many solutions. If, however, the additional assumption is imposed that the vector  $\mathbf{f}$  has sparse representation, then the recovery can be realised by searching for the sparsest vector  $\mathbf{f}'^*$  which is consistent with the measurement vector  $\mathbf{z} = \Phi\Psi\mathbf{f}'$ . The finest recovery  $\mathbf{f}^* = \Psi\mathbf{f}'^*$  is achieved when the



sparsest vector  $\mathbf{f}^*$  is found. This leads to solving the  $L_0$ -minimisation problem:

$$\min \|\mathbf{f}'\|_0 \quad \text{subject to} \quad \mathbf{z} = \Phi \Psi \mathbf{f}' \quad (7.1.1)$$

Unfortunately, the combinatorial  $L_0$ -minimisation problem is NP hard in general [Natarajan 1995] as it contains the subset sum problem, and so is computationally infeasible for all but the tiniest data sets. The work of Candes et al [Candès et al. 2006], in which it was shown that the  $L_1$  norm is equivalent to the  $L_0$  norm under some conditions, leads one to solve an easier linear program, for which efficient methods already exist. The  $L_1$ -minimisation approach considers the solution of

$$\min \|\mathbf{f}'\|_1 \quad \text{subject to} \quad \mathbf{z} = \Phi \Psi \mathbf{f}' \quad (7.1.2)$$

which fortunately is a convex optimisation problem and can be seen as a convex relaxation of Equation (7.1.1). Various efficient convex optimisation techniques can be applied to solve this problem [Boyd and Vandenberghe 2004]. In the real-valued case, Equation (7.1.2) is equivalent to a linear program and in the complex-valued case it is equivalent to a second order cone program. Two practical approaches have been proposed in the literature: 1) convex relaxation leading to  $L_1$ -minimisation, also called basis pursuit [Chen et al. 1999], and greedy algorithms, for example various matching pursuits [Tropp and Gilbert 2005].

### 7.1.3 Incoherence Sampling

Equation (7.1.2) does not provide unconditional recovery in all cases. The recovery ability actually depends on the properties of the measurement/sensing matrix  $\Phi$ . This is characterised by the coherence between the sensing basis  $\Phi$  and the repre-

sensation basis  $\Psi$  given by:

$$\mu(\Phi, \Psi) = \sqrt{N} \max_{l,k \in [1,N]} |\langle \phi_l, \psi_k \rangle|$$

where,  $\phi_l$  is a row of  $\Phi$ , and  $\psi_k$  is a column of  $\Psi$ . To simplify the notation,  $\phi_l$  can be concatenated as the basis with  $N$  elements so that  $\langle \phi_l, \psi_k \rangle$  is always computable. The coherence measures the largest correlation between any two elements of  $\Phi$  and  $\Psi$ . If  $\Phi$  and  $\Psi$  contain correlated elements, the coherence is large. Otherwise, it is small. In this definition, the coherence has a range  $\mu(\Phi, \Psi) \in [1, \sqrt{N}]$ . The work in [Candes and Romberg 2007] demonstrates a strong theorem that asserts that when  $\mathbf{f}$  is sufficiently sparse, the recovery via  $L_1$ -minimisation is provably exact.

**Theorem 7.1.1.** *Fix  $\mathbf{f} \in R^N$  and suppose that the sequence  $\mathbf{f}'$  of  $\mathbf{f}$  in the basis  $\Psi$  is  $K$ -sparse. Select  $M$  measurements in the  $\Phi$  domain uniformly at random. Then if*

$$M \geq C\mu^2(\Phi, \Psi)K \log N$$

*for some positive constant  $C$ , the solution to Equation (7.1.2) is exact with overwhelming probability.*

[Candes and Romberg 2007] also shows that the probability of success exceeds  $1 - \delta$  if  $M \geq C\mu^2(\Phi, \Psi)K \log N$ . In addition, the result is only guaranteed for nearly all sign sequences  $\mathbf{f}'$  with a fixed support. The smaller the coherence, the fewer samples are needed, hence Compressive Sensing is mainly concerned with low coherence. If the coherence  $\mu(\Phi, \Psi)$  is equal or close to one, then on the order of  $K \log N$  samples suffice instead of  $N$ . The signal  $\mathbf{f}$  can be exactly recovered, regardless of any prior knowledge about the number of nonzero coordinates of  $\mathbf{f}'$ , their locations, or their amplitudes which are assumed all completely unknown a priori.

### 7.1.4 Restricted Isometry Property

To simplify the notation and representation of the problem, we discuss the abstract problem (derived from the above section) of recovering a vector  $\mathbf{f}' \in \mathbb{R}^N$  from the data:

$$\mathbf{z} = \mathbf{A}\mathbf{f}' + \boldsymbol{\eta} \quad (7.1.3)$$

where  $\mathbf{A}$  is an  $M \times N$  matrix giving us information about  $\mathbf{f}'$ , and  $\boldsymbol{\eta}$  is a stochastic or deterministic unknown error term.  $\mathbf{A}$  can be defined as  $\mathbf{A} = \boldsymbol{\Phi}\boldsymbol{\Psi}$ . Then the more general tool for studying the robustness of Compressive Sensing, the Restricted Isometry Property [Candes and Tao 2005] (RIP) can be defined as follows:

For each integer  $K = 1, 2, \dots$ , define the isometry constant  $\delta_K$  of a matrix  $\mathbf{A}$  as the smallest number such that

$$(1 - \delta_K) \leq \frac{\|\mathbf{A}\mathbf{f}'\|_2^2}{\|\mathbf{f}'\|_2^2} \leq (1 + \delta_K)$$

holds for all  $K$ -sparse vectors  $\mathbf{f}'$ .

A matrix  $\mathbf{A}$  obeys the RIP of order  $K$  if  $\delta_K$  is not too close to one. When this property holds,  $\mathbf{A}$  approximately preserves the Euclidean length of the  $K$ -sparse signals. This implies that the  $K$ -sparse vectors cannot be in the null space of  $\mathbf{A}$  and so Equation (7.1.3) has a solution. An equivalent description of the RIP is to say that all subsets of  $K$  columns taken from  $\mathbf{A}$  are in fact nearly orthogonal. if the RIP holds, the linear program

$$\min \|\mathbf{f}'\|_1 \quad \text{subject to} \quad \mathbf{z} = \mathbf{A}\mathbf{f}' \quad (7.1.4)$$

recovers  $\mathbf{f}'$  accurately. A stronger theorem expressed in [Candès et al. ] deals with not only  $K$ -sparse signals, but all signals.

**Theorem 7.1.2.** *Assuming that  $\delta_{2K} < \sqrt{3} - 1$ , then the solution  $\mathbf{f}'^*$  to Equation 7.1.4 obeys*

$$\|\mathbf{f}'^* - \mathbf{f}'\|_2 \leq C_0 \frac{\|\mathbf{f}' - \mathbf{f}'_K\|_1}{\sqrt{K}} \text{ and } \|\mathbf{f}'^* - \mathbf{f}'\|_1 \leq C_0 \|\mathbf{f}' - \mathbf{f}'_K\|_1$$

for some constant  $C_0$ , where  $\mathbf{f}'_K$  is the vector  $\mathbf{f}'$  with all but the largest  $K$  components set to 0.

If  $\mathbf{f}'$  is not  $K$ -sparse, the above theorem asserts that the quality of the recovered signal is determined by the degree to which the locations of the  $K$  largest values of  $\mathbf{f}'$  are known in advance and those  $K$  values can be measured directly. What is more, if the matrix  $\mathbf{A}$  obeys the hypothesis of the theorem, there will be recovery of all sparse  $K$ -vectors exactly, and essentially the  $K$  largest entries of all vectors, with no probability of failure.

When noisy data is involved in Equation (7.1.3),  $L_1$ -minimisation with inequality constraints is used to recover  $\mathbf{f}'$ :

$$\min \|\mathbf{f}'\|_1 \quad \text{subject to} \quad \|\mathbf{A}\mathbf{f}' - \mathbf{z}\|_2 \leq \epsilon \quad (7.1.5)$$

where  $\epsilon$  bounds the amount of noise in the data. This can be efficiently solved by a second-order cone program [Candès et al. ]. The equivalent theorem for noisy data is then outlined by:

**Theorem 7.1.3.** *Assuming that  $\delta_{2K} < \sqrt{3} - 1$ , then the solution  $\mathbf{f}'^*$  to Equation (7.1.5) obeys*

$$\|\mathbf{f}'^* - \mathbf{f}'\|_2 \leq C_0 \frac{\|\mathbf{f}' - \mathbf{f}'_K\|_1}{\sqrt{K}} + C_1 \epsilon$$

for some constants  $C_0$  and  $C_1$ .

This shows that small perturbations in the data only cause small perturbations in the reconstruction. Therefore Compressive Sensing is robust to noisy data and can be applied in practical settings.

### 7.1.5 RIP Random Sensing

One of the most powerful results from Compressive Sensing is that sensing matrices obeying the RIP with values of  $K$  close to  $M$  can be determined in a random manner. Consider the following sensing matrices: i)  $\mathbf{A}$  formed by sampling  $n$  column vectors

uniformly at random on the unit sphere of  $R^M$ ; ii)  $\mathbf{A}$  formed by sampling i.i.d. entries from the normal distribution with mean 0 and variance  $1/m$ ; iii)  $\mathbf{A}$  formed by sampling a random projection  $\mathbf{P}$  normalised by  $\mathbf{A} = \mathbf{P}\sqrt{N/M}$ ; and iv)  $\mathbf{A}$  formed by sampling i.i.d. entries from a symmetric Bernoulli distribution ( $\mathbf{P}(\mathbf{A}_{i,j} = \pm 1) = 1/2$ ) or other sub-Gaussian distribution. With overwhelming probability, all these matrices obey the RIP provided that

$$M \geq CK \log N/K \quad (7.1.6)$$

where  $C$  is some constant depending on the instance. In all the above cases i)-iv), the probability of sampling a matrix not obeying the RIP when (7.1.6) holds is exponentially small in  $M$ . Therefore, if fixing  $\Phi$  and populating  $\Psi$  as in i)-iv), then with overwhelming probability, the matrix  $\mathbf{A} = \Phi\Psi$  obeys the RIP provided that (7.1.6) is satisfied, where again  $C$  is some constant depending on the instance. These random measurement matrices  $\Phi$  are in a sense universal [Baraniuk et al. 2008] and the sparsity basis need not even be known when designing the measurement system. In some encoding/decoding applications, the sparse basis  $\Psi$  may be unknown at the encoder or impractical to implement for data compression. A randomly designed  $\Phi$  can be considered a universal encoding strategy, as it need not be designed with regards to the structure of  $\Psi$ . The knowledge and ability to implement  $\Psi$  are required only for the recovery of  $\mathbf{f}$  at the decoding side. This universality may be particularly helpful for distributed source coding in multi-signal settings such as sensor networks.

## 7.2 Discrete Wavelet Transform

While the field of Discrete Wavelet Transforms (DWT) is too large to present in its entirety, below we give a very brief review on DWT in order to help reveal an intuitive connection between multilevel/multiresolution wavelets and the coarse-to-fine nature of the annealing schedule in the next section. For more details and information, the reader is encouraged to refer to [Mallat 1989; Mallat 1999].

A wavelet, in the sense of the Discrete Wavelet Transform, is an orthogonal function which can be applied to a finite group of data. Functionally, it is very much like the Discrete Fourier Transform, in that the transforming function is orthogonal, a signal passed twice through the transformation is unchanged, and the input signal is assumed to be a set of discrete-time samples. Both transforms are convolutions. The DWT of a signal  $x$  is recursively calculated by passing it through a series of filters. Filtering a signal corresponds to the mathematical operation of convolution of the signal with the impulse response  $f$  of the filter. Let us suppose that  $x[n]$  is the original signal, spanning a frequency band of 0 to  $\pi$  radians. The convolution operation in discrete time is defined as follows:

$$y[n] = (x * g)[n] = \sum_k x[k]f[n - k]$$

The DWT analyses the signal at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and information on the details. DWT employs two sets of functions, called scaling functions and wavelet functions, which are associated with low- and highpass filters, respectively. The decomposition of the signal into different frequency bands is simply obtained by successive high- and lowpass filtering of the time domain signal. The original signal  $x[n]$  is first passed through a halfband highpass filter  $g[n]$  and a lowpass filter  $h[n]$ . After the filtering, half of the samples can be eliminated according to Nyquist's rule, since the signal now has a highest frequency of  $\pi/2$  radians instead of  $\pi$ . The signal can therefore be subsampled by 2, simply by discarding every other sample. Note, however, that the subsampling operation after filtering does not affect the resolution, since removing half of the spectral components from the signal makes half of the samples redundant and they can be discarded without any loss of information. This consti-

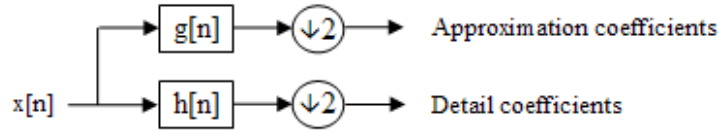


Figure 7.1: Block diagram of one level of DWT decomposition, with the subsampling operator  $\downarrow$  (courtesy of Wikipedia [Wikipedia 2011])

tutes one level of decomposition and can mathematically be expressed as follows:

$$y_{high}[n] = \sum_k x[k]g[2n - k]$$

$$y_{low}[n] = \sum_k x[k]h[2n - k]$$

where  $y_{high}[n]$  and  $y_{low}[n]$  are approximation and detail coefficients – the outputs of the highpass and lowpass filters, respectively – after subsampling by 2. As illustrated in Figure 7.1, the 1D-DWT produces a pyramidal decomposition of a given input signal into different resolution bands. Each level generates a pair of approximation and detail signals from the approximation band of the previous level. As their names suggest, approximation is a coarse-grained representation of its predecessor, and detail contains the high-frequency details that have been removed. Both of them have half the resolution of their predecessor. This procedure can be repeated for processing the next level of the decomposition.

The 2D-DWT is usually obtained by applying a separate 1D transform along each dimension. The most common approach, known as the square decomposition, alternates between computations on image rows and columns. This process is applied recursively to the quadrant containing the coarse scale approximation in both directions. This way, the data on which computations are performed is reduced to a quarter in each step. As shown in Figure 7.2, at each step we decompose the  $j - 1$  level approximation coefficients  $A_{j-1}$  (for  $j > 0$  – the original image is regarded as level 0) into four wavelet subbands  $A_j$ ,  $D_{j,1}$ ,  $D_{j,2}$ , and  $D_{j,3}$ . We first convolve the rows of  $A_j$

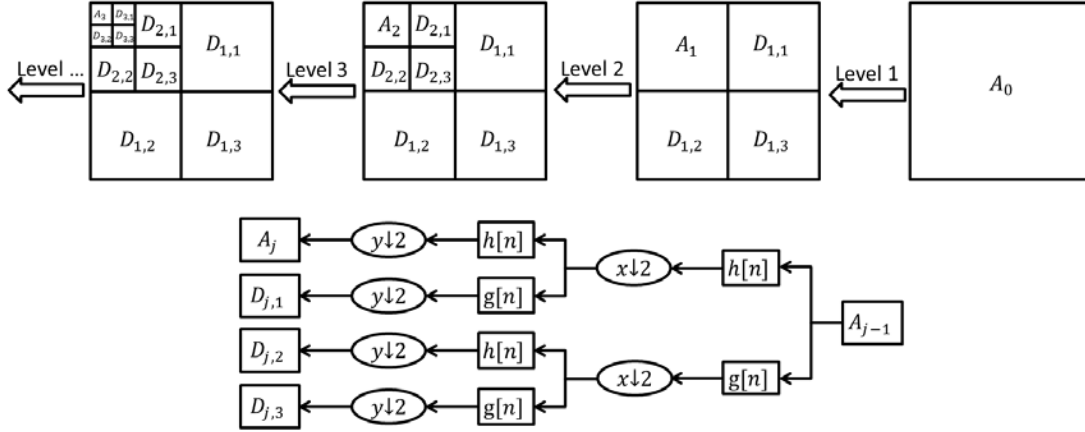


Figure 7.2: The Pyramid structure of DWT decomposition of a 2D image signal

with a one dimensional lowpass  $h[n]$  and highpass  $g[n]$  filter, subsample by 2, convolve the columns of the resulting signals with another one dimensional lowpass  $h[n]$  and highpass  $g[n]$  filter and subsample by 2. We compute the wavelet transform of an image  $A_0$  by repeating this process until the desired level is reached.

