R-Norm: Improving Inter-Speaker Variability Modelling at the Score Level via Regression Score Normalisation

David Vandyke[1], *Michael Wagner*[1,2], *Roland Goecke*[1,2]

[1]Human-Centered Computing Laboratory, University of Canberra, Australia
[2]College of Engineering and Computer Science, Australian National University, Australia
{david.vandyke,michael.wagner}@canberra.edu.au, roland.goecke@ieee.org

Abstract

This paper presents a new method of score post-processing which utilises previously hidden relationships among client models and test probes that are found within the scores produced by an automatic speaker recognition system. We suggest the name *r*-Norm (for Regression Normalisation) for the method, which can be viewed as both a score normalisation process and as a novel and improved modelling technique of inter-speaker variability. The key component of the method lies in learning a regression model between development data scores and an 'ideal' score matrix, which can either be derived from clean data or created synthetically. To generate scores for experimental validation of the proposed idea we perform a classic GMM-UBM experiment employing mel-cepstral features on the 1sp-female task of the NIST 2003 SRE corpus. Comparisons of the *r*-Norm results are made with standard score post-processing/normalisation methods *t*-Norm and *z*-Norm. The *r*-Norm method is shown to perform very strongly, improving the EER from 18.5% to 7.01%, significantly outperforming both *z*-Norm and *t*-Norm in this case. The baseline system performance was deemed acceptable for the aims of this experiment, which were focused on evaluating and comparing the performance of the proposed *r*-Norm idea.

Index Terms: Score Post-Processing, Score Normalisation, Speaker Recognition, Inter-Speaker Variation

1. Introduction

In this paper we introduce a versatile and novel technique for increasing the performance of any speaker recognition system by using information about how a test probe scores against all enrolled client models and how these scores are related. We name the approach *r*-Norm for regression-normalisation, and depending upon the choice of data used in learning the *r*-Norm model it may be viewed as a normalisation method and/or as a performance boosting approach. Twin Gaussian Process Regression [1], a structured learning method, is used to train the *r*-Norm regression model, as described in Section 2.

Reasons for the requirement to normalise the scores output by a system are varied; it may be to achieve speaker and system independent thresholds, to compensate for nuisance variations that are present within the training and testing speech sets, or to adjust for a mismatch of acoustic conditions between these two sets. It may also be that a clever mapping of scores can reliably increase performance across many situations.

Normalisation may occur at the feature level, e.g. feature warping and cepstral mean subtraction [2, 3], or at the model level, e.g. factor-analysis with an eigenchannel space for channel variations [4]. One of the virtues of score normalisation however is that it can be applied to any system, independent of feature and modelling choice. Normalisation of scores remains a standard step even in current best performing system such as those based on factor analysis [5], i-vectors [6] or support-vectors machines [7], despite all of which having modelling methods designed to compensate for the nuisance variations that partially introduce the requirement for normalisation.

Common score normalisation methods apply a standard normal $N(0,1)$ transform. Normalising scores in this approach, working under the assumption that the impostor and target scores are normally distributed, was first proposed in 1988 [8] (*z*-Norm) and is now standard in speech processing. This approach was designed to compensate for inter-speaker variation and was followed by other similar transforms that have proved useful such as test-normalisation (*t*-Norm) [9], and handset-normalisation (*h*-Norm) [10]. Others have been suggested for text-dependent speaker recognition such as *u*-Norm [11], but all of these may be grouped under the theory of a $N(0,1)$ mapping. The proposed regression-normalisation method introduced here is different in implementation, and also in purpose as it aims to increase performance in all circumstances by modelling deeper relationships than these aforementioned normalisation methods.

Most systems assume an equal prior on client speakers and adopt a Bayesian approach for obtaining the posterior probability for the test speech against a client model, as such outputting a likelihood ratio where the numerator is a similarity measure (likelihood of speech data against a client model) and is normalised by a typicality value (likelihood of speech data against a world model). This implicit normalisation is different to the distribution scaling that *z*-Norm and *t*-Norm perform. Score normalisation is fundamentally about changing the relative distributions of impostor and target scores. Approximating the impostor and target score distributions with $N(\mu_i, \sigma_i)$ and $N(\mu_t, \sigma_t)$ Gaussians respectively, then the system Equal-Error Rate (EER) is given by the cumulative standard normal $\Phi(Score_{EER})$ where $Score_{EER} = \frac{\mu_i - \mu_t}{\sigma_i + \sigma_t}$. This Gaussian approximation on the scores is common and well validated experimentally. The aim then is to minimise $Score_{EER}$. We hypothesise that there exists a relationship between the scores of test probes and client models that has not been attempted to be captured yet and we propose the flexible new regression-normalisation method *r*-Norm for adjusting the scores output by a system for the purpose of achieving this aim.

