FOUR WEIGHTINGS AND A FUSION: A CEPSTRAL-SVM SYSTEM FOR SPEAKER RECOGNITION

Sachin S. Kajarekar

Speech Technology and Research Laboratory
SRI International, Menlo Park, CA, USA
sachin@speech.sri.com

ABSTRACT

A new speaker recognition system is described that uses Mel-frequency cepstral features. This system is a combination of four support vector machines (SVMs). All the SVM systems use polynomial features and they are trained and tested independently using a linear inner-product kernel. Scores from each system are combined with equal weight to generate the final score. We evaluate the combined SVM system using extensive development sets with diverse recording conditions. These sets include NIST 2003, 2004 and 2005 speaker recognition evaluation datasets, and FISHER data. The results show that for 1-side training, the combined SVM system gives comparable performance to a system using cepstral features with a Gaussian mixture model (baseline), and combination of the two systems improves the baseline performance. For 8-side training, the combined SVM system is able to take advantage of more data and gives a 29% improvement over the baseline system.

1. INTRODUCTION

A commonly used baseline system in speaker recognition uses Mel-frequency cepstral coefficients (MFCC) in a Gaussian mixture model (GMM) framework [1]. Such a system is used to measure the effectiveness of novel features and modeling approaches. Research on novel features has ranged from transformations of cepstral features to discrete high-level features modeling prosodic events [2]. Modeling approaches have been of two types: generative models like GMMs [1], and discriminative models such as support vector machines (SVMs) [2-5].

This work investigates an SVM-based approach for modeling cepstral features. It is motivated by the generalized linear discriminant sequential (GLDS) kernel approach proposed by Campbell [6]. The GLDS approach uses polynomial features, modeling higher-order moments, derived from MFCCs. These features or transformations of these features are used in an SVM, and the resulting system was shown to outperform a baseline system.

For the speaker recognition task, the SVM is trained to classify between features for impostors and a target speaker; there are more instances of impostors (on the order of thousands) than of true speakers (up to 8). When polynomial features (on the order of tens of thousands) are used as features with an SVM, a peculiar situation arises. Since there are more features than impostor speakers, the distribution of features in a high dimensional space lies in a lower dimensional subspace spanned by the background (or impostor) speakers. This lower dimensional subspace is referred to as the *background subspace*. A subspace orthogonal to this subspace captures all the variation in the feature space not observed between the background speakers. This is called the *background-complement subspace*. It is evident that these two subspaces have different characteristics for speaker recognition.

When using polynomial features, the SVM is typically trained in high-dimensional feature space. However, if the background and background-complement subspaces have different characteristics, the classification can be divided into two parts: (1) classification based on information in the background subspace and (2) classification based on the background-complement subspace. Researchers have studied the background subspace and found improvements using its transformations [7, 8]. However, no such work has been done investigating the background-complement subspace. As we will show, the combination of two decisions based on the two subspaces can do better than a single decision based on the complete feature space.

This paper is organized as follows. Section 2 describes the datasets used for the experimentation. Section 3 describes the baseline system in brief. Section 4 describes the combined SVM system and Section 5 compares the results of this system with the baseline. The paper concludes with a summary in Section 6.

2. DATASETS

We used five different databases for the experiments: (1) NIST 2002 cellular speaker recognition evaluation (SRE) (Switchboard cellular), (2) NIST 2003 extended data SRE (Switchboard-II, landline) [9], (3) FISHER [10], (4) NIST 2004 SRE (Mixer) [9], and (5) NIST 2005 SRE data (Mixer) [9]. Note that all the databases contain conversational speech recorded over telephone (landline or cellular) networks. The first dataset is used exclusively as a part of the background[1] dataset. The next two datasets are divided into a background

[1] Background data refers to the data used to train the background model in a GMM system. The same data is used as impostor data to train an SVM system.

dataset and the development dataset. The last two datasets are used exclusively for evaluation of systems. Table 1 shows the statistics of the speaker models and trials (model-test pair) in each dataset.

When training a system, we use a common background (or impostor) speaker set across all datasets. We report results on a 1 conversation side (1side) and 8 conversation side (8side) training conditions, and a 1 conversation side testing condition. Typically a conversation side contains 2.5 minutes of speech. The scores from a system are normalized using TNORM [11], where TNORM speakers are chosen specific to the dataset (as described in the following subsections). Overall, we have five results for the 1side-1side condition and four results for the 8side-1side condition.