The Haar wavelet decomposition is the simplest of the wavelet decompositions. Each step in the Haar decomposition calculates a set of wavelet difference coefficients and a set of average coefficients. If a data set  $s_0, s_1, \dots, s_{N-1}$  contains  $N$  elements, there will be  $N/2$  averages and  $N/2$  coefficient values. The averages are stored in the lower half of the  $N$  element array and the coefficients are stored in the upper half. The averages become the input for the next step in the wavelet calculation, where for iteration  $i + 1$ ,  $N_{i+1} = N_i/2$ . The recursive iterations continue until a single average and a single difference coefficient are calculated. This replaces the original data set of  $N$  elements with an average, followed by a set of difference coefficients whose size is an increasing power of two (e.g.,  $2^0, 2^1, 2^2 \dots N/2$ ).

### 7.3 Compressive Annealed Particle Filter

Recalling the APF Framework proposed in Chapter 3, the pose space model needs to be optimised in a dynamic structure that consists of a sequence of estimate poses  $\mathbf{x}_t$  at



successive times  $t = 1, 2, \dots$ , where each pose is associated with an image observation  $\mathbf{y}_t^{obs}$ . In this section, we derive an observation likelihood in the compressive domain. At time  $t$ , the compressive measurement  $\mathbf{z}_t^d$  can be defined by:

$$\begin{aligned}\mathbf{z}_t^d &= \Phi \Psi \mathbf{y}_t^d \\ &= \Phi \Psi (\mathbf{y}_t^{obs} - \mathbf{y}_t^{bg}) \\ &= \mathbf{z}_t^{obs} - \mathbf{z}_t^{bg}\end{aligned}\tag{7.3.1}$$

where  $\Psi$  denotes the wavelet basis. In particular,  $\mathbf{y}_t^d$  is the difference image generated by subtracting the background image  $\mathbf{y}_t^{bg}$  from the original observation image  $\mathbf{y}_t^{obs}$ . It is known that images acquired from natural scenes have highly sparse representations in the wavelet domain. The difference image calculated by subtracting the static background from the observation image has more pixel values close to zero, hence, the difference image  $\Psi \mathbf{y}_t^d$  is also highly sparse and compressible in general.

On the other hand, given the estimate state  $\mathbf{x}_t$ , the estimate compressive measurement  $\hat{\mathbf{z}}_t^d$  of the difference image can be calculated by subtracting the background image  $\mathbf{y}_t^{bg}$  from a synthetic foreground image  $s^{fg}(\mathbf{x}_t)$ , which is generated by projecting a human model with pose  $\mathbf{x}_t$  and camera parameters onto the image plane. This difference image is also compressible in the wavelet domain, and it can be defined by:

$$\begin{aligned}\hat{\mathbf{y}}_{t,i}^d &= \text{sil}_i(\mathbf{x}_t) * (s_i^{fg}(\mathbf{x}_t) - \mathbf{y}_{t,i}^{bg}) \quad i = 1, \dots, N \\ \hat{\mathbf{z}}_t^d &= \Phi \Psi \hat{\mathbf{y}}_t^d\end{aligned}\tag{7.3.2}$$

where,  $\text{sil}(\mathbf{x}_t)$  is a synthetic silhouette mask generated by the estimate state  $\mathbf{x}_t$  which has 0s on all background entries and 1s on all the foreground entries. This mask operation is used to make the synthetic difference image comparable to the original difference image.

### 7.3.1 Restricted Isometry Property and Pairwise Distance Preservation

Another important result of CS is the Restricted Isometry Property (RIP) [Candes and Tao 2005] which characterises the stability of nearly orthonormal measurement matrices. A matrix  $\Phi$  satisfies RIP of order  $K$  if there exists an isometry constant  $\sigma_K \in (0, 1)$  as the smallest number, such that  $(1 - \sigma_K)\|\mathbf{f}'\|_2^2 \leq \|\Phi\mathbf{f}'\|_2^2 \leq (1 + \sigma_K)\|\mathbf{f}'\|_2^2$  holds for all  $\mathbf{f}' \in \Sigma_K = \{\mathbf{f}' \in \mathbb{R}^N : \|\mathbf{f}'\|_0 \leq K\}$ . In other words,  $\Phi$  is an approximate isometry for signals restricted to be  $K$ -sparse and approximately preserves the Euclidean length, interior angles and inner products between the  $K$ -sparse signals. This reveals the reason why CS recovery is possible because  $\Phi$  embeds the sparse signal set  $\Sigma_K$  in  $\mathbb{R}^M$  while no two sparse signals in  $\mathbb{R}^N$  are mapped to the same point in  $\mathbb{R}^M$ . Recently, Baron et al. in [Baron et al. 2009] have revealed how the relationship between the sparsity level  $K$  and the number of measurements  $M$  affects the approximate isometry properties of  $\Phi$ . One of the important theorems gives conditions that guarantee isometry properties of  $\Phi$ :

**Theorem 7.3.1.** [Baron et al. 2009] *If  $\Phi$  has i.i.d. Gaussian entries and  $M \geq 2K$ , then, with probability one, there always exists  $\sigma_{2K} \in (0, 1)$  such that all pair-wise distances between  $K$ -sparse signals are well preserved:*

$$(1 - \sigma_{2K}) \leq \frac{\|\Phi\mathbf{f}'_i - \Phi\mathbf{f}'_j\|_2^2}{\|\mathbf{f}'_i - \mathbf{f}'_j\|_2^2} \leq (1 + \sigma_{2K}). \quad (7.3.3)$$

Straightforward proof [Baron et al. 2009] can be outlined as follows:

First, if  $K \geq N/2$ , then with probability one, the matrix  $\Phi$  has rank  $N$ , and there is a unique projection for the specific signal. Thus we assume that  $K < N/2$ . With probability one, all subsets of up to  $2K$  columns drawn from  $\Phi$  are linearly independent. Assuming this holds, we construct two arbitrary  $K$ -column subsets  $\Phi_1$  and  $\Phi_2$ , where  $\Phi_1 \neq \Phi_2$ . Then we form a subspace  $\Omega$  by  $\text{colpan}(\Phi_1) \cap \text{colpan}(\Phi_2)$ , which has dimension equal to the number of columns common to both  $\Phi_1$  and  $\Phi_2$ . A  $K$ -sparse signal  $\mathbf{f}'$  projects to this common space, only if its coefficients are nonzero on exactly

these (fewer than  $K$ ) common columns; Since  $\|\mathbf{f}'\|_0 = K$ , this does not occur. Thus every  $K$ -sparse signal projects to a unique point in  $R^M$ . When  $K < M < 2K$ , there will necessarily exist  $K$ -sparse signals that cannot be uniquely projected, However, these signals form a set of measure zero within the set of all  $K$ -sparse signals and can safely be avoided with high probability if  $\Phi$  is randomly generated independently of  $\mathbf{f}'$ . When  $M \leq K$ ,  $K$ -sparse signals projected can not be uniquely identified in  $R^M$ .

On the other hand, the classic Johnson-Lindenstrauss (JL) lemma [Johnson and Lindenstrauss 1984] asserts that any set of  $n$  points in  $d$ -dimensional Euclidean space can be embedded into  $k$ -dimensional Euclidean space, where  $k$  is logarithmic in  $n$ ,  $O(\log n/\epsilon^2)$  and independent of  $d$ —so that all pairwise distances are maintained by a factor of  $1 \pm \epsilon$ , for any  $0 < \epsilon < 1$ . In [Baraniuk and Wakin 2009], Baraniuk and Wakin present a JL lemma formulation with stable embedding of a finite point cloud under a random orthogonal projection, which has a tighter lower bound for  $M$ :

**Lemma 7.3.2.** [Baraniuk and Wakin 2009] Let  $\mathbb{Q}$  be a finite collection of points in  $\mathbb{R}^N$ . Fix  $0 < \sigma < 1$  and  $\beta > 0$ . Let  $\Phi \in \mathbb{R}^{M \times N}$  be a random orthogonal matrix and

$$M \geq \left( \frac{4 + 2\beta}{\sigma^2/2 + \sigma^3/3} \right) \ln(\#\mathbb{Q})$$

If  $M \leq N$ , then, with probability exceeding  $1 - (\#\mathbb{Q})^{-\beta}$ , the following statement holds: for every  $\mathbf{f}'_i, \mathbf{f}'_j \in \mathbb{Q}$  and  $i \neq j$

$$(1 - \sigma) \sqrt{\frac{M}{N}} \leq \frac{\|\Phi \mathbf{f}'_i - \Phi \mathbf{f}'_j\|_2}{\|\mathbf{f}'_i - \mathbf{f}'_j\|_2} \leq (1 + \sigma) \sqrt{\frac{M}{N}}$$

where a random orthogonal matrix can be constructed by performing the Householder transformation [Householder 1958] on  $M$  random length- $N$  vectors having i.i.d. Gaussian entries, assuming the vectors are linearly independent.

### 7.3.2 Multilevel Wavelet Likelihood Evaluation on Compressive Measurements

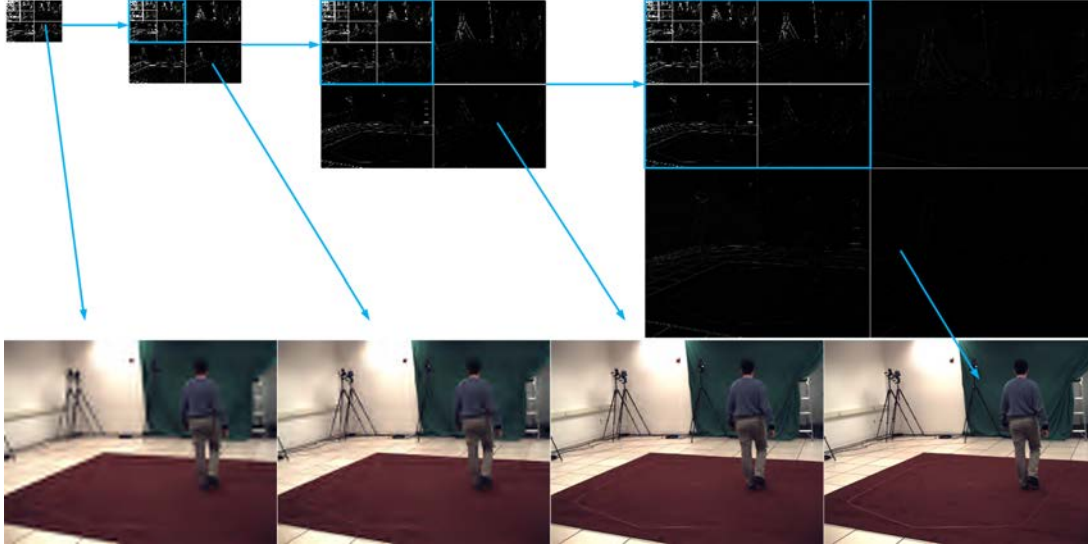


Figure 7.3: The number of wavelet coefficients is progressively elevated in the wavelet decomposition process so that details are gradually enhanced through the annealing schedule. At the top of the figure, we show 4 levels of wavelet decomposition coefficients, from left to right: 1) At level 4, using only the  $K_4 = 2805$  largest coefficients (about 18.39% of all level 4 coefficients), 2)  $K_3 = 4345$  (7.18%) at level 3, 3)  $K_2 = 12086$  (5.01%) at level 2 and 4)  $K_1 = 30000$  (3.11%) at level 1. The observation images at the bottom are reconstructed by using corresponding  $K_g$  sparse wavelet coefficients.

Theorem (7.3.1), Lemma (7.3.2) and orthonormality of  $\Psi$  guarantee pairwise distance to be approximately preserved provided that  $M$  is sufficient large. Therefore CS recovery is not necessary to evaluate the observation likelihood. Instead, the observation likelihood can be directly calculated via distances in compressive measurements using Equations (7.3.1) and (7.3.2).

$$p(\mathbf{y}_t | \mathbf{x}_t) = \exp\{-\lambda \|\mathbf{z}_t^d - \hat{\mathbf{z}}_t^d\|_1\} \quad (7.3.4)$$

Notice for  $\lambda > 0$ , the above equation can be transformed:

$$\begin{aligned} p(\mathbf{y}_t|\mathbf{x}_t) &= \exp\{-\|\lambda\mathbf{z}_t^d - \lambda\hat{\mathbf{z}}_t^d\|_1\} \\ &= \exp\{-\|\Phi\lambda(\Psi\mathbf{y}_t^d - \Psi\hat{\mathbf{y}}_t^d)\|_1\} \end{aligned} \quad (7.3.5)$$

where  $\Psi\mathbf{y}_t^d$  and  $\Psi\hat{\mathbf{y}}_t^d$  are wavelet coefficients. The ultimate purpose is to substitute  $\lambda(\Psi\mathbf{y}_t^d - \Psi\hat{\mathbf{y}}_t^d)$  by a series of coarse-to-fine wavelet coefficient images, which conform to gradually elevating  $\lambda$  in the annealing schedule, so that the computation of the observation likelihood is effectively reduced and data modification is intuitively integrated with the annealing schedule. More concretely, we expect to construct a series of  $\Psi_l(\mathbf{y}_t^d)$  and  $\Psi_l(\hat{\mathbf{y}}_t^d)$ :

$$p(\mathbf{y}_t|\mathbf{x}_t) = \exp\{-\|\Phi_l(\Psi_l(\mathbf{y}_t^d) - \Psi_l(\hat{\mathbf{y}}_t^d))\|_1\} \quad (7.3.6)$$

where  $\Psi_l(\mathbf{y}_t^d)$  represents the wavelet coefficients of  $\mathbf{y}_t^d$  at the  $l$  layer associated with the level  $g$  decomposition, having  $N_l$  wavelet coefficients. Where  $l$  is increasing,  $g$  is decreasing and more details encoded in the wavelet coefficients  $\Psi_l(\mathbf{y}_t^d)$  are used.  $\Phi_l$  is an  $M_l \times N_l$  sub-matrix of  $\Phi$ .  $M_l = 2K_g$  is determined according to the sparsity level  $K_g$  of the  $g$  level wavelet coefficients.

### 7.3.2.1 Construct Increasing Wavelet Coefficient Image

According to the multilevel wavelet decomposition in Section 7.2, any image can be DWT decomposed to wavelet coefficients (e.g. shown in the top of Figure 7.3. To simplify the following procedure, we add a large positive constant  $\zeta$  to the wavelet coefficients such that all wavelet coefficients are always positive. This is possible due to the fact that the image pixel value is bounded and the wavelet decomposition can be finished in a finite number of steps, and the positive constant  $\zeta$  is maintained. Note that adding a large positive constant  $\zeta$  to both  $\Psi\mathbf{y}_t^d$  and  $\Psi\hat{\mathbf{y}}_t^d$  in Equation (7.3.5) does

not change the value of the observation likelihood evaluation. The overall optimisation remains valid.