The remainder of the paper is structured as follows: Section 2 introduces and describes the theory and method of the the proposed *r*-Norm technique. In Section 3 standard score normalisation techniques *z*-Norm and *t*-Norm are compared to the proposed regression-normalisation. Section 4 presents results of the proposed technique applied to the NIST 2003 data. These

results are discussed in Section 5, where conclusions, limitations and future work can also be found.

2. R-Norm: Regression Score Post-Processing

We now describe the proposed Regression Score Post-Processing/Normalisation technique *r*-Norm. We are focused on adjusting (distribution scaling and/or normalising) the scores output by an automatic speaker recognition system (verification or identification), which we shall refer to as the raw scores. We assume that scores are organised in a matrix where client models correspond to rows and test probes to columns. To introduce the method we concern our description only with closed-set recognition (any one test probe was uttered by a client for whom we have a model). We have three disjoint sets of data; a training set for estimating client models, a development set scored by the system and these scores used in learning the *r*-Norm model, and an unseen testing data set.

The central concept of *r*-Norm lies in learning a regression model from the development data score matrix $\mathcal{D}$ to a matrix that represents the scores of the development data hypothetically output by an idealised, ultra recogniser[1]. We refer to this matrix as the Ideal matrix, and denote it by $\mathcal{I}$. The regression function that we learn we denote by $\mathbf{r}$.

We use Twin Gaussian Process Regression (TGPR) [1] as the regression model for learning this relationship between $\mathcal{D}$ and $\mathcal{I}$. TGPR is a structured prediction method that firstly builds models for the relationships found within $\mathcal{D}$ and $\mathcal{I}$ separately, before learning the regression function $\mathbf{r}$ between these preliminary models. This is shown in Step 1 of Figure 1.

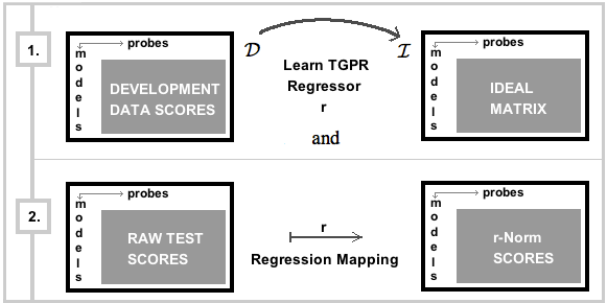

Figure 1: *Schematic outlining the stages of the r-Norm method. In step 1 the Twin Gaussian Process Regression function* $\mathbf{r}$ *is learnt; the arrow here implies capturing the relationship between the development data matrix and the Ideal score matrix. In step 2 the function* $\mathbf{r}$ *is used to map the raw test scores to adjusted r-Norm versions; the arrow here implies a mathematical mapping of scores under the function* $\mathbf{r}$.

By performing this structured prediction we aim to capture any relationships found within the raw scores $\mathcal{D}$ between client models and the scores of a test probe that we postulate exist due to correlations between client models (derived from true similarities in actual speakers voices pending accurate speaker modelling). We then aim to make use of these discovered relationships, held within the regression function $\mathbf{r}$, by mapping raw test scores under $\mathbf{r}$ where these inter-speaker correlations have been accounted for by accentuating target scores and diminishing incorrectly high impostor scores. This mapping is the second and final stage of the *r*-Norm process and is shown in Step 2 of Figure 1. Note that in implementation, in applying the regression function $\mathbf{r}$, we require the test probe to be scored against all client models in order to produce a score vector that is mapped. The *r*-Norm process adjusts the score of an test utterance against a model with reference to how the test probe scores against all other client models of the system. This of course increases online computational time during the verification process in direct proportion to the number of enrolled clients, but in most modern automatic systems implemented by average CPUs the scoring of a single utterance against one model is sufficiently quick that this should not be of large concern. *The r-Norm process is summarised here:*

1. Select an Ideal score matrix $\mathcal{I}$.
2. Learn the TGPR regression function $\mathbf{r}$ from the raw development score matrix $\mathcal{D}$ to the Ideal score matrix $\mathcal{I}$.
3. Map the test scores vector under $\mathbf{r}$ to its *r*-Norm version.