2.1 NIST 2003 extended SRE data

We used version 1 of the control file according to the NIST evaluation specification. Each conversation side in this set is about 2.5 minutes (excluding silence). In the original setup, the data was divided into 10 non-overlapping splits. Typically, the splits are divided into two sets: 1-5 and 6-10. Data for training the background model and for score normalization is taken from one set and evaluation is performed on the other set.

We modified the data as follows. First we removed conversations from the speakers who were used to train the automatic speech recognition system[2]. Then, we took two splits each from these two sets, such that the performance of these sets before and after removing the splits is similar. These four splits were used to train the background model. The remaining three splits from each set were used to evaluate the systems. While evaluating a split, TNORM speakers are chosen from the other split.

2.2. Fisher

The Fisher dataset is created from a subset of the Fisher [10] database, collected and distributed by LDC. We selected two sets of speakers from this data; one set where speakers participated in a single conversation and a second set where speakers participated in more than one conversation. Each set has an equal proportion of speakers with different genders and an equal proportion of different handsets (electret, carbon-button, cell-phone) used in the conversation. The first set is used to create the background model for the different systems and the second set is divided equally into two splits of 249 speakers each.

An evaluation set is created from these splits for the 1 and 2 conversation side training conditions. The trials are chosen as follows. First all the speakers are scored against all the test data. All the true speaker trials are preserved. Impostor trials are obtained by evenly sampling the overall impostor score distribution. Only the data for 1 conversation side training is used as a development set (devset) for the evaluation. Each conversation in the evaluation set is about 2.5 minutes (excluding silence) and each conversation in the background set is about 5 minutes (excluding silence). When evaluating on a split, TNORM speakers are obtained from the other split.

2.3 NIST 2004 and 2005 SRE data

This data was distributed as a part of the NIST evaluations [9]. It is a part of the Mixer collection [12]. The data was collected in different languages and in different recording conditions. In this paper, we report results on the English-only trial condition for the SRE 2004 dataset and for common-condition trials for the SRE 2005 dataset. For both datasets, TNORM is performed using speakers from the Fisher dataset.

Table 1 Model and trial statistics for different datasets

Data	#Train sides	Split	**Referred to as**	Models	Trials
NIST 2003	1side	1	**e03-1s-1**	578	9765
		2	**e03-1s-2**	585	9977
	8side	1	**e03-8s-1**	546	4911
		2	**e03-8s-2**	559	5298
NIST 2004	1side	1	**e04-1s**	479	15317
	8side	1	**e04-8s**	225	7336
NIST 2005	1side	1	**e05-1s**	598	20907
	8side	1	**e05-8s**	464	16053
Fisher	1side	1	**fsh-1s-1**	734	16578
		2	**fsh-1s-2**	702	14598

3. BASELINE SYSTEM

This system uses 13 Mel frequency cepstral coefficients (MFCCs). They were estimated by a 300-3300 Hz bandwidth front end consisting of 19 Mel filters. Cepstral features are normalized using cepstral mean subtraction (CMS), and are appended with delta, double-delta, and triple-delta coefficients. For channel normalization, the feature transformation described in [1] is applied to the features.

Our baseline uses 2048 Gaussian components for the background model. This GMM is trained using gender and handset-balanced data (electret, carbon-button, cell-phone). We use approximately 300 hours of data from FISHER, part of the NIST 2003 extended SRE data, and the NIST 2002 cellular SRE development data.

Target GMMs are adapted from the background GMM using MAP adaptation of the means of the Gaussian components. Verification is performed using 5-best Gaussian components per frame selected with respect to the background model scores. **Error! Reference source not found.** shows the performance of this system on different datasets.

4. CEPSTRAL SVM SYSTEM

This system is an equally weighted combination of the outputs from four SVM systems (referred to as "combined SVM" system). The four systems use a linear kernel and a cost function that makes false rejection 500 times more costly than false acceptance. To train any SVM system, we use 1673 unique speakers from the background set as impostor data points.

[2] The systems described in this paper do not use ASR output but some other SRI systems do use it. The purpose of removing speakers is to make a consistent dataset for evaluating all the different systems, which may or may not use ASR output.