We begin by constructing two wavelet coefficient sequences  $\mathbf{C} = \{\mathbf{c}_i | \mathbf{c}_i > 0, i = 1, 2, \dots\}$  and  $\hat{\mathbf{C}} = \{\hat{\mathbf{c}}_i | \hat{\mathbf{c}}_i > 0, i = 1, 2, \dots\}$  for  $\Psi \mathbf{y}_t^d$  and  $\Psi \hat{\mathbf{y}}_t^d$  in Equation (7.3.5). Sequence  $\mathbf{C}$  is the sub-quarter iteration of the multilevel wavelet decomposition as shown in the top of Figure 7.3 from left to right. The current level wavelet coefficients are always a subset of the super level wavelet coefficients,  $\mathbf{c}_i \subset \mathbf{c}_{i+1}$ . Hence,  $\|\mathbf{c}_i\|_1 < \|\mathbf{c}_{i+1}\|_1$  and  $\mathbf{C}$  is considered a monotonically increasing sequence in terms of magnitude (the same can be applied to  $\hat{\mathbf{C}}$ ). Since the components of  $\mathbf{C}$  and  $\hat{\mathbf{C}}$  are positive and monotonically increasing in their magnitude, it is easy to induce that  $\mathbf{C}^\Delta = \mathbf{C} - \hat{\mathbf{C}}$  has the same monotonically increasing property  $\|\mathbf{c}_i^\Delta\|_1 < \|\mathbf{c}_{i+1}^\Delta\|_1$ . If we define an increasing sequence of variables as  $\{\lambda_i | \lambda_i = \|\mathbf{c}_{i+1}^\Delta\|_1 / \|\mathbf{c}_1^\Delta\|_1, \lambda_i < \lambda_{i+1}, i = 1, 2, \dots\}$ , then the monotonically increasing sequence  $\mathbf{C}^\Delta$  can be described by  $\mathbf{C}^\Delta = \{\mathbf{c}_1^\Delta, \lambda_1 \mathbf{c}_1^\Delta, \lambda_2 \mathbf{c}_1^\Delta, \dots\}$ . In other words, we can always construct a monotonically increasing wavelet coefficient sequence  $\mathbf{C}^\Delta$  that can absorb the counterpart series of  $\lambda$  and simulate the  $\lambda$  effect on the data domain directly. The direct consequence of this is that Equation (7.3.6) is valid. The precise value of  $\lambda$  for each annealing layer is not critical, since  $\lambda$  is only used to roughly control the optimisation convergence rate. Therefore, we design direct evaluation of the coarse-to-fine wavelet coefficients in different levels to simulate increasing  $\lambda_l$  at each layer  $l$ .

## 7.4 Experiments

Experiments were conducted on the benchmark dataset HumanEvaII [Sigal and Black 2006a] that contains two 1260-frame image sequences from 4 calibrated colour cameras synchronised with Mocap data at 60Hz. The tracking subjects perform three different actions: walking, jogging and balancing. To generate compressive measurements, we apply the 8-level Haar wavelet 2D decomposition [Daubechies 1992]

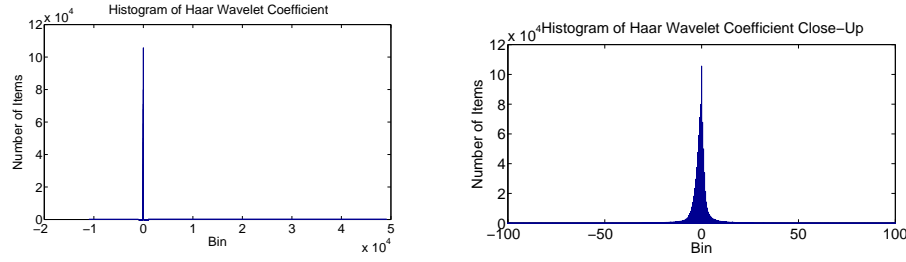


Figure 7.4: Wavelet Coefficient Histogram (left) close-up view (right) showing that 95% of coefficients have very small values close to zero.

to all observation images. The wavelet coefficients appear highly sparse, most of them close to zero as illustrated in Figure 7.4. For instance, using solely the 30000 largest wavelet coefficients we have been able to reconstruct the 964320 colour components of a  $656 \times 490$  RGB image with scarcely noticeable perceptual loss. For the multilevel evaluation (Equation 7.3.6), the four sparsity levels  $K_1 = 30000$ ,  $K_2 = 12086$ ,  $K_3 = 4345$  and  $K_4 = 2805$  are evenly allocated in the 10 anneal layers<sup>1</sup>. The  $M_l = 2K_g$  rows of  $\Phi$  are drawn i.i.d. from the normal distribution  $N(0, 1/M_l)$  to approximately preserve isometry as shown in Equation (7.3.3). On the other hand, the single level evaluation Equation (7.3.4) is used with the tight lower bound for  $M$  shown in Lemma (7.3.2). We presume there is one observation image and maximum  $2000^2$  synthetic images generated in the evaluation of each view and each frame. Then, for the 1260-frame sequence, there are a total of 2521260 unique compressive measurements required for tracking. Let  $\sigma = 0.1$ ,  $\beta = 1$  and  $\#\mathbb{Q} = 2521260$ , so  $M = \left(\frac{4+2\beta}{\sigma^2/2+\sigma^3/3}\right) \ln(\#\mathbb{Q}) = 16583$ . Moreover, the  $M$  rows of  $\Phi$  are constructed by drawing i.i.d. entries from the normal distribution  $N(0, 1/M)$  and performing the Householder transformation to orthogonalise  $\Phi$ . Therefore, with high probability  $1 - 1/2521260$ ,  $\Phi$  approximately preserves the pairwise distances. We also verified the performance of the number of compressive measurements in cases of  $M = 10000$  and  $M = 5000$ .

<sup>1</sup>using  $M_1 = 2 \times 2805$ ,  $M_2 = 2 \times 2805$ ,  $M_3 = 2 \times 2805$ ,  $M_4 = 2 \times 4345$ ,  $M_5 = 2 \times 4345$ ,  $M_6 = 2 \times 4345$ ,  $M_7 = 2 \times 12086$ ,  $M_8 = 2 \times 12086$ ,  $M_9 = 2 \times 30000$ ,  $M_{10} = 2 \times 30000$

<sup>2</sup>Given 10 layers and 200 particles as the maximum

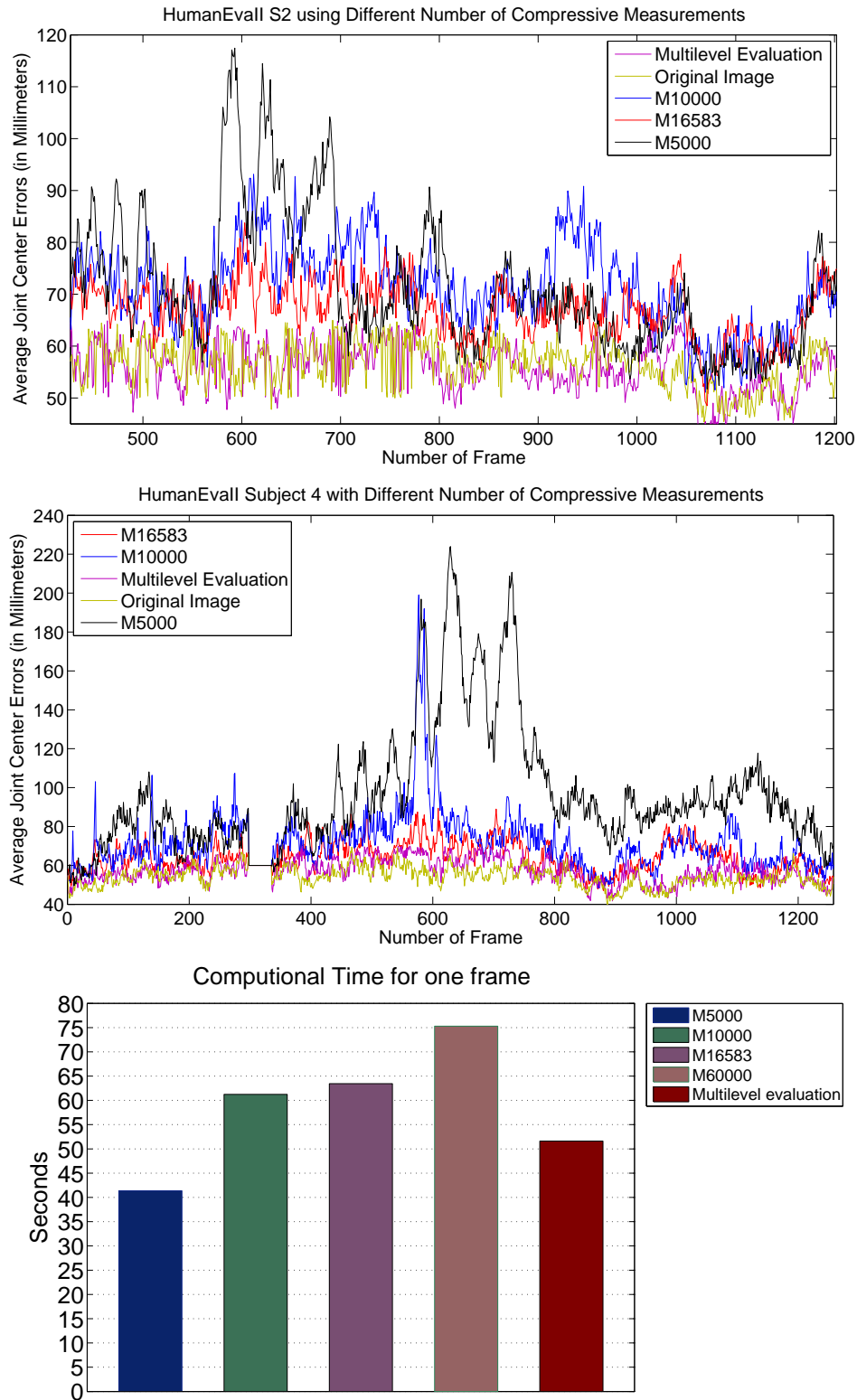


Figure 7.5: From top to bottom, 1) tracking results of HumanEvaII Subject 2, 2) tracking results of HumanEvaII Subject 4 (note that the ground truth data is corrupted in frames 298-335) and 3) computational time for one frame using different numbers of compressive measurements.



As illustrated in the experimental results of HumanEvaII Subject 2 (the top of Figure 7.5), the evaluation using the original images as input obtains  $54.5837 \pm 4.7516mm^3$ . The multilevel evaluation achieves the stable result  $56.9442 \pm 4.4581mm$  which is comparable with the result using original images. When using single level evaluation with  $M = 16583$  compressive measurements, the tracking performance appears poorer than with multilevel evaluation but still maintains within  $65.7548 \pm 5.4351mm$ . When the number of compressive measurements are further reduced to  $M = 10000$  and  $M = 5000$ , the performance is degraded dramatically and we obtain  $70.4249 \pm 7.5613mm$  and  $68.2124 \pm 11.6153mm$ , respectively. The middle of Figure 7.5 shows the experimental results with HumanEvaII Subject 4. The evaluation using original images achieves  $54.2207 \pm 4.9250mm$  which is slightly better than  $57.1705 \pm 6.0227mm$  achieved by the multilevel evaluation. Using  $M = 16583$  compressive measurements results in slightly more fluctuations compared with the results of Subject 2. When the number of compressive measurements is decreased to  $M = 10000$  and  $M = 5000$ , there are significant mistrackings and drifts with larger errors, giving  $71.6053 \pm 15.4005mm$  and  $96.3663 \pm 32.8075mm$ . More visual tracking results are shown in Figure 7.6.

The computational performance is also evaluated via the computational time for one frame using the different numbers of compressive measurements shown in the bottom of Figure 7.5. As expected, computational times ranging from 40 to 75 seconds roughly correspond to increasing the number of the compressive measurements  $M$ . On the other hand, the multilevel evaluation is able to reach a level of computational speed similar to using fewer compressive measurements. Overall, the utilisation of progressive coarse-to-fine multilevel evaluation allows our approach to achieve a computational efficiency comparable to using only  $M = 10000$  compressive measurements, maintaining tracking accuracy comparable to using the original images.

---

<sup>3</sup>The results are statistically presented by mean  $\pm$  standard deviation in millimetres



Figure 7.6: HumanEvaII visual tracking results of Subject 4 and 2 are depicted in the top four rows and the bottom four rows, respectively. A transparent visual model is overlaid on the tracking subject.

---

# Conclusion

---

To summarise, this work begins with a comprehensive literature review on the generative approach, learning based approaches and tracking via graphic based image segmentation. Then it outlines a general architecture with basic components comprising the human body model, observation likelihood, temporal dynamical model and optimisation on pose estimation, as well as how these components are incorporated into the sequential dynamical framework to realise a human motion capture system. The work is then unfolded based on these primary components: In Chapter 4, the generic articulated skeleton is described as well as how to parameterise and optimise 3D rotation joints. Two shape parametrisations, needle based and data driven, are then described. The chapter ends with the novel automatic initialisation method for markerless motion capture. In Chapter 5, the focus is put on nature-inspired methods: Simulated Annealing, Particle Swarm Optimisation and Covariance Matrix Adaptation Evolution Strategy as well as our novel Covariance Matrix Adaptation Annealing. Subsequent experiments with four benchmark optimisation problems demonstrate the efficiency of our approach compared to other methods. Chapter 6 describes several strategies to improve the robustness of the observation likelihood evaluation, including Incremental Relaxation by Fast March Method, Colour and Texture Incorporation, Maximisation of Mutual Information and Gradual Sampling for Annealed Particle Filter. In Chapter 7, a novel compressive sensing technique with multilevel wavelet decomposition is seamlessly integrated into Simulated Annealing to exploit the sparsity level in observed images in the coarse-to-fine fashion.

We have presented a pretty comprehensive study on markerless motion capture from a broad perspective. Methods presented in this work have many advantages but also have limitations and conditions on their use. These are outlined as follows:

1. **Subject Specific Body Shape Modelling and Automatic Initialisation:** In general, more information gathered results in more accurate tracking, and this method serves this purpose very well. The most accurate tracking requires a reasonable amount of initial calibrations. Our approach favours the standard graphics skeleton parameterisation and dimensionality reduced data-driven shape parameterisation. This is very flexible and suitable for most circumstances, but places a high demand on the accuracy of input data. For instance, it requires relatively precise silhouettes and synchronised multiview images covering most perspectives. If these early calibrations can not be accomplished reasonably well, subsequent tracking may suffer mistracking.
2. **Nature Inspired Global Optimisation:** This is a very general and powerful method for solving high dimensional and multimodal optimisation problems in markerless motion capture. This kind of approach is recognised as a good way, possibly because markerless motion capture is too complex for researchers to develop regular patterns and principles, and therefore cannot be efficiently solved by existing/common methods. However, this kind of approach may just be a temporary good solution. The major drawback of such stochastic optimisation is the convergence speed. This drawback severely limits its practical usage in real world settings. Nevertheless, with the exponential growth of computational power, there is an increasing possibility to apply stochastic optimisation to medium sized practical problems in the near future.
3. **Robust Evaluation and Compressive Evaluation:** From our experience, robust evaluation leads optimisation to converge to the desired global optimum. However, optimisation could end up with local optimums or a corrupted global op-

---

timum. Thus, modelling the energy function is crucial for successful tracking. This evaluation is often the main computational bottleneck, and an overly sophisticated evaluation will undoubtedly damage overall tracking performance. Our approach used a coarse-to-fine order in the annealing schedule to avoid wasting computations on premature convergence. The price of this approach is that it requires extra processing on the input data.