There are no constraints (except dimensions) on the choice of matrix $\mathcal{I}$, but the choice does influence how the *r*-Norm process may be described. If we choose a purely synthetic matrix that scores targets as $+1$ and impostors as 0 then we may view the *r*-Norm method as a score post-processing step for improving recognition performance where the scores of the raw system may be viewed as scalar features for further modelling. Alternatively if the system is to be tested on speech that has different characteristics to that used for training or is challenging in some sense (channels,noise,babble,microphone), then the Ideal score matrix $\mathcal{I}$ may be taken from the scores of clean data (or data matching that used to train speaker models) and the development data should be as similar as possible to the anticipated testing speech. Like this the $\mathbf{r}$-Norm process may be viewed as a compensation and normalisation method. Due to space constraints we investigate only the first viewpoint in this paper, where $\mathcal{I}$ is highly synthetic.

3. Contrasting *r*-Norm with Common Normalisation Techniques

The typical score output from an automatic speaker recognition system is a log likelihood ratio, denoted by φ in Eq. 1 for the score between a client model λ_{client} and a test probe X:

$$\varphi(\lambda_{client}, X) = \frac{P(X \mid \lambda_{client})}{P(X \mid \lambda_{UBM})} \quad (1)$$

As mentioned, a common assumption is that each of the impostor and target score distributions is well approximated by a single Gaussian. Most score normalisation methods aim to adjust either the impostor or target distribution of scores to a standard distribution, commonly a standard $N(0,1)$. Further as it requires much score data to accurately estimate mean and variance statistics most normalisation methods are impostor centric. Common score normalisation methods *z,t,h*-Norm all attempt to scale the impostor score distribution via a standard normal mapping of scores using *a priori* parameters that estimate the raw impostor scores curve. The development data used to learn these *a priori* parameters is dependent on the aims of the normalisation. *Z*-Norm [8] compensates for inter-speaker variation by using estimates of the mean and variance of $\varphi(\lambda_{client}, \cdot)$ to normalise all probe scores against λ_{client}. *T*-Norm [9] aims to compensate for inter-session differences by performing a standard normal mapping of $\varphi(\cdot, X)$ that is based on an *a priori* approximation of the distribution of $\varphi(\cdot, X)$. These two common

[1] Hence $\mathcal{I}$ has the same dimensions as $\mathcal{D}$

approaches are conveyed graphically in Figure 2. The proposed *r*-Norm method contrasts with these approaches (compare to Figure 1) in that it uses the relations found between client models and development probes through analysing scores of a development data set to then adjust the scores of a test utterance against all client models.

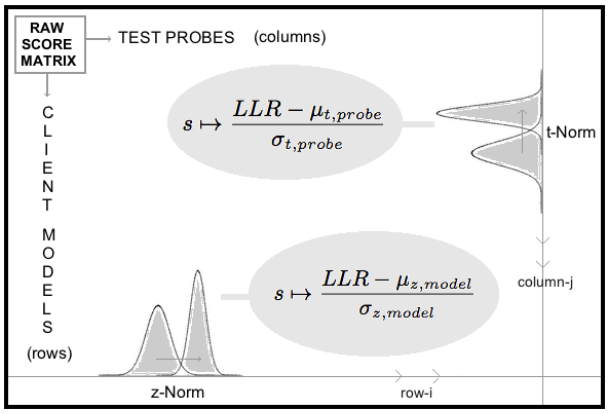

Figure 2: *Outline of the z-Norm and t-Norm score adjustment processes which operate via a standard normal type mapping of scores on a model-by-model and probe-by-probe basis respectively. The purpose of this diagram is to contrast the nature of z-Norm and t-Norm, with the proposed r-Norm method which considers the relationships of scores over the whole matrix.*

We anticipate that these pure normalisation methods *z*-Norm and *t*-Norm are required if implementing *r*-Norm when there is any significant mis-match between the development data used for learning the regression function $\mathbf{r}$ and the anticipated testing data. In such a situation we predict benefits in applying a *z*-Norm mapping (before applying *r*-Norm) using parameters for each client model estimated on data that is similar to that used for regression model development. This remains to be validated experimentally.

Like the other methods mentioned here, **r**-Norm may be useful in a wide range of other pattern recognition domains.