These four systems use the same polynomial features that are derived from the basic features used in the baseline system. They are estimated as shown in Figure 1. The basic features are 13 MFCCs with delta and double-delta coefficients. This vector is processed using CMS and feature transformation to mitigate the effects of handset variations. The transformed vector is appended with second- and third-order polynomial[3] coefficients. The resulting vector has 11479 dimensions and is referred to as the polynomial feature vector. Finally, we estimate the mean and standard deviations of these features over a given utterance.

Two systems use mean polynomial (MP) feature vectors as follows. Principal component analysis (PCA) [13] is performed on the polynomial features for the background speaker utterances. A mentioned earlier, the number of features (F=11479) is much larger than the number of impostor speakers (S=1673). The distribution of high-dimensional features lies in a lower dimensional speaker subspace. Only S-1 leading eigenvectors (also referred to as principal components, PCs) have non-zero eigenvalues. The remaining F-S+1 eigenvectors have zero eigenvalues.

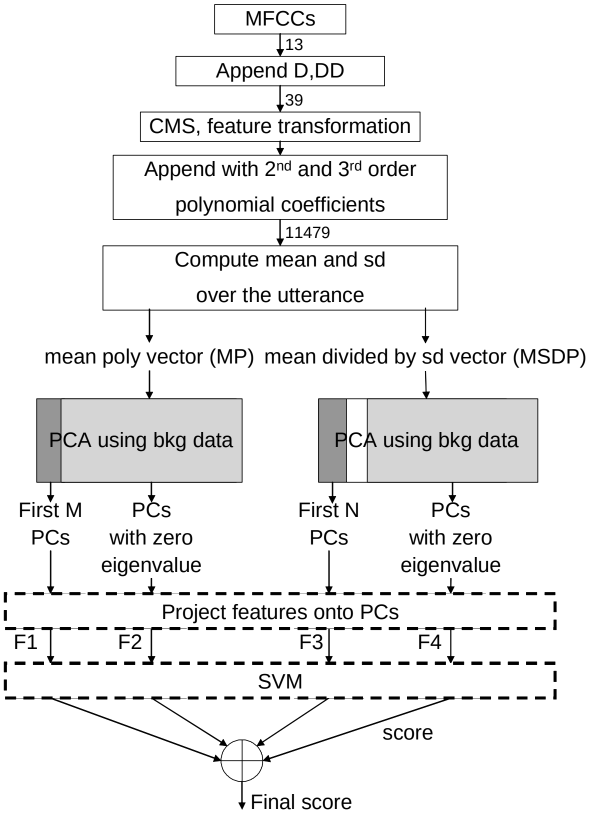

Figure 1 Flowchart of the SVM feature extraction and score combination strategy.

The leading eigenvectors are normalized by the corresponding eigenvalues. All the leading eigenvectors are selected because the total variance is distributed evenly across them. The mean polynomial features are projected onto the normalized S-1 eigenvectors, and the resulting coefficients are used in one SVM system. Similarly, the mean polynomial vectors are projected onto the remaining F-S+1 unnormalized eigenvectors and the resulting coefficients are used in the second SVM system.

In the remaining two systems, we modify the kernel to include the confidence estimate obtained from the standard deviation. If $\overline{X}$ and $\overline{Y}$ are two mean polynomial vectors, the kernel used in the first two systems can be described as

$$k\left(\overline{X},\overline{Y}\right)=\overline{X}^T\overline{Y}=\sum \overline{x}_i\,\overline{y}_i\ .$$

We modified this kernel to be

$$k\left(\overline{X},\overline{Y}\right)=\sum \frac{\overline{x}_i}{\sigma_{x_i}}\frac{\overline{y}_i}{\sigma_{y_i}}=\overline{X}_1^T\overline{Y}_1$$

This implies that the inner product is scaled by the standard deviation of the individual features, where the standard deviation is computed separately over each utterance. Instead of modifying the kernel, we modify the features by obtaining a new feature vector that is the mean polynomial vector divided by the standard deviation polynomial vector (MSDP).

PCA is performed on the MSDP as described earlier. We get two sets of eigenvectors. The first set corresponds to nonzero eigenvalues and second set corresponds to zero eigenvalues. In the first set, the eigenvalues are not spread evenly as they are for mean polynomial vectors. This is due to the scaling by the standard deviation terms. We keep only the first 500 leading eigenvectors (corresponding to 99% of the total variance) and use coefficients obtained from them in the first system. The second system uses as features the coefficients obtained using the trailing eigenvectors corresponding to zero eigenvalues.