Author's publications:

1. Yifan Lu, Lei Wang, Richard Hartley, Hongdong Li and Dan Xu, "Compressive evaluation in human motion tracking," ACCV, 2010.
2. Yifan Lu, Lei Wang, Richard Hartley, Hongdong Li and Dan Xu, "Gradual sampling and mutual information maximisation for markerless motion capture," ACCV, 2010.
3. Yifan Lu, Dan Xu, Lei Wang, Richard I. Hartley, Hongdong Li, "Illumination invariant sequential filtering human tracking," ICMLC 2010: 2133-2138.
4. Yifan Lu, Lei Wang, Richard Hartley, Hongdong Li and Chunhua Shen, "Multi-view human motion capture with an improved deformation skin model," DICTA08, 2008, pp. 420 – 427.
5. Yifan Lu, "Markerless human motion capture: An application of simulated annealing and Fast Marching Method," ICPR 2008: 1-4.



# Appendix

## A.1 Perspective Projection

Considering a three dimensional coordinate system as shown in Figure A.1, the origin is at the centre of projection and the  $Z$  axis is along the optical axis. This coordinate system is called the standard coordinate system of the camera. A point  $\mathbf{M}$  on an object with coordinates  $(X, Y, Z)$  will be imaged at some point  $\mathbf{m} = (x, y)$  on the image plane. The coordinate of  $\mathbf{m}$  is with respect to a coordinate system whose origin is at the intersection of the optical axis and the image plane, and the  $x$  and  $y$  axes are parallel to the  $X$  and  $Y$  axes. For an ideal camera model without radial distortions, providing the intrinsic and extrinsic parameters of the camera calibration are known, the camera frame rotation  $\mathbf{R}$ , the camera frame translation  $\mathbf{t}$ , the focal length  $(f_x, f_y)$  in both orientations and the principle point  $(u_c, v_c)$ , the pipeline of the camera perspective projection can be formulated as:

$$\begin{aligned} s \begin{pmatrix} \mathbf{m} \\ 1 \end{pmatrix} &= \begin{pmatrix} f_x & 0 & u_c \\ 0 & f_y & v_c \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{M} \\ 1 \end{pmatrix} \\ &= \mathbf{K} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \mathbf{T} \begin{pmatrix} \mathbf{M} \\ 1 \end{pmatrix} \end{aligned}$$

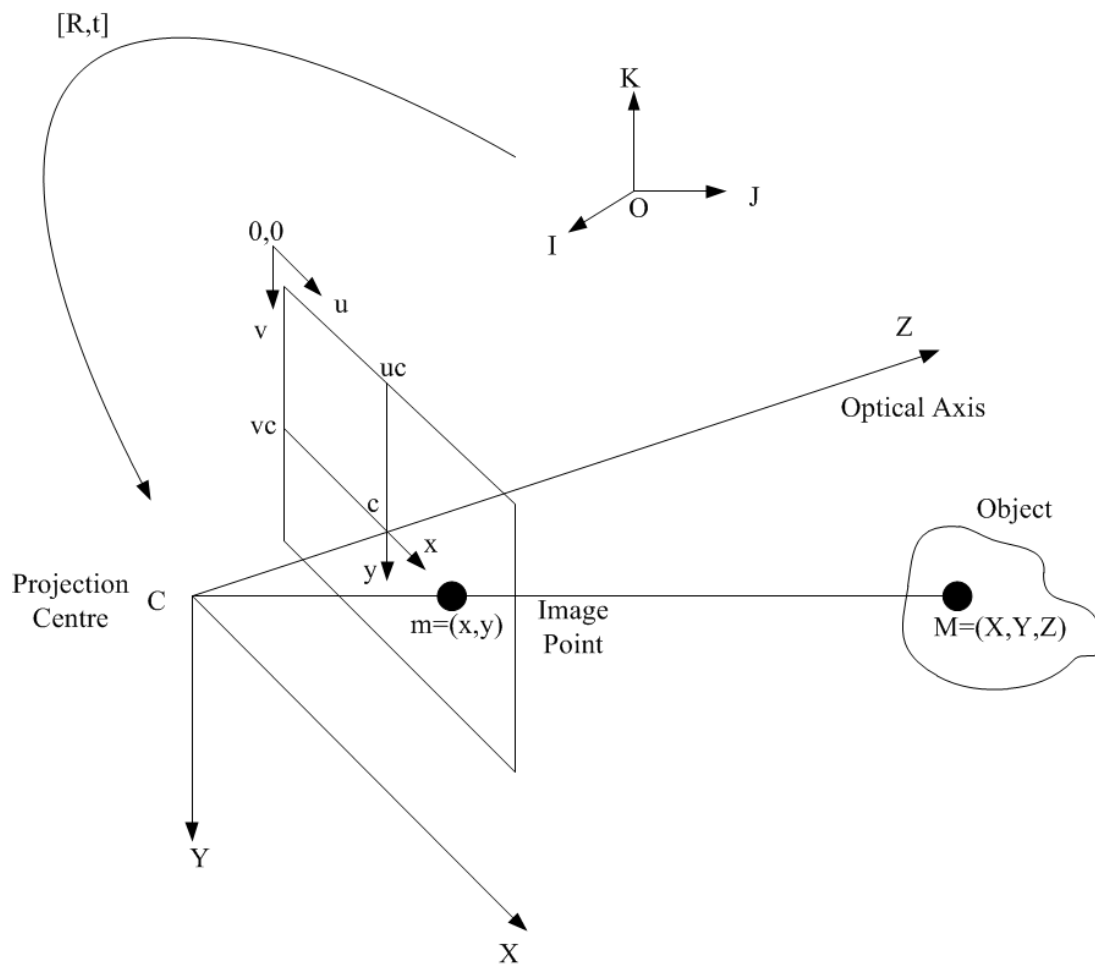


Figure A.1: Camera Calibration Coordinate System



where,  $s$  is the scale factor of the homogeneous pixel coordinate, having value  $Z$ .  $\mathbf{K}$  contains the intrinsic parameters, and  $\mathbf{T}$  contains the transformation parameters.

For the camera model with lens distortions  $\mathbf{D}_c$  and skew coefficient  $\alpha_c$  defining the angle between the  $x$  and  $y$  pixel, the point  $M$  is first transformed into the camera reference frame by  $\mathbf{M}_c = \mathbf{T}(\mathbf{M}, 1)^T$ . Then the  $\mathbf{M}_c$  is normalised to obtain  $\mathbf{M}_n = (X_c/Z_c, Y_c/Z_c)^T$ . The distorted point  $\mathbf{M}_d$  is calculated by:

$$\mathbf{M}_d = (1 + \mathbf{D}_c(1)r^2 + \mathbf{D}_c(2)r^4 + \mathbf{D}_c(5)r^6)\mathbf{M}_n + \begin{pmatrix} 2\mathbf{D}_c(3)X_nY_n + \mathbf{D}_c(4)(r^2 + 2X_n^2) \\ 2\mathbf{D}_c(4)X_nY_n + \mathbf{D}_c(3)(r^2 + 2Y_n^2) \end{pmatrix}$$

where,  $r^2 = X_n^2 + Y_n^2$ . Finally, the point  $\mathbf{m}$  on the image can be calculated by  $\mathbf{m} = \mathbf{K}\mathbf{M}_d$ .

However, the distorted  $\mathbf{K}$  matrix should be modified as:

$$\mathbf{K} = \begin{pmatrix} f_x & \alpha_c f_x & u_c \\ 0 & f_y & v_c \\ 0 & 0 & 1 \end{pmatrix}$$

For more details on camera calibration, readers are recommended to refer to the book [Hartley and Zisserman 2004] by Hartley and Zisserman.

## A.2 Importance Resampling

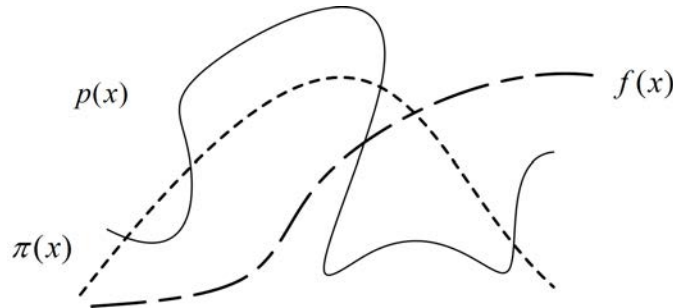


Figure A.2: Importance Sampling

The ultimate goal of all importance sampling techniques is to evaluate the expected

tation of some function  $f(x)$  with respect to a probability distribution  $p(x)$ . However, it is often difficult to sample directly from the probability distribution  $p(x)$ , a simpler/known distribution - the importance distribution  $\pi(x)$  - is thus introduced to sample from, and the corresponding terms are adjusted by the importance weights  $w = \frac{p(x)}{\pi(x)}$ . Moreover, using importance distributions also allows easy incorporation of domain knowledge and assumptions.

Let  $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^N$  denotes a set of  $N$  random samples  $\mathbf{x}_t^i$  with associated normalised importance weights  $w_t^i$  ( $\sum_{i=1}^N w_t^i = 1$ ) at time  $t$ . Provided that the number of samples  $N$  is reasonably large with respect to the dimensions of the state vector  $\mathbf{x}$ , an empirical estimate of posterior  $p(\mathbf{x}_t|\mathbf{y}_{1:t})$  at time  $t$  can be approximated as:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{j=1}^N w_t^j \delta_{\mathbf{x}_t^j}(\mathbf{x}_t)$$

Further, the state  $\hat{\mathbf{x}}_t$  can be estimated by the expectation of the posterior probability:

$$\hat{\mathbf{x}}_t = E[\mathbf{x}_t]_{p(\mathbf{x}_t|\mathbf{y}_{1:t})} = \int \mathbf{x}_t p(\mathbf{x}_t|\mathbf{y}_{1:t}) d\mathbf{x}_t \approx \sum_{i=1}^N w_t^i \mathbf{x}_t^i$$

When the number of samples  $N$  is fixed, the performance of the algorithm is dominated by the estimation of importance weights  $w_t^i$ . Further, the importance distribution should allow recursive evaluations in time of the importance weights as successive observations become available. It should therefore satisfy (for simplicity of notation,  $i$  superscripts are dropped):

$$\pi(\mathbf{x}_{t-1:t}|\mathbf{y}_{1:t}) = \pi(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{1:t})\pi(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \quad (\text{A.2.1})$$

The posterior equation can be derived in terms of the immediately previous state and available observations:

$$p(\mathbf{x}_{t-1:t}|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_{t-1:t}, \mathbf{y}_{1:t-1})p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{1:t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}$$

By applying Markov assumptions:

$$p(\mathbf{x}_{t-1:t}|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})} \quad (\text{A.2.2})$$

With equations A.2.1 and A.2.2, the importance weights can be defined as:

$$w_t^i = \frac{p(\mathbf{x}_{t-1:t}^i|\mathbf{y}_{1:t})}{\pi(\mathbf{x}_{t-1:t}^i|\mathbf{y}_{1:t})} = \frac{p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})}{\pi(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})} \cdot \frac{p(\mathbf{y}_t^i|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{\pi(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_{1:t}^i)p(\mathbf{y}_t^i|\mathbf{y}_{1:t-1}^i)} = w_{t-1}^i \frac{p(\mathbf{y}_t^i|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{\pi(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_{1:t}^i)p(\mathbf{y}_t^i|\mathbf{y}_{1:t-1}^i)}$$

Removing the normalised constant  $p(\mathbf{y}_t^i|\mathbf{y}_{1:t-1}^i)$ :

$$w_t^i \propto w_{t-1}^i \frac{p(\mathbf{y}_t^i|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{\pi(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_{1:t}^i)}$$

Assuming that the current state and observation are dependent solely upon the immediately previous state and current observation, the update equation can be reformulated as:

$$w_t^i \propto w_{t-1}^i \frac{p(\mathbf{y}_t^i|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{\pi(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_t^i)} \quad (\text{A.2.3})$$

## A.3 Human Body Segments and Joint Angle Ranges

### A.4 Parameterisations of Three DOF Rotations

#### A.4.1 Rotation Matrices

Each rotation can be represented as a 3 by 3 orthogonal matrix whose determinant is equal to 1. The set of all such matrices forms the special group  $SO(3)$ . It is also a closed set under matrix multiplication. Whenever a rotation matrix multiplies with another rotation matrix, the product remains a rotation matrix. Because of the linearity of matrices, directly optimising the rotation matrix often leads to well-defined linear objective functions. However, in order to ensure the matrix remains in  $SO(3)$ , there are six non-linear constraints that need to be enforced. Three constraints are to restrict

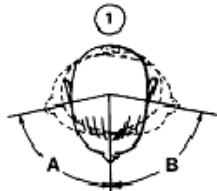
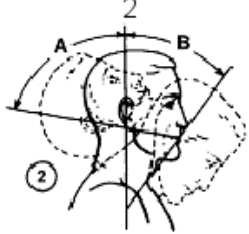
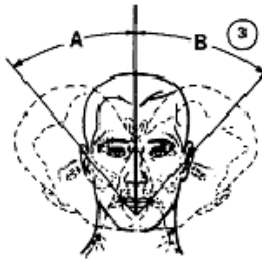
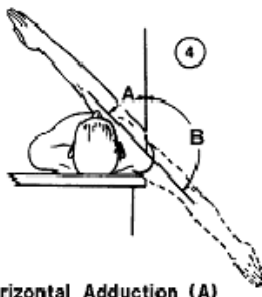
Figure	Joint movement (note b)	Range of motion (degrees)			
		Males (note a)		Female (note a)	
		5th percentile	95th percentile	5th percentile	95th percentile
1  Neck Rotation Right (A) Left (B)	Neck, rotation right (A)	73.3	99.6	74.9	108.8
	Neck, rotation left (B)	74.3	99.1	72.2	109.0
2  Neck Extension [A] Flexion (B)	Neck, flexion (B)	34.5	71.0	46.0	84.4
	Neck, extension (A)	65.4	103.0	4.9	103.0
3  Neck Lateral Bend Right (A) Left (B)	Neck, lateral bend right (A)	34.9	63.5	37.0	63.2
	Neck, lateral bend left (B)	35.5	63.5	29.1	77.2
4  Horizontal Adduction (A) Horizontal Abduction (B)	Shoulder, abduction	173.2	188.7	172.6	192.9