4. Experiments

To empirically test the *r*-Norm idea we required some scores from an automatic speaker recognition system. For these we performed a text-independent speaker verification experiment using a Gaussian Mixture Model (GMM)- Universal Background Model (UBM) system [12] on the 1speaker Female portion of the NIST 2003 SRE data [13]. We used mel-cepstra features, taking the first 12 MFCC plus log energy and appending first order deltas for a 26 dimensional feature vector, extracted from 25ms speech frames incremented by 10ms shifts.

The UBM, which contained 1024 mixtures, was trained by Expectation-Maximisation (EM) [14] on the union of all Female speakers data from the NIST 2000 and 2001 SRE corpora. A fast implementation of the k-means clustering algorithm [15] was used to generate initial estimates of the mixture means. Available computation resources limited us to performing only 10 iterations of EM, and this is the most significant reason for the weak overall performance of the baseline system reported (see Figure 3). We deemed this acceptable for the aims of this investigation; namely to explore how well the proposed *r*-Norm technique could improve recognition accuracy post obtaining the raw scores[2]. These results at a minimum demonstrate the benefit of using *r*-Norm in circumstances where the modelling has been substandard due to the training data or otherwise.

Speaker models for all 207 female NIST 2003 speakers were MAP [16] adapted from the UBM using the single training utterance for each speaker within the corpus. We considered only closed-set speaker verification and thus removed the test utterances not attributed to any of the 207 clients. This left 1899 testing utterances from which the first 1000 where used for development data (learning the *r*-Norm regression model), and the remaining 899 utterances where used for testing. For learning the TGPR model for *r*-Norm we used the MATLAB implementation supplied by the authors of the TGPR method [1]. In this early examination of the *r*-Norm idea we did not perform any parameter search to optimise the TGPR model, employing only the default TGPR parameters given in the code. The authors knowledge of the use of TGPR for image problems (pose estimation and occlusion detection) in computer vision suggests that a parameter search could be beneficial in future.

We explore two *r*-Norm implementations by learning a regression onto two separate Ideal score matrices. The first, Ideal 1, consisted of only 0 impostor scores and 1 target scores (zero-variance distributions). In the second exploration, Ideal 2, the Ideal matrix was based on the actual raw impostor and target score data from the development utterances. The impostor distribution and target distribution means were calculated from these data and Ideal 2 matrix scores were adjusted by adding the impostor mean or target mean for impostor or target scores respectively. This transform resulted in Ideal 2 impostor and target scores that were also entirely separated, but had non-zero variance, unlike in Ideal 1.

A summary of results, reporting Equal-Error-Rates (EER) and minimum Detection Cost Function (DCF) values (using the NIST 2003 DCF parameters), is given in Table 1.

Table 1: *EER and minDCF for each normalisation method.*

Normalisation Method	EER	min. DCF-2003
none	18.8%	0.061
z-Norm	18.2%	0.068
t-Norm	19.6%	0.069
r-Norm: Ideal 1	**7.01%**	**0.030**
r-Norm: Ideal 2	9.3%	0.036

The disjoint data used for *z*-Norm utterances and *t*-Norm GMM model building was taken from Female NIST 2000 SRE speakers. We use 110 utterances for *z*-Norm and train 60 speaker models for *t*-Norm. We would expect better *z*-Norm and *t*-Norm results with a larger number of utterances and models respectively [5], however computational resources restricted us to these numbers.

Detection Error Trade-off (DET) curves are shown in Figure 3. The performance of *r*-Norm, in improving the EER from 19% to 7% using the Ideal 1, zero-variance $\mathcal{I}$ distributions, is shown to be very promising. Note that both applied *r*-Norms (with Ideal 1 and Ideal 2 matrices) significantly outperformed *z*-Norm and *t*-Norm in this instance.

The effect of *r*-Norm (with Ideal 1) on the bimodal distribution of impostor and target scores is shown in Figure 4.

[2] A similar experiment was performed on the small and clean ANDOSL speaker recognition corpus using a well trained UBM trained on a disjoint section of the data. An EER of $< 1\%$ was achieved and *r*-Norm made the results no worse.