Figure A.3: Joint Angle Ranges

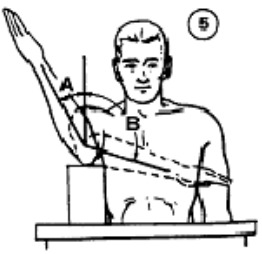
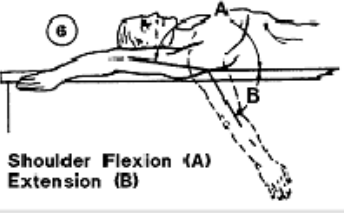
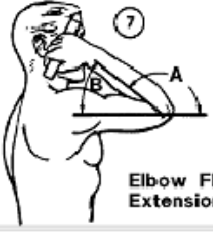
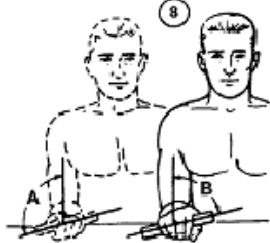
<p>5</p>  <p>Shoulder Rotation Lateral (A) Medial (B)</p>	Shoulder, rotation lateral (A)	46.3	96.7	53.8	85.8
	Shoulder, rotation medial (B)	90.5	126.6	95.8	130.9
<p>6</p>  <p>Shoulder Flexion (A) Extension (B)</p>	Shoulder, flexion (A)	164.4	210.9	152.0	217.0
	Shoulder, extension (B)	39.6	83.3	33.7	87.9
<p>7</p>  <p>Elbow Flexion (A) Extension (B)</p>	Elbow, flexion (A)	140.5	159.0	144.9	165.9
<p>8</p>  <p>Forearm Supination (A) Pronation (B)</p>	Forearm, pronation (B)	78.2	116.1	82.3	118.9
	Forearm, supination (A)	83.4	125.8	90.4	139.5

Figure A.4: Joint Angle Ranges

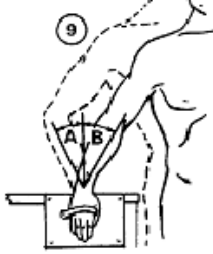
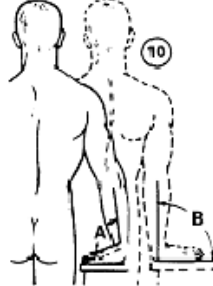

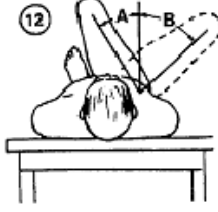


<p>9</p>  <p>Wrist Ulnar Bend (A) Radial Bend (B)</p>	<p>Wrist, radial bend (B)</p> <p>Wrist, ulnar bend (A)</p>	<p>16.9</p> <p>18.6</p>	<p>36.7</p> <p>47.9</p>	<p>16.1</p> <p>21.5</p>	<p>36.1</p> <p>43.0</p>
<p>10</p>  <p>Wrist Flexion (A) Extension (B)</p>	<p>Wrist, flexion (A)</p> <p>Wrist, extension (B)</p>	<p>61.5</p> <p>40.1</p>	<p>94.8</p> <p>78.0</p>	<p>68.3</p> <p>42.3</p>	<p>98.1</p> <p>74.7</p>
<p>11</p>  <p>Hip Flexion</p>	<p>Hip, flexion</p>	<p>116.5</p>	<p>148.0</p>	<p>118.5</p>	<p>145.0</p>
<p>12</p>  <p>Hip Adduction (A) Abduction (B)</p>	<p>Hip, abduction (B)</p>	<p>26.8</p>	<p>53.5</p>	<p>27.2</p>	<p>55.9</p>

Figure A.5: Joint Angle Ranges

<p>13</p>  <p>Knee Flexion, Prone</p>	Knee, flexion	118.4	145.6	125.2	145.2
<p>14</p>  <p>Ankle Plantar Extension (A) Dorsi Flexion</p>	Ankle, plantar extension (A)	36.1	79.6	44.2	91.1
	Ankle, dorsi flexion (B)	8.1	19.9	6.9	17.4

Notes:

a. Data was taken 1979 and 1980 at NASA-JSC by Dr. William Thornton and John Jackson. The study was made using 192 males (mean age 33) 22 females (mean age 30) astronaut candidates (see [Reference 365](#)).

b. Limb range is average of right and left limb movement.

Figure A.6: Joint Angle Ranges

Two-joint movement	Full range of A (degrees)	Change in range of movement of A (degrees)				
		Movement of B (fraction of full range)				
		Zero	1/3	1/2	2/3	Full
Shoulder extension (A) with elbow flexion (B)	59.3 deg		+1.6 deg (102.7%)		+0.9 deg (101.5%)	+5.3 deg (108.9%)
Shoulder flexion (A) with elbow flexion (B)	190.7 deg		-24.9 deg (86.9%)		-36.1 deg (81.0%)	-47.4 deg (75.0%)
Elbow flexion (A) with shoulder extension (A)	152.2 deg			-3.78 deg (97.5%)		-1.22 deg (99.2%)
Elbow flexion (A) with shoulder flexion (B)	152.2 deg		-0.6 deg (99.6%)		-0.8 deg (99.5%)	-69.0 deg (54.7%)
Hip flexion (A) with shoulder flexion (B)	53.3 deg	-35.6 deg * (33.2%)	-24.0 deg (55.0%)		-6.2 deg (88.4%)	-12.3 deg (76.9%)
Ankle plantar flexion (A) with knee flexion (B)	48.0 deg		-3.4 deg (92.9%)		+0.2 deg (100.4%)	+1.6 deg (103.3%)
Ankle dorsiflexion (A) with knee flexion (B)	26.1 deg		-7.3 deg (72.0%)		-2.7 deg (89.7%)	-3.2 deg (87.7%)
Knee flexion (A) with ankle plantar flexion (B)	127.0 deg			-9.9 deg (92.2%)		-4.7 deg (96.3%)
Knee flexion (A) with ankle dorsiflexion (B)	127.0 deg					-8.7 deg (93.0%)
Knee flexion (A) with hip flexion (B)	127.0 deg			-19.6 deg (84.6%)		-33.6 deg (73.5%)

Notes:

\* The knee joint is locked and the unsupported leg extends out in front of the subject.

The following is an example of how the Figure is to be used. The first entry is as follows: the shoulder can be extended as far as 59.3 degrees (the mean of the subjects tested) with the elbow in a neutral position (locked in hyperextension). When shoulder extension was measured with the elbow flexed to 1/3 of its full joint range, the mean value of shoulder extension was found to increase by 1.6 degrees, or 102.7% of the base value. The results for other movements and adjacent joint positions are presented in a similar manner.

Figure A.7: Joint Angle Ranges



column vectors to unit length, and three to keep them mutually orthogonal.

### A.4.2 Euler Angles

Euler angles define an arbitrary rotation by composition of three coordinate axis-angle rotations. They are widely used in various areas, owing to the fact that they are a more intuitive representation of rotation, being similar to human perception. However, the composition of rotations (matrix multiplication) is not commutative – the Euler angle representation of a rotation is not unique. Different axis orders yield different sets of Euler angles, even if these sets of Euler angles have the same effect on the object. In a right hand coordinate system, the three coordinate axis  $x$ ,  $y$  and  $z$  rotation is given by:

$$\begin{aligned} \mathbf{R}_x(\psi) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \psi & -\sin \psi \\ 0 & \sin \psi & \cos \psi \end{pmatrix} \\ \mathbf{R}_y(\theta) &= \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \\ \mathbf{R}_z(\phi) &= \begin{pmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

The final rotation matrix  $R$  can be composed in the  $XYZ$  order by:

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_z \mathbf{R}_y \mathbf{R}_x \\ &= \begin{pmatrix} \cos \theta \cos \phi & \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi & \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi \\ \cos \theta \sin \phi & \sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi & \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi \\ -\sin \theta & \sin \psi \cos \theta & \cos \psi \cos \theta \end{pmatrix} \end{aligned}$$

Conversely, given a rotation matrix  $R$ , Euler angles can be decomposed in the XYZ order by:

$$\begin{cases} \theta = -\arcsin R_{31} & \psi = \arctan\left(\frac{R_{32}}{\frac{R_{32}}{\cos\theta}}\right) & \phi = \arctan\left(\frac{R_{21}}{\frac{R_{11}}{\cos\theta}}\right) & \text{if } R_{31} \neq \pm 1 \\ \phi = 0 & \theta = \mp \frac{\pi}{2} & \psi = \mp \arctan\left(\frac{R_{12}}{R_{13}}\right) & \text{if } R_{31} = \pm 1 \end{cases}$$

### A.4.3 Axis Angle

The axis angle is established on the basis that any rotation or sequence of rotations in three-dimensional space is equivalent to a pure rotation about a single fixed axis. It parameterises a rotation by a unit vector  $\omega$  indicating the direction of an axis and an angle  $\theta$  describing the magnitude of the rotation about the axis. It is often given as a 4-element vector by:

$$\langle \text{axis, angle} \rangle = \left( \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}, \theta \right)$$

This can also be merged into one 3-element vector with magnitude  $\theta$  as:

$$\langle \text{axis, angle} \rangle = \left( \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}, \theta \right) = \begin{pmatrix} \omega_x \theta \\ \omega_y \theta \\ \omega_z \theta \end{pmatrix} \quad (\text{A.4.1})$$

The exponential map is used as a transformation from the axis angle to the rotation matrix representation.

$$\begin{aligned} R &= \exp(\hat{\omega}\theta) = \sum_{k=0}^{\infty} \frac{(\hat{\omega}\theta)^k}{k!} = I + \hat{\omega}\theta + \frac{1}{2}(\hat{\omega}\theta)^2 + \frac{1}{6}(\hat{\omega}\theta)^3 + \dots \\ R &= I + \hat{\omega} \left( \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \dots \right) + \hat{\omega}^2 \left( \frac{\theta^2}{2!} - \frac{\theta^4}{4!} + \frac{\theta^6}{6!} - \dots \right) \\ R &= I + \hat{\omega} \sin(\theta) + \hat{\omega}^2 (1 - \cos(\theta)) \end{aligned}$$

This is the so-called Rodrigues' rotation formula, where  $R$  is a rotation matrix and  $\hat{\omega}$  is a skew-symmetric (or antisymmetric) matrix:

$$\hat{\omega} = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}$$

Conversely, to retrieve the axis angle from the rotation matrix, the log map can be outlined as below:

$$\begin{aligned} \theta &= \arccos\left(\frac{\text{trace}(R) - 1}{2}\right) \\ \omega &= \frac{1}{2\sin(\theta)} \begin{pmatrix} R(3,2) - R(2,3) \\ R(1,3) - R(3,1) \\ R(2,1) - R(1,2) \end{pmatrix} \end{aligned}$$

Note that there are singularities around  $\theta = 0$  and  $\|\theta\| = \pi$  which need to be managed. Alternatively, the eigen-decomposition of  $R$  yields the three eigenvalues 1 and  $\cos\theta \pm i\sin\theta$ . The axis  $\omega$  is the eigenvector corresponding to the eigenvalue 1. The angle  $\theta$  can now be calculated from one of the remaining Eigenvalues. The consistency of the direction of the axis and angle should be checked. The quaternion representation has similar transformations although extra steps are needed to maintain normalisation of quaternions.

#### A.4.4 Optimisation on Axis Angle

In motion capture, 3D rotations are optimised in a temporal manner. The 3D rotation parameterisation is expected to be continuous over the entire space, thus smooth motion can be represented as the accumulative 3D rotations. Euler angles do not satisfy this expectation. When a motion is near the gimbal lock singularities, two Euler angle axes tend to be overlapped, and two axis rotations are rotated along the roughly same

---

direction. This implies degeneration (or even loss) of one degree of freedom. The axis angle and quaternions have better behaviour. However optimisation of the axis angle or quaternions introduces one additional degree of freedom compared with Euler angles. Although the axis angle has 3-element vector representation as in Equation (A.4.1), it has difficulties in imposing optimisation constraints on the rotation axis and angle separately. In fact, separation of these constraints is desired in motion capture.

In motion capture, the joint rotations change slowly between successive frames. It is safe to assume the current rotation is a small incremental adjustment of the previous one, so we can restrict the search space to the small neighbourhood around the previous joint rotation. In addition, when variation of the rotational increment is very small, this increment in a particular direction could become negligible, and the increment will only manifest in the other two orthogonal directions. This inspired an efficient optimisation method for 3D rotation, proposed by Schmidt and Niemann in [Schmidt and Niemann 2001]. As illustrated in Figure A.8, given an initial rotation  $P_0$ , the tangent hyperplane through  $P_0$  can be formed by finding two arbitrary orthonormal bases  $u, v$  that are perpendicular to the normal through  $P_0$ . A pair of orthonormal bases can be established by Gram-Schmidt Orthogonalisation or Singular Value Decomposition (SVD). Subsequently, any rotation  $p$  can be projected onto the tangential hyperplane, then perturbed along these two bases. Finally, it can be recovered to the unit hypersphere (the original rotation space). This approach avoids gimbal lock in the Euler angles and maintains the single 3D rotation parameterisation as 3 DOF rather than 4 DOF. It also provides a sensible way to constrain search space in the optimisation procedure.

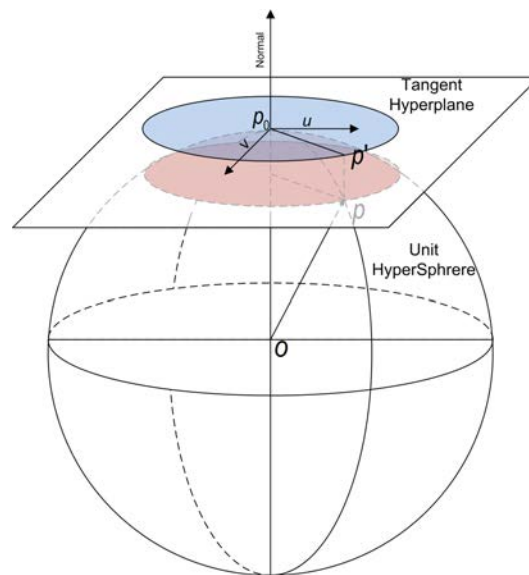


Figure A.8: Optimising 3D rotation in the tangential HyperPlane



---

# Bibliography

---

2000. *Comparing inertia weights and constriction factors in particle swarm optimization*, Volume 1 (2000). (p. 92)
2001. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation (Genetic Algorithms and Evolutionary Computation)*. Springer. (p. 99)
- ACKLEY, D. H. 1987. *A connectionist machine for genetic hillclimbing*. Kluwer, Boston. (pp. 5, 110)
- AGARWAL, A. AND TRIGGS, B. 2004a. 3d human pose from silhouettes by relevance vector regression. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2004), pp. 882–888. (p. 22)
- AGARWAL, A. AND TRIGGS, B. 2004b. Learning to track 3d human motion from silhouettes. In *International Conference on Machine Learning* (2004). (p. 22)
- AGARWAL, A. AND TRIGGS, B. 2006. Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 1, 44–58. (pp. 1, 22, 25)
- AHMED, N., DE AGUIAR, E., THEOBALT, C., MAGNOR, M., AND SEIDEL, H.-P. 2005. Automatic generation of personalized human avatars from multi-view video. In *VRST '05: Proceedings of the ACM symposium on Virtual reality software and technology* (Monterey, USA, December 2005), pp. 257–260. Association for Computing Machinery (ACM): ACM. (p. 74)
- AHMED, N., LENSCH, H., AND SEIDEL, H.-P. 2007. Seeing people in different light-joint shape, motion, and reflectance capture. *IEEE Transactions on Visualization and Computer Graphics* 13, 4, 663–674. Member-Christian Theobalt and Member-

Marcus Magnor. (p.129)

AHMED, N., THEOBALT, C., RÖSSL, C., THRUN, S., AND SEIDEL, H.-P. 2008. Dense correspondence finding for parametrization-free animation reconstruction from video. In *CVPR (Anchorage, Alaska, 2008)*. IEEE Computer Society. (p.129)

ALEXA, M. 2003. Differential coordinates for local mesh morphing and deformation. *The Visual Computer* 19, 2-3, 105–114. (p.20)

ALLARD, J., FRANCO, J.-S., MENIER, C., BOYER, E., AND RAFFIN, B. 2006. The grimage platform: A mixed reality environment for interactions. In *ICVS '06: Proceedings of the Fourth IEEE International Conference on Computer Vision Systems* (Washington, DC, USA, 2006), pp. 46. IEEE Computer Society.

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2003. The space of human body shapes: reconstruction and parameterization from range scans. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers* (New York, NY, USA, 2003), pp. 587–594. ACM Press. (pp.52, 53, 70)

ALLEN, B., CURLESS, B., POPOVIĆ, Z., AND HERTZMANN, A. 2006. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2006), pp. 147–156. Eurographics Association. (pp.52, 53)

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005a. Scape: shape completion and animation of people. *ACM Trans. Graph.* 24, 3, 408–416. (pp.16, 52, 53, 54)

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005b. Scape: shape completion and animation of people. *ACM Trans. Graph.* 24, 3, 408–416.

ARULAMPALAM, S., MASKELL, S., GORDON, N., AND CLAPP, T. 2002. A tuto-



- 
- rial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing* 50, 2 (Feb.), 174–188. (p. 44)
- AZOUZ, Z. B., RIOUX, M., SHU, C., AND LEPAGE, R. 2006. Characterizing human shape variation using 3d anthropometric data. *The Visual Computer* 22, 5, 302–314. (p. 69)
- BAAK, A., ROSENHAHN, B., MÜLLER, M., AND SEIDEL, H.-P. 2009. Stabilizing motion tracking using retrieved motion priors. In *IEEE 12th International Conference on Computer Vision* (Sept. 2009), pp. 1428–1435.
- BÄCK, T. 1996. *Evolutionary algorithms in theory and practice*. Oxford University Press. (p. 110)
- BALAN, A. AND BLACK, M. 17-22 June 2006. An adaptive appearance model approach for model-based articulated object tracking. *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* 1, 758–765. (pp. 1, 54)
- BALAN, A. O., SIGAL, L., AND BLACK, M. J. 2005. A quantitative evaluation of video-based 3d person tracking. In *ICCCN '05: Proceedings of the 14th International Conference on Computer Communications and Networks* (2005), pp. 349–356. IEEE Computer Society. (p. 126)
- BALAN, A. O., SIGAL, L., BLACK, M. J., DAVIS, J. E., AND HAUSSECKER, H. W. 2007. Detailed human shape and pose from images. In *CVPR* (2007). (p. 54)
- BANDOUC, J. AND BEETZ, M. 2009. Tracking humans interacting with the environment using efficient hierarchical sampling and layered observation models. In *IEEE Int. Workshop on Human-Computer Interaction (HCI). In conjunction with ICCV2009* (2009). (pp. 149, 151)
- BARANIUK, R. G., DAVENPORT, M. A., DEVORE, R. A., AND WAKIN, M. B. 2008. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation* 28, 3 (Dec.), 253–263. (p. 163)

- BARANIUK, R. G. AND WAKIN, M. B. 2009. Random projections of smooth manifolds. *Foundations of Computational Mathematics* 9, 1, 51–77. (p.169)
- BARON, D., DUARTE, M. F., WAKIN, M. B., SARVOTHAM, S., AND BARANIUK, R. G. 2009. Distributed compressive sensing. *the Computing Research Repository abs/0901.3403*. (p.168)
- BAUM, L. E., PETRIE, T., SOULES, G., AND WEISS, N. 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics* 41, 1, 164–171. (p.43)
- BELKIN, M. AND NIYOGI, P. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* 15, 6, 1373–1396.
- BELONGIE, S., MALIK, J., AND PUZICHA, J. 2002. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.* 24, 4, 509–522. (pp.23, 25)
- BERNHARD, S. M., SCHOLKOPF, B., SMOLA, A., ROBERT M ULLER, K., SCHOLZ, M., AND ATSCH, G. R. 1999. Kernel pca and de-noising in feature spaces. In *Advances in Neural Information Processing Systems* 11 (1999), pp. 536–542. MIT Press. (p.27)
- BERNIER, O. 2006. Real-time 3d articulated pose tracking using particle filters interacting through belief propagation. *International Conference on Pattern Recognition* 1, 90–93.
- BERTALMÍO, M., SAPIRO, G., CASELLES, V., AND BALLESTER, C. 2000. Image inpainting. In *SIGGRAPH* (2000), pp. 417–424. (p.133)
- BESL, P. J. AND MCKAY, N. D. 1992. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* 14, 2, 239–256. (p.55)
- BISHOP, C. M. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

- 
- BISHOP, C. M. AND TIPPING, M. E. 1998. A hierarchical latent variable model for data visualization. *IEEE Trans. Pattern Anal. Mach. Intell.* 20, 3, 281–293.
- BLACK, M. J. AND FLEET, D. J. 2000. Probabilistic detection and tracking of motion boundaries. *Int. J. Comput. Vision* 38, 3, 231–245.
- BLAKE, A. AND ZISSERMAN, A. 1987. *Visual Reconstruction*. MIT Press. (p. 121)
- BLANZ, V. AND VETTER, T. 1999. A Morphable Model for the Synthesis of 3D Faces. In A. ROCKWOOD Ed., *Siggraph 1999, Computer Graphics Proceedings* (Los Angeles, 1999), pp. 187–194. Addison Wesley Longman. (pp. 69, 70)
- BLENDER. 2010. Blender is the free open source 3d content creation suite. <http://www.Blender.org>. (pp. 70, 130)
- BOYD, S. AND VANDENBERGHE, L. 2004. *Convex Optimization*. Cambridge University Press, New York, NY, USA. (p. 159)
- BRAY, M., KOHLI, P., AND TORR, P. H. S. 2006. Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *ECCV (2)* (2006), pp. 642–655. (pp. 30, 31)
- BREGLER, C., MALIK, J., AND PULLEN, K. 2004. Twist based acquisition and tracking of animal and human kinematics. *Int. J. Comput. Vision* 56, 3, 179–194.
- BRESENHAM, J. E. 1965. Algorithm for Computer Control of a Digital Plotter. *IBM System Journal* 4, 1, 25–30. (p. 65)
- BRUBAKER, M. A., FLEET, D. J., AND HERTZMANN, A. 2007. Physics-based person tracking using simplified lower-body dynamics. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* 0, 1–8.
- BRUBAKER, M. A., FLEET, D. J., AND HERTZMANN, A. 2010. Physics-based person tracking using the anthropomorphic walker. *Int. J. Comput. Vision* 87, 1-2, 140–155.
- BĂLAN, A. O. AND BLACK, M. J. 2006. An adaptive appearance model approach

- for model-based articulated object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, Volume 1 (2006), pp. 758–765.
- CALVERT, T. W., CHAPMAN, J., AND PATLA, A. 1982. Aspects of the kinematic simulation of human movement. *IEEE Comput. Graph. Appl.* 2, 9, 41–50. (p.1)
- CANDES, E. J. AND ROMBERG, J. 2007. Sparsity and incoherence in compressive sampling. *Inverse Problems* 23, 3 (June), 969–985. (pp.155, 160)
- CANDÈS, E. J., ROMBERG, J. K., AND TAO, T. (pp.155, 161, 162)
- CANDÈS, E. J., ROMBERG, J. K., AND TAO, T. 2006. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory* 52, 2, 489–509. (pp.7, 155, 159)
- CANDES, E. J. AND TAO, T. 2005. Decoding by linear programming. *Information Theory, IEEE Transactions on* 51, 12, 4203–4215. (pp.155, 156, 161, 168)
- CANDÈS, E. J. AND TAO, T. 2006. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory* 52, 12, 5406–5425. (p.7)
- CANDES, E. J. AND WAKIN, M. B. 2008. An introduction to compressive sampling. *IEEE Signal Processing Magazine* 25, 2 (March), 21–30. (p.7)
- CARRANZA, J., THEOBALT, C., MAGNOR, M. A., AND SEIDEL, H.-P. 2003. Free-viewpoint video of human actors. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers* (New York, NY, USA, 2003), pp. 569–577. ACM. (p.129)
- CEVHER, V., SANKARANARAYANAN, A., DUARTE, M., REDDY, D., BARANIUK, R., AND CHELLAPPA, R. 2008. Compressive sensing for background subtraction. In D. FORSYTH, P. TORR, AND A. ZISSERMAN Eds., *Computer Vision ECCV 2008*, Volume 5303 of *Lecture Notes in Computer Science*, Chapter 12, pp. 155–168. Springer Berlin Heidelberg. (p.155)
- CHABERT, C.-F., EINARSSON, P., JONES, A., LAMOND, B., MA, W.-C., SYLWAN,

- 
- S., HAWKINS, T., AND DEBEVEC, P. 2006. Relighting human locomotion with flowed reflectance fields. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches* (New York, NY, USA, 2006), pp. 76. ACM. (p.20)
- CHEN, S. S., DONOHO, D. L., AND SAUNDERS, M. A. 1999. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing* 20, 1, 33–61. (p.159)
- CHENG, S. Y. AND TRIVEDI, M. M. 2007. Articulated human body pose inference from voxel data using a kinematically constrained gaussian mixture model. In *2-nd Workshop on Evaluation of Articulated Human Motion and Pose Estimation (EHUM2)*, CVPR (2007). IEEE. (pp.149, 151)
- CIPOLLA, R. AND GIBLIN, P. 2000. *Visual motion of curves and surfaces*. Cambridge University Press, New York, NY, USA. (pp.65, 66)
- COOPER, S., HERTZMANN, A., AND POPOVIĆ, Z. 2007. Active learning for real-time motion controllers. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers* (New York, NY, USA, 2007), pp. 5. ACM.
- CORAZZA, S., GAMBARETTO, E., MUNDERMANN, L., AND ANDRIACCHI, T. P. 2009. Automatic generation of a subject specific model for accurate markerless motion capture and biomechanical applications. *IEEE Transactions on Biomedical Engineering*. (pp.15, 16, 74)
- CORAZZA, S., MNDERMANN, L., GAMBARETTO, E., AND ANDRIACCHI, T. P. 2010. Markerless motion capture through visual hull, articulated icp and subject specific model generation. *International Journal of Computer Vision* 87, 156–169. (pp.15, 16, 54, 151)
- CORAZZA, S., MUNDERMANN, L., CHAUDHARI, A., DEMATTIO, T., COBELLI, C., AND ANDRIACCHI, T. June 2006. A markerless motion capture system to study musculoskeletal biomechanics: Visual hull and simulated annealing approach. *Annals of Biomedical Engineering* 34, 1019–1029(11).

- CZYŻ, J. 2006. Object detection in video via particle filters. *icprInternational Conference on Pattern Recognition 1*, 820–823.
- DAUBECHIES, I. 1992. *Ten Lectures on Wavelets (C B M S - N S F Regional Conference Series in Applied Mathematics)*. Society for Industrial and Applied Mathematics. (p.172)
- DE AGUIAR, E., STOLL, C., THEOBALT, C., AHMED, N., SEIDEL, H.-P., AND THRUN, S. 2008. Performance capture from sparse multi-view video. In *SIGGRAPH '08: ACM SIGGRAPH 2008 papers* (New York, NY, USA, 2008), pp. 1–10. ACM. (pp.17, 19, 55)
- DE AGUIAR, E., THEOBALT, C., MAGNOR, M., AND SEIDEL, H.-P. 2005. Reconstructing human shape and motion from multi-view video. In *2nd European Conference on Visual Media Production (CVMP)* (London, UK, December 2005), pp. 42–49. The IEE. (p.74)
- DE AGUIAR, E., THEOBALT, C., STOLL, C., AND SEIDEL, H.-P. 2007a. Marker-less 3d feature tracking for mesh-based motion capture. In A. ELGAMMAL, B. ROSENHAHN, AND R. KLETTE Eds., *Human Motion - Understanding, Modeling, Capture and Animation*, Volume 4814 of *Lecture Notes in Computer Science* (Rio de Janeiro, Brazil, October 2007), pp. 1–15. Springer. (p.129)
- DE AGUIAR, E., THEOBALT, C., STOLL, C., AND SEIDEL, H.-P. 2007b. Marker-less deformable mesh tracking for human shape and motion capture. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)* (Minneapolis, USA, June 2007), pp. XX–XX. IEEE: IEEE.
- DE AGUIAR, E., THEOBALT, C., THRUN, S., AND SEIDEL, H.-P. 2008. Automatic conversion of mesh animations into skeleton-based animations. *Comput. Graph. Forum* 27, 2, 389–397. (p.129)
- DE LA GORCE, M., PARAGIOS, N., AND FLEET, D. J. 2008. Model-based hand tracking with texture, shading and self-occlusions. In *CVPR* (2008).

- 
- DENNY, M. 2001. Introduction to importance sampling in rare-event simulations. *European Journal of Physics* 22, 4, 403–411. (p.44)
- DEUTSCHER, J., BLAKE, A., AND REID, I. 2000. Articulated body motion capture by annealed particle filtering. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, Volume 2 (2000), pp. 126–133 vol.2. (pp.10, 147, 148)
- DEUTSCHER, J., DAVISON, A., AND REID, I. 2001. Automatic partitioning of high dimensional search spaces associated with articulated body motion capture. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* 2, 669. (p.10)
- DEUTSCHER, J., NORTH, B., BASCLE, B., AND BLAKE, A. 1999. Tracking through singularities and discontinuities by random sampling. In *ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2* (Washington, DC, USA, 1999), pp. 1144. IEEE Computer Society. (p.10)
- DEUTSCHER, J. AND REID, I. 2005. Articulated body motion capture by stochastic search. *International Journal of Computer Vision* 61, 2, 185–205. (pp.10, 85, 87, 126)
- DONOHO, D. L. 2006. Compressed sensing. *IEEE Transactions on Information Theory* 52, 4, 1289–1306. (p.7)
- DOUCET, A., GODSILL, S., AND ANDRIEU, C. 2000. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing* 10, 3, 197–208. (pp.10, 44)
- DUARTE, M. F., DAVENPORT, M. A., TAKHAR, D., LASKA, J. N., SUN, T., KELLY, K. F., AND BARANIUK, R. G. 2008. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine* 25, 2 (March), 83–91. (p.155)
- DUCHENNE, O., AUDIBERT, J.-Y., KERIVEN, R., PONCE, J., AND SEGONNE, F. 2008. Segmentation by transduction. *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8. (pp.33, 34)

- EISEMANN, M., DECKER, B. D., MAGNOR, M., BEKAERT, P., DE AGUIAR, E., AHMED, N., THEOBALT, C., AND SELLENT, A. 2008. Floating Textures. *Computer Graphics Forum (Proc. Eurographics EG'08)* 27, 2 (4), xx–xx. (p.129)
- EK, C., TORR, P., AND LAWRENCE, N. 2008. Gaussian process latent variable models for human pose estimation. pp. 132–143. (p.25)
- EL-MARAGHI, T. F. 2003. *Robust online appearance models for visual tracking*. PhD thesis, Toronto, Ont., Canada, Canada. Adviser-Allan D. Jepson.
- ELGAMMAL, A. AND LEE, C.-S. 2007. Nonlinear manifold learning for dynamic shape and dynamic appearance. *Comput. Vis. Image Underst.* 106, 1, 31–46.
- ELGAMMAL, A. AND LEE, C.-S. 2009. Tracking people on a torus. *IEEE Trans. Pattern Anal. Mach. Intell.* 31, 3, 520–538. (pp.27, 28)
- FLEET, D. J., BLACK, M. J., AND NESTARES, O. 2003. Bayesian inference of visual motion boundaries. pp. 139–173.
- GALL, J., POTTHOFF, J., SCHNOERR, C., ROSENHAHN, B., AND SEIDEL, H.-P. 2007. Interacting and annealing particle filters: Mathematics and a recipe for applications. *Journal of Mathematical Imaging and Vision* 28, 1, 1–18. (pp.12, 50)
- GALL, J., ROSENHAHN, B., BROX, T., AND SEIDEL, H.-P. 2010. Optimization and filtering for human motion capture - a multi-layer framework. *International Journal of Computer Vision* 87, 75–92. (pp.15, 149, 151)
- GALL, J., ROSENHAHN, B., AND SEIDEL, H.-P. 2008. Drift-free tracking of rigid and articulated objects. In *CVPR (2008)*. (p.14)
- GALL, J., STOLL, C., DE AGUIAR, E., THEOBALT, C., ROSENHAHN, B., AND SEIDEL, H.-P. 2009. Motion capture using joint skeleton tracking and surface estimation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (June 2009)*, pp. 1746–1753. (pp.15, 54)
- GINSBERG, C. M. AND MAXWELL, D. 1984. "graphical marionette". *SIGGRAPH*