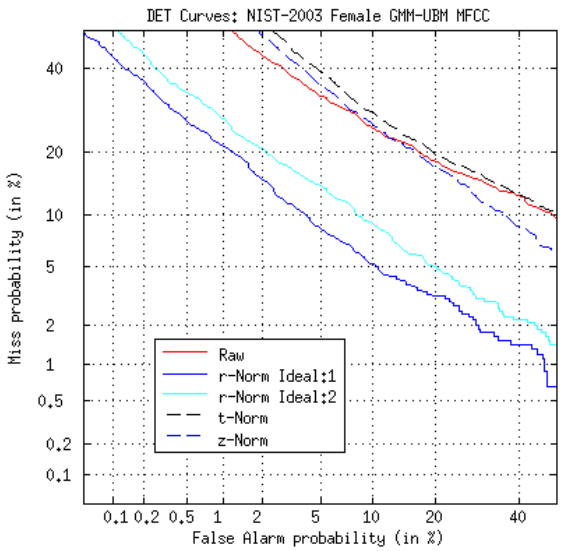

Figure 3: *DET plots for raw (red) and z,t and r-Norm scores. R-Norm is shown to improve system performance significantly with both Ideal matrices. The baselines weak performance was due to the use of a poor UBM as explained in Section 4.*

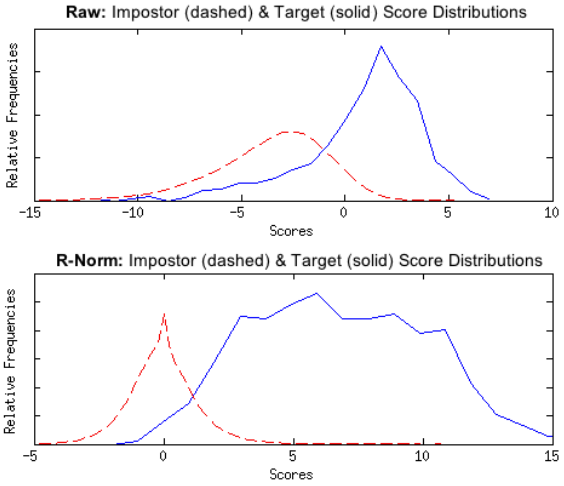

Figure 4: *The effect of r-Norm (Ideal 1) on Impostor and Target score distributions is shown via relative frequency histograms.*

5. Discussion

The proposed *r*-Norm score post-processing step has been shown to perform very strongly on the NIST 2003 Female SRE data. Using the function **r** learnt on the Ideal 1 matrix, which contained only two values, 1 for target scores and 0 for impostor scores, reduced the EER to 7.01%. Learning the TGPR regression function **r** on the Ideal 2 matrix which represented well separated target and impostor score distributions but with non-zero variance reduced the EER to 9.3%. Both of these results were significantly better than the compared normalisation methods *z*-Norm and *t*-Norm. It must be noted however that we expect these methods to perform better if a larger number of *t*-Norm speaker models where built, or *z*-Norm utterances used [5, 7], however nowhere in the literature have the authors found these methods to increase system performance as significantly as the observed *r*-Norm results here.

There are choices in implementing the *r*-Norm process as to what the development data used for creating the raw score matrix $\mathcal{D}$ and what the Ideal score matrix $\mathcal{I}$ should be, and these should be informed by both the nature of the testing data and what the aims in applying *r*-Norm are.

As mentioned the experiments performed here have used the *r*-Norm method from the viewpoint of score post-processing to improve recognition rates. The testing data, whilst completely disjoint from training and development data, presumably shared acoustic characteristics with the development data that generated the raw score matrix that the TGPR function **r** was learnt on. Future work in developing further the *r*-Norm method and demonstrating it experimentally should focus on cases where there is no *a priori* information as to what the characteristics of the testing speech will be, necessitating that the development data set should be large and acoustically varied, and/or that the Ideal matrix should be representative of a *z*-Norm mapped score matrix and that the test scores should undergo *z*-Norm before applying *r*-Norm. There are many mismatch scenarios that have several choices for combinations of development data and Ideal matrix. In each case there exist theoretically justifiable reasons for the choices of $\mathcal{D}$ and $\mathcal{I}$ and they remain to be tried experimentally.

The *r*-Norm method may also focus on pure normalisation alone, where the emphasis is not on boosting system performance by capturing correlations between client models and test probe scores that relate to inter-speaker variability, but on compensating and overcoming mismatch conditions between training and testing. A potential configuration of the *r*-Norm system for dealing with large differences between training and testing speech could be selecting the development data used in forming the raw score matrix $\mathcal{D}$ to match as well as possible the anticipated testing data type and basing the Ideal score matrix on scores derived from clean data (or data well matching that used to train client models). This, due to space, is left for future work.