- 
- Comput. Graph.* 18, 1, 26–27.
- GONG, R. H. AND ABOLMAESUMI, P. 2008. 2d/3d registration with the cma-es method. Volume 6918 (2008), pp. 69181M. SPIE.
- GRAU, O., HILTON, A., KILNER, J., MILLER, G., SARGEANT, T., AND STARCK, J. 2006. A free-viewpoint video system for visualisation of sport scenes. *International Broadcasting Convention (IBC)*.
- GRIEWANK, A. 1981. Generalized descent for global optimization. *Journal of Optimization Theory and Applications* 11, 11–39. Berlin Heidelberg. (p. 114)
- GROCHOW, K., MARTIN, S. L., HERTZMANN, A., AND POPOVIĆ, Z. 2004. Style-based inverse kinematics. *ACM Trans. Graph.* 23, 3, 522–531.
- GUIASU, S. 1977. *Information Theory with Applications*. McGraw-Hill, New York. (p. 144)
- HANSEN, N. AND KERN, S. 2004. Evaluating the cma evolution strategy on multimodal test functions. In *PPSN* (2004), pp. 282–291. (pp. 93, 95)
- HANSEN, N., MÜLLER, S. D., AND KOUMOUTSAKOS, P. 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary Computation* 11, 1, 1–18. (p. 93)
- HANSEN, N. AND OSTERMEIER, A. 1996. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In *International Conference on Evolutionary Computation* (1996), pp. 312–317. (pp. 92, 95)
- HANSEN, N. AND OSTERMEIER, A. 2001. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation* 9, 2, 159–195. (pp. 93, 95)
- HARTLEY, R. I. AND ZISSERMAN, A. 2000. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049.

- HARTLEY, R. I. AND ZISSERMAN, A. 2004. *Multiple View Geometry in Computer Vision* (Second ed.). Cambridge University Press, ISBN: 0521540518. (p.183)
- HASLER, N., ROSENHAHN, B., THORMAHLEN, T., WAND, M., GALL, J., AND SEIDEL, H.-P. 2009. Markerless motion capture with unsynchronized moving cameras. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (June 2009), pp. 224–231.
- HASLER, N., STOLL, C., SUNKEL, M., ROSENHAHN, B., AND SEIDEL, H.-P. 2009. A statistical model of human pose and body shape. In P. DUTR’E AND M. STAMMINGER Eds., *Computer Graphics Forum (Proc. Eurographics 2008)*, Volume 2 (Munich, Germany, March 2009). (pp.52, 54)
- HEIN, M., YVES AUDIBERT, J., AND LUXBURG, U. V. 2005. From graphs to manifolds - weak and strong pointwise consistency of graph laplacians. In *Proceedings of the 18th Conference on Learning Theory (COLT (2005))*, pp. 470–485. Springer. (p.34)
- HILTON, A. AND STARCK, J. 2004. Multiple view reconstruction of people. *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 357–364.
- HILTON, A., STARCK, J., AND COLLINS, G. 2002. From 3d shape capture to animated models. *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 246–257.
- HILTON, A., STARCK, J., COLLINS, G., AND KALKAVOURAS, M. 2002. 3d shape capture for archiving and animation. In *Advanced Information Visualization in Archaeology Workshop* (2002).
- HORAUD, R. P., NISKANEN, M., DEWAELE, G., AND BOYER, E. 2009. Human motion tracking by registering an articulated surface to 3-d points and normals. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 1 (January), 158–164.
- HOTELLING, H. 1933. Analysis of a complex of statistical variables into principal

- 
- components. *Journal of Educational Psychology* 24, 417–441.
- HOU, S., GALATA, A., CAILLETTE, F., THACKER, N., AND BROMILEY, P. 14-21 Oct. 2007. Real-time body tracking using a gaussian process latent variable model. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 1–8. (p.1)
- HOUSEHOLDER, A. S. 1958. Unitary triangularization of a nonsymmetric matrix. *Journal of the ACM* 5, 4, 339–342. (p.169)
- HUMAN ANIMATION WORKING GROUP. Information technology computer graphics and image processing humanoid animation (h-anim). *ISO/IEC FCD 19774:200x version 1.1*. (p.55)
- IGEHY, H. AND PEREIRA, L. 1997. Image replacement through texture synthesis. *Image Processing, International Conference on* 3, 186. (p.133)
- INGBER, L. 1996. Adaptive simulated annealing (asa): Lessons learned. *Control and Cybernetics* 25, 33–54.
- ISARD, M. 2003. Pampas: Real-valued graphical models for computer vision. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* 1, 613. (p.21)
- ISARD, M. AND BLAKE, A. 1998a. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision* 29, 1, 5–28. (p.10)
- ISARD, M. AND BLAKE, A. 1998b. CONDENSATIONConditional Density Propagation for Visual Tracking. *International Journal of Computer Vision* 29, 1, 5–28. (p.85)
- ISO/IEC MOVING PICTURE EXPERTS GROUP. 2008. Information technology – coding of audio-visual objects – part 2: Visual. *ISO/IEC 14496-2:2004/Amd 4:2008*. (p.55)
- JACOBSON, N. 2009. *Basic Algebra II: Second Edition* (2 ed.). Dover Publications. (p.57)

- JAEGGLI, T., KOLLER-MEIER, E., AND GOOL, L. J. V. 2007. Learning generative models for monocular body pose estimation. In Y. YAGI, S. B. KANG, I.-S. KWEON, AND H. ZHA Eds., *ACCV (1)*, Volume 4843 of *Lecture Notes in Computer Science* (2007), pp. 608–617. Springer.
- JEPSON, A., FLEET, D., AND EL-MARAGHI, T. 2003. Robust online appearance models for visual tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25, 10 (Oct.), 1296–1311.
- JIE ZHAO, G.-Z. M., WEN QIAO. 2009. An approach based on mean shift and kalman filter for target tracking under occlusion. Volume 4 (July 2009), pp. 2058–2062.
- JOHNSON, W. AND LINDENSTRAUSS, J. 1984. Extensions of Lipschitz mappings into a Hilbert space. In *Conference in modern analysis and probability (New Haven, Conn., 1982)*, Volume 26 of *Contemporary Mathematics*, pp. 189–206. American Mathematical Society. (p.169)
- KAC, M. Jan., 1949. On distributions of certain wiener functionals. *Transactions of the American Mathematical Society* 65, 1 (Jan), 1–13. (p.12)
- KASAP, M. AND MAGNENAT-THALMANN, N. 2007. Parameterized human body model for real-time applications. In *CW '07: Proceedings of the 2007 International Conference on Cyberworlds* (Washington, DC, USA, 2007), pp. 160–167. IEEE Computer Society.
- KASS, M., WITKIN, A., AND TERZOPOULOS, D. 1988. Snakes: Active contour models. *INTERNATIONAL JOURNAL OF COMPUTER VISION* 1, 4, 321–331. (p.60)
- KE, Y. AND SUKTHANKAR, R. 2004. Pca-sift: A more distinctive representation for local image descriptors. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* 2, 506–513. (p.14)

- 
- KEHL, R. AND VAN GOOL, L. 2006. Markerless tracking of complex human motions from multiple views. *Comput. Vis. Image Underst.* 104, 190–209. (p.54)
- KENNEDY, J. AND EBERHART, R. 1995. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, Volume 4 (August 1995), pp. 1942–1948. (pp.88, 90, 91)
- KILNER, J., STARCK, J., AND HILTON, A. 2006. A comparative study of free-viewpoint video techniques for sports events. *European Conference on Visual Media Production (CVMP)*.
- KIRKPATRICK, S., GELATT, C. D., AND VECCHI, M. P. 1983. Optimization by simulated annealing. *Science, Number 4598, 13 May 1983* 220, 4598, 671–680. (pp.10, 83, 85, 120)
- KITAOKA, A. 2007. Pictorial explanation of the silhouette illusion. <http://www.psy.ritsumei.ac.jp/~akitaoka/illnews8e.html>. (p.128)
- KOHLI, P., RIHAN, J., BRAY, M., AND TORR, P. H. 2008. Simultaneous segmentation and pose estimation of humans using dynamic graph cuts. *Int. J. Comput. Vision* 79, 3, 285–298. (pp.30, 31)
- KOHLI, P. AND TORR, P. H. S. 2007. Dynamic graph cuts for efficient inference in markov random fields. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 12, 2079–2088. (p.30)
- KOLMOGOROV, V. AND ZABIH, R. 2002. What energy functions can be minimized via graph cuts? In A. HEYDEN, G. SPARR, M. NIELSEN, AND P. JOHANSEN Eds., *Computer Vision ECCV 2002*, Volume 2352 of *Lecture Notes in Computer Science*, Chapter 5, pp. 185–208. Berlin, Heidelberg: Springer Berlin Heidelberg. (p.32)
- LATHAUWER, L. D., MOOR, B. D., AND VANDEWALLE, J. 2000. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.* 21, 4, 1253–1278. (p.28)
- LAURENTINI, A. 1994. The visual hull concept for silhouette-based image under-

- 
- standing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16, 2, 150–162. (pp.4, 120)
- LAWRENCE, N. 2005. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *J. Mach. Learn. Res.* 6, 1783–1816. (p.25)
- LEE, A., CHAN-SU; ELGAMMAL. 14-21 Oct. 2007. Modeling view and posture manifolds for tracking. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 1–8. (p.27)
- LEE, C.-S. AND ELGAMMAL, A. M. 2006. Carrying object detection using pose preserving dynamic shape models. In *Articulated Motion and Deformable Objects* (2006), pp. 315–325. (p.1)
- LEE, D. D. AND SEUNG, H. S. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 6755 (October), 788–791. (p.23)
- LEWIS, J. P., CORDNER, M., AND FONG, N. 2000. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *SIGGRAPH* (2000), pp. 165–172. (p.59)
- LIU, J. S. AND CHEN, R. 1998. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association* 93, 443, 1032–1044. (p.49)
- LIU, JUN S. AND CHEN, RONG. 1995. Blind deconvolution via sequential imputations. *Journal of the American Statistical Association* 90, 430 (jun), 567–576. (p.49)
- LÓPEZ, F. J. P. AND FISHER, R. B. Eds. 2006. *Articulated Motion and Deformable Objects, 4th International Conference, AMDO 2006, Port d'Andratx, Mallorca, Spain, July 11-14, 2006, Proceedings*, Volume 4069 of *Lecture Notes in Computer Science* (2006). Springer.
- LOWE, D. G. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60, 2, 91–110.
- LUO, M. R., CUI, G., AND RIGG, B. 2001. The development of the cie 2000 colour-

- 
- difference formula: Ciede2000. *Color Research and Application* 26, 5 (Feb.), 340–350. (pp.134, 135)
- M. EMRE CELEBI, H. A. K. AND CELIKER, F. 2009. Fast color space transformations using minimax approximations. *IET Image Processing* 1, 3, 134–142.
- MACCORMICK, J. AND ISARD, M. 2000. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II* (London, UK, 2000), pp. 3–19. Springer-Verlag. (p.10)
- MACKAY, D. J. C. 1998. Introduction to monte carlo methods. In *Proceedings of the NATO Advanced Study Institute on Learning in graphical models* (Norwell, MA, USA, 1998), pp. 175–204. Kluwer Academic Publishers. (p.10)
- MAGENAT-THALMANN, N., LAPERRIERE, R., AND THALMANN, D. 1988. Joint-dependent local deformations for hand animation and object grasping. In *Proceedings on Graphics interface '88* (1988), pp. 26–33. Canadian Information Processing Society. (pp.59, 63)
- MAGENAT-THALMANN, N. AND SEO, H. 2004. Data-driven approaches to digital human modeling. In *3DPVT* (2004), pp. 380–387. (pp.52, 53)
- MAKEHUMAN. 2010. Open source tool for making 3d characters. <http://www.makehuman.org>. (p.70)
- MALLAT, S. 1999. *A Wavelet Tour of Signal Processing, Second Edition (Wavelet Analysis & Its Applications)* (2 ed.). Academic Press. (p.163)
- MALLAT, S. G. 1989. A theory for multiresolution signal decomposition: the wavelet representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 11, 7, 674–693. (p.163)
- MCDONALD, R. AND SMITH, K. J. 1995. Cie94-a new colour-difference formula. *Journal of the Society of Dyers and Colourists* 111, 12, 376–379. (p.134)

- MEI, X. AND LING, H. 2009. Robust visual tracking using  $l_1$  minimization. In *ICCV09 (2009)*, pp. 1436–1443. (p.155)
- METROPOLIS, N., ROSENBLUTH, A. W., ROSENBLUTH, M. N., TELLER, A. H., AND TELLER, E. 1953. Equations of state calculations by fast computing machine. *Journal of Chemical Physics* 21, 1087–1091(1092?). (p.84)
- MICHOUD, B., GUILLOU, E., BRICEÑO, H. M., AND BOUAKAZ, S. 2007. Real-time marker-free motion capture from multiple cameras. In *ICCV (2007)*, pp. 1–7.
- MICROSOFT. 2010. Natal project, <http://www.xbox.com/en-us/live/projectnatal/>.
- MILLER, G., HILTON, A., AND STARCK, J. 2005. Interactive free-viewpoint video. *European Conference on Visual Media Production (CVMP)*, 52–61.
- MILLER, G., STARCK, J., AND HILTON, A. 2006. Projective surface refinement for free-viewpoint video. *European Conference on Visual Media Production (CVMP)*.
- MILLONAS, M. M. 1994. Swarms, phase transitions and collective intelligence. In C. LANGTON Ed., *Artificial Life III*. Addison-Wesley. (p.90)
- MORAL, P. D. AND DOUCET, A. 2003. On a class of genealogical and interacting metropolis models. In *In Seminaire de Probabilites XXXVII (2003)*, pp. 415–446. Springer. (p.14)
- MOTION, O. 2010. <http://www.organicmotion.com/>.
- MÜHLENBEIN, H., SCHOMISCH, D., AND BORN, J. 1991. The Parallel Genetic Algorithm as Function Optimizer. *Parallel Computing* 17, 6-7, 619–632. (p.111)
- MUNDERMANN, L., CORAZZA, S., AND ANDRIACCHI, T. P. 2007. Accurately measuring human movement using articulated icp with soft-joint constraints and a repository of articulated models. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* 0, 1–6. (pp.15, 17)



- 
- NATARAJAN, B. K. 1995. Sparse approximate solutions to linear systems. *SIAM J. Comput.* 24, 2, 227–234. (p.159)
- NEAL, R. M. 2001. Annealed importance sampling. *Statistics and Computing* 11, 2, 125–139.
- P., D. 2003. Sequential monte carlo methods in practice. *Journal of the American Statistical Association* 98, 496–497.
- PEDERSEN, M. E. H. AND CHIPPERFIELD, A. J. 2010. Simplifying particle swarm optimization. *Appl. Soft Comput.* 10, 618–628. (p.92)
- POLI, R. 2008. Analysis of the publications on the applications of particle swarm optimisation. *J. Artif. Evol. App.* 2008, 1 (January), 1–10. (p.90)
- POLI, R., KENNEDY, J., AND BLACKWELL, T. 2007. Particle swarm optimization. *Swarm Intelligence* 1, 1 (June), 33–57. (p.90)
- PRAUN, E. AND HOPPE, H. 2003. Spherical parametrization and remeshing. *ACM Trans. Graph.* 22, 3, 340–349. (p.18)
- PRICE, M., CHANDARIA, J., GRAU, O., THOMAS, G., CHATTING, D., THORNE, J., MILNTHORPE, G., WOODWARD, P., BULL, L., ONG, E.-J., HILTON, A., MITCHELSON, J., AND STARCK, J. 2002. Real-time production and delivery of 3D media. *Proceedings of the International Broadcasting Convention*.
- PUPILLI, M. AND CALWAY, A. 2006. Real-time camera tracking using known 3d models and a particle filter. *International Conference on Pattern Recognition* 1, 199–203.
- RAMANAN, D., FORSYTH, D. A., AND ZISSERMAN, A. 2007. Tracking people by learning their appearance. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 1, 65–81.
- RECHENBERG, I. 1973. *Evolutionstrategie: optimierung technischer systeme nach prinzipien der biologischen evolution*. Frommann-Holzboog. (p.93)

- RECHENBERG, I. 1994. *Evolutionsstrategie 94*, Volume 1 of *Werkstatt Bionik und Evolutionstechnik*. Frommann-Holzboog, Stuttgart. (p.93)
- ROBERTS, T. J., MCKENNA, S. J., AND RICKETTS, I. W. 2006. Human tracking using 3d surface colour distributions. *Image and Vision Computing* 24, 12, 1332 – 1342.
- RODRGUEZ-CARRANZA, C., LOEW, M., AND BUTZ, T. 1999. Global optimization weighted mutual information. In *In MICCAI (1999)*, pp. 549–556. Springer-Verlag. (p.144)
- ROS, R. AND HANSEN, N. 2008. A simple modification in cma-es achieving linear time and space complexity. In *PPSN (2008)*, pp. 296–305. (pp.93, 95)
- ROSENBROCK, H. H. 1960. An automatic method for finding the greatest or least value of a function. *The Computer Journal* 3, 3, 175–184. (p.116)
- ROSENHAHN, B. AND BROX, T. 17-22 June 2007. Scaled motion dynamics for markerless motion capture. *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, 1–8.
- ROSENHAHN, B., BROX, T., AND WEICKERT, J. 2007. Three-dimensional shape knowledge for joint image segmentation and pose tracking. *International Journal of Computer Vision* 73, 3, 243–262. (pp.1, 14)
- ROSENHAHN, B., KLETTE, R., AND METAXAS, D. 2007. *Human Motion - Understanding, Modeling, Capture and Animation*, Volume 36 of *Computational Imaging and Vision*. Springer, Dordrecht, The Netherlands.
- ROSENHAHN, B., PERWASS, C., AND SOMMER, G. 2005. Pose estimation of 3d free-form contours. *Int. J. Comput. Vision* 62, 3, 267–289.
- ROWEIS, S. 1998. EM algorithms for PCA and SPCA. In M. I. JORDAN, M. J. KEARNS, AND S. A. SOLLA Eds., *Advances in Neural Information Processing Systems*, Volume 10 (1998). The MIT Press.

- 
- ROWEIS, S. T. AND SAUL, L. K. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 5500 (December), 2323–2326. (p.27)
- SAND, P., MCMILLAN, L., AND POPOVIĆ, J. 2003. Continuous capture of skin deformation. *ACM Trans. Graph.* 22, 3, 578–586. (p.62)
- SCHMIDT, J. AND NIEMANN, H. 2001. Using quaternions for parametrizing 3-d rotations in unconstrained nonlinear optimization. In *VMV '01: Proceedings of the Vision Modeling and Visualization Conference 2001* (2001), pp. 399–406. Aka GmbH. (p.194)
- SCHOLKOPF, B., MIKA, S., SMOLA, A., RAUSCH, G., AND MULLER, K.-R. 1998. Kernel pca pattern reconstruction via approximate pre-images. In *International Conference on Artificial Neural Networks* (1998), pp. 147152. (p.27)
- SEITZ, S. M. AND DYER, C. R. 1997. Photorealistic scene reconstruction by voxel coloring. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)* (Washington, DC, USA, 1997), pp. 1067. IEEE Computer Society.
- SETHIAN, J. A. 1999. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press. (pp.123, 126)
- SHAMS, R. AND BARNES, N. 2007. Speeding up mutual information computation using nvidia cuda hardware. In *Digital Image Computing Techniques and Applications, 9th Biennial Conference of the Australian Pattern Recognition Society on* (Dec. 2007), pp. 555–560.
- SHANG, Y.-W. AND QIU, Y.-H. 2006. A note on the extended rosenbrock function. *Evol. Comput.* 14, 119–126. (p.116)
- SHANNON, C. E. 1948. A mathematical theory of communication. *Bell Systems Technical Journal* 27, 3 (July), 379–423. Continued 27(4):623-656, October 1948.

- SHANNON, C. E. 1949. Communication in the Presence of Noise. *Proceedings of the IRE* 37, 1, 10–21. (p.156)
- SHARMA, G., WU, W., AND DALAL, E. N. 2005. The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research and Application* 30, 1 (Feb.), 21–30. Paper pre-print, spreadsheet, and data available electronically at: <http://www.ece.rochester.edu/~gsharma/ciede2000/>.
- SHI, Y. AND EBERHART, R. C. 1998. Parameter Selection in Particle Swarm Optimization. In *EP '98: Proceedings of the 7th International Conference on Evolutionary Programming VII* (London, UK, 1998), pp. 591–600. Springer-Verlag. (p.92)
- SIDENBLADH, H. AND BLACK, M. J. 2001. Learning image statistics for bayesian tracking. In *International Conference On Computer Vision* (2001), pp. 709–716.
- SIDENBLADH, H. AND BLACK, M. J. 2003. Learning the statistics of people in images and video. *Int. J. Comput. Vision* 54, 1-3, 181–207.
- SIDENBLADH, H., BLACK, M. J., AND SIGAL, L. 2002. Implicit probabilistic models of human motion for synthesis and tracking. In *European Conference on Computer Vision* (2002), pp. 784–800.
- SIDENBLADH, H., DE LA TORRE, F., AND BLACK, M. J. 2000. A framework for modeling the appearance of 3d articulated figures. In *FG '00: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000* (Washington, DC, USA, 2000), pp. 368. IEEE Computer Society.
- SIGAL, L. 2008. Continuous-state graphical models for object localization, pose estimation and tracking. *PhD Thesis*. (p.22)
- SIGAL, L., BALAN, A., AND BLACK, M. 2010. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*. (pp.149, 151, 152)

- 
- SIGAL, L., BHATIA, S., ROTH, S., BLACK, M., AND ISARD, M. 2004. Tracking loose-limbed people. *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on 1*, I-421–I-428 Vol.1. (pp.21, 54)
- SIGAL, L. AND BLACK, M. J. 2006a. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical report, Brown University, Department of Computer Science. (pp.136, 140, 149, 172)
- SIGAL, L. AND BLACK, M. J. 2006b. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on 2*, 2041–2048. (p.21)
- SMINCHISESCU, C., KANAUIA, A., AND METAXAS, D. 2006. Conditional models for contextual human motion recognition. *Comput. Vis. Image Underst.* 104, 2, 210–220.
- SMINCHISESCU, C., KANAUIA, A., AND METAXAS, D. Nov. 2007. Bme : Discriminative density propagation for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 11, 2030–2044. (p.1)
- SMINCHISESCU, C., METAXAS, D., AND DICKINSON, S. 2005. Incremental model-based estimation using geometric constraints. *IEEE Trans. Pattern Anal. Mach. Intell.* 27, 5, 727–738.
- SMINCHISESCU, C. AND TRIGGS, B. 2003. Estimating Articulated Human Motion with Covariance Scaled Sampling. *The International Journal of Robotics Research* 22, 6, 371. (pp.5, 11, 12, 54)
- SOFTKINETIC. 2010. <http://www.softkinetic.net/>.
- STARCK, J., COLLINS, G., SMITH, R., HILTON, A., AND ILLINGWORTH, J. 2003. Animated statues. *Machine Vision Applications* 14, 4, 248–259.
- STARCK, J. AND HILTON, A. 2003a. Model-based multiple view reconstruction of

- people. *IEEE International Conference on Computer Vision (ICCV)*, 915–922. (p. 129)
- STARCK, J. AND HILTON, A. 2003b. Towards a 3D virtual studio for human appearance capture. *IMA International Conference on Vision, Video and Graphics (VVG)*, 17–24.
- STARCK, J. AND HILTON, A. 2005a. Spherical matching for temporal correspondence of non-rigid surfaces. *IEEE International Conference on Computer Vision (ICCV)*, 1387–1394. (p. 129)
- STARCK, J. AND HILTON, A. 2005b. Virtual view synthesis of people from multiple view video sequences. *Graphical Models* 67, 6, 600–620. (p. 129)
- STARCK, J. AND HILTON, A. 2006. Free-viewpoint video for interactive character animation. *Proc. 4th. Symposium on "Intelligent Media Integration for Social Information Infrastructure, Nagoya JAPAN."*, 25–30. (p. 129)
- STARCK, J. AND HILTON, A. 2007a. Surface capture for performance based animation. *IEEE Computer Graphics and Applications* 27(3), 21–31. (p. 55)
- STARCK, J. AND HILTON, A. 2007b. Surface capture for performance based animation. *IEEE Computer Graphics and Applications* 27(3), 21–31. (pp. 17, 129)
- STARCK, J., HILTON, A., AND ILLINGWORTH, J. 2001. Human shape estimation in a multi-camera studio. *British Machine Vision Conference (BMVC)*, 573–582.
- STARCK, J., HILTON, A., AND ILLINGWORTH, J. 2002. Reconstruction of animated models from images using constrained deformable surfaces. *International Conference on Discrete Geometry for Computer Imagery (DGCI)*, 382–391.
- STARCK, J., MILLER, G., AND HILTON, A. 2005. Video-based character animation. *ACM Symposium on Computer Animation (SCA)*, 49–58. (p. 129)
- STARCK, J., MILLER, G., AND HILTON, A. 2006. Volumetric stereo with silhouette and feature constraints. *British Machine Vision Conference (BMVC)* 3, 1189–1198. (pp. 60, 129)

- 
- STAUFFER, C. AND GRIMSON, W. 1999. Adaptive background mixture models for real-time tracking. *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* 2, –252 Vol. 2.
- STEPHENS, C. P. AND BARITOMPA, W. 1998. Global optimization requires global information. *J. Optim. Theory Appl.* 96, 3, 575–588.
- SUDDERTH, E. B., IHLER, A. T., FREEMAN, W. T., AND WILLSKY, A. S. 2003. Non-parametric belief propagation. In *CVPR (1)* (2003), pp. 605–612. (p. 21)
- SZU, H. AND HARTLEY, R. 1987. Fast simulated annealing. *Physics Letters A* 122, 3–4, 157 – 162.
- TANGKUAMPIEN, T. AND SUTER, D. 2006. Real-time human pose inference using kernel principal component pre-image approximations. In *British Machine Vision Conference* (2006). (p. 27)
- THEOBALT, C., AHMED, N., DE AGUIAR, E., ZIEGLER, G., LENSCH, H., MAGNOR, M., AND SEIDEL, H.-P. 2005. Joint motion and reflectance capture for reliable 3d video. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Sketches* (New York, NY, USA, 2005), pp. 73. ACM. (p. 129)
- TIPPING, M. E. 2001. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research* 1, 211–244. (p. 23)
- TIPPING, M. E. AND BISHOP, C. M. 1999. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61, 611–622(12).
- TÖRN, A. AND ZILINSKAS, A. 1989. Global Optimization. *Lecture Notes in Computer Science* 350. (p. 111)
- TROPP, J. AND GILBERT, A. 2005. Signal recovery from partial information via orthogonal matching pursuit. <http://www.dsp.ece.rice.edu/CS/tropp.pdf>. (p. 159)
- TUNG, T., NOBUHARA, S., AND MATSUYAMA, T. 2008. Simultaneous super-

- resolution and 3d video using graph-cuts. In *CVPR* (2008). (p. 60)
- UNSER, M. 2000. Sampling—50 Years after Shannon. *Proceedings of the IEEE* 88, 4 (April), 569–587.
- URTASUN, R., FLEET, D. J., AND FUA, P. 2006a. 3d people tracking with gaussian process dynamical models. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Washington, DC, USA, 2006), pp. 238–245. IEEE Computer Society. (pp. 24, 26)
- URTASUN, R., FLEET, D. J., AND FUA, P. 2006b. Temporal motion models for monocular and multiview 3d human body tracking. *Comput. Vis. Image Understand.* 104, 2, 157–177.
- VIOLA, P. AND WELLS, W. M., III. 1997. Alignment by maximization of mutual information. *Int. J. Comput. Vision* 24, 2, 137–154. (p. 140)
- VLASIC, D., ADELSBERGER, R., VANNUCCI, G., BARNWELL, J., GROSS, M., MATUSIK, W., AND POPOVIĆ, J. 2007. Practical motion capture in everyday surroundings. *ACM Trans. Graph.* 26, 3, 35. (p. 1)
- VLASIC, D., BARAN, I., MATUSIK, W., AND POPOVIĆ, J. 2008. Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph.* 27, 3, 1–9. (pp. 17, 19, 54)
- VLASIC, D., PEERS, P., BARAN, I., DEBEVEC, P., POPOVIĆ, J., RUSINKIEWICZ, S., AND MATUSIK, W. 2009. Dynamic shape capture using multi-view photometric stereo. *ACM Transactions on Graphics* 28, 5, 174. (pp. 17, 20)
- WANG, J. M., FLEET, D. J., AND HERTZMANN, A. 2006. Gaussian process dynamical models. In *Neural Information Processing Systems*, Volume 18 (2006), pp. 1441–1448. (p. 24)
- WANG, J. M., FLEET, D. J., AND HERTZMANN, A. Feb. 2008. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Ma-*



- 
- chine Intelligence* 30, 2, 283–298. (pp. 1, 25)
- WIKIPEDIA. 2011. Wikipedia, the free encyclopedia. <http://www.wikipedia.org>. (pp. 122, 165)
- WRIGHT, J., YANG, A. Y., GANESH, A., SASTRY, S. S., AND MA, Y. 2009. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 2 (February), 210–227. (pp. 155, 156)
- XI, P., SHU, C., AND RIOUX, M. 2007. Principal components analysis of 3-d scanned human heads. In *ACM SIGGRAPH 2007 posters, SIGGRAPH '07* (New York, NY, USA, 2007). ACM. (p. 69)
- XU, X. AND LI, B. 2007. Learning motion correlation for tracking articulated human body with a rao-blackwellised particle filter. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 1–8.