The interpretation of the likelihood ratio after regression normalisation is perhaps a larger issue than with zero-norm and test-norm. It remains to conclude whether it may be interpreted still as a likelihood ratio or simply as a score, although this is much less of an concern for automatic speaker recognition systems and more for forensic voice comparison, where with some calibration [17] it may again be interpretable as such. Also extending to open set verification is not conceptually difficult but remains to be explored.

Finally it remains to be tested how the proposed method improves the accuracy of sophisticated automatic systems. [18] suggests that score normalisation is not a factor in the performance of advanced speaker recognition systems. This is from a normalisation perspective however, as these systems have modelling methods to cope and adjust for nuisance variations that give reason to the requirement for score normalisation. The *r*-Norm approach, viewing it as a post-score modelling methodology by using a synthetic Ideal score matrix that is designed to leverage inter-speaker differences, should still have a purpose here. It remains to test *r*-Norm on well trained JFA and i-vector systems on recent years NIST SRE corpora in order to draw any conclusions on this point.

Encouraged by these first results there remains much to explore regarding the proposed regression-normalisation, score post-processing concept *r*-Norm.

6. References

[1] L. Bo and C. Sminchisescu, "Twin gaussian processes for structured prediction," *International Journal of Computer Vision*, vol. 87, no. 1-2, pp. 28–52, 2010. [Online]. Available: www2.maths.lth.se/matematiklth/personal/sminchis/code/TGP.html

[2] C. Barras and J.-L. Gauvain, "Feature and score normalization for speaker verification of cellular data," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, april 2003, pp. II – 49–52 vol.2.

[3] D. Wu, B. Li, and H. Jiang, *Speech Recognition, Technologies and Applications: Normalisation and Transformation Technologies for Robust Speaker Recognition*, F. Mihelic and J. Zibert, Eds. intechopen, 2008.

[4] P. Kenny, "Joint factor analysis of speaker and session variability: Theory and algorithms," in *Technical Report: CRIM*, 2006. [Online]. Available: http://www.crim.ca/perso/patrick.kenny

[5] P. Kenny, N. Dehak, P. Ouellet, V. Gupta, and P. Dumouchel, "Development of the primary crim system for the nist 2008 speaker recognition evaluation," in *INTERSPEECH*, 2008, pp. 1401–1404.

[6] M. Senoussaoui, P. Kenny, N. Dehak, and P. Dumouchel, "An i-vector extractor suitable for speaker recognition with both microphone and telephone speech," 2010.

[7] N. Brummer, L. Burget, P. Kenny, P. Matjka, E. V. de, M. Karafit, M. Kockmann, O. Glembek, O. Plchot, D. Baum, and M. Senoussauoi, "Abc system description for nist sre 2010," in *Proceedings of the NIST 2010 Speaker Recognition Evaluation*. National Institute of Standards and Technology, 2010, pp. 1–20.

[8] K.-P. Li and J. Porter, "Normalizations and selection of speech segments for speaker recognition scoring," in *International Conference on Acoustics, Speech, and Signal Processing*, 1988, pp. 595–598 vol.1.

[9] R. Auckenthalera, C. Michael, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, pp. 42–54, 2000.

[10] D. A. Reynolds, "Comparison of background normalization methods for text-independent speaker verification," in *EUROSPEECH*, 1997.

[11] D. Garcia-Romero, J. Gonzalez-Rodriguez, J. Fiérrez-Aguilar, and J. Ortega-Garcia, "U-norm likelihood normalization in pin-based speaker verification systems," in *AVBPA*, 2003, pp. 208–213.

[12] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," in *Digital Signal Processing*, 2000, pp. 19–41.

[13] (Speaker Recognition Evaluations) National Institute of Standards and Technology. [Online]. Available: http://www.itl.nist.gov/iad/mig//tests/spk/

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[15] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

[16] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291 –298, apr 1994.

[17] N. Brummer, "Measuring, refining and calibrating speaker and language information extracted from speech," Ph.D. dissertation, University of Stellenbosch, October 2010.

[18] N. Brummer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. van Leeuwen, P. Matejka, P. Schwarz, and A. Strasheim, "Fusion of heterogeneous speaker recognition systems in the stbu submission for the nist speaker recognition evaluation 2006," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2072–2084, sept. 2007.