The four systems are trained separately. Scores from the four systems are summed with equal weights to produce the final scores.

4.1 Implementation

The background and background-complement transforms are estimated as follows. As mentioned earlier, the covariance matrix from the features (F) for background speakers (S) is a low-rank matrix. Its rank is S-1. Instead of performing PCA in feature space, we perform PCA in speaker space. This is analogous to kernel PCA. Then we transform the S-1 kernel PCs to the corresponding PCs in feature space. They are divided by the eigenvalues to get (S-1)*F background transforms.

The background-complement transform is implemented implicitly. A given feature vector is projected onto the eigenvectors of the background transform. The resulting coefficients are used to reconstruct the feature vector in the original space. The difference between the original and reconstructed feature vectors is used as the feature vector in the background-complement subspace. Note that this is F-dimensional subspace.

An interesting property of the background-complement subspace is that all the feature vectors corresponding to the background speakers get mapped to the origin. Therefore, SVM training is very easy. The origin is a single impostor data

[3] The second order polynomial of X=[x_1 x_2] is

poly(X,2)=[X x_1^2 $x_1 x_2$ x_2^2]

and the third order polynomial is

poly(X,3)=[p(X,2) x_1^3 $x_1^2x_2$ $x_1x_2^2$ x_2^3].

point (irrespective of the number of impostors) and one or more transformed feature vectors from the target training data are the true speaker data points. This is very different from the training in the background speaker subspace, where there are S impostor data points and one or more target speaker data points.

5. RESULTS

Table 3 shows the performance of the three systems using MP vectors and their combination. The performance is reported in terms of equal error rate (EER). This is the point on the detection error tradeoff (DET) curve – false acceptance vs. false rejection – where the two errors rates are equal. Later, results are also presented in terms of minimum value of the detection cost function (DCF) as defined by NIST [9]. Although the results are presented on 4 different datasets, the latest results are on NISR 2004 and 2005 SREs. These columns are highlighted for the better readability. In addition, the results from individual experiments are differentiated from the combination experiments using *italic* font for the later.

Note that all the combinations of the SVM systems are performed at the score level using equal weights. Combinations of SVM and GMM systems are performed using a neural network combiner as explained in the next subsection.

Results show that systems using the background transform (A1) and background complement transform (A2) perform comparably to each other, specifically A2 performing better than both A1 and the system using original features (A0) in 8side training. The combination of the systems using the transform (A3=A1+A2) gives an average performance of both systems, which is worse than A0 in the 1side training and better than A0 in the 8side training. The combination of all three systems (A4=A0+0.5*(A1+A2)) gives improvement over A0 and A3. This is a surprising result because systems A1 and A2 use subspaces of the original feature space used in A0. However, they appear to be making complementary decisions.

Table 4 shows the performance of systems using MSDP and their combinations. Results show that the system using the background transform (B1) is better than the system using the background complement transform (B2) on 1side training and vice versa for 8side training. Unlike MP, the combination of B1 and B2 (B3) outperforms the system using original features (B0). Finally, when B0 and B3 are combined (B4), no significant gains are observed. This shows that classification in subspaces is better than the complete space for MSDP. Note that the difference between A3 and B3 is that A1 uses all the eigenvectors using non-zero eigenvalues and B1 uses only the 500 leading eigenvectors.

Table 5 shows various combinations using the A and B systems. A0 and B0 perform comparably and combining them (C0) does improve performance. Since A0 combined well with A3 (A1+A2), we combine it with B3 to get C1. This gives a small improvement over either one of them. Finally, the combination of A3 and B3 into C2, gives the best overall performance and is either the same as or better than the component systems.

5.1 Comparison and combination with baseline

Table 6 shows the performance of the combined SVM system with the baseline and their combination. The combination is performed using a neural network combiner (LNKnet software [14]). The combiner is trained with two layers (without a hidden layer) and a sigmoid nonlinearity at the output layer. The training priors are optimized for the NIST DCF. One combiner is trained on the Fisher dataset and tested on SRE 2004 data and another combiner is trained on SRE 2004 data and is tested on SRE 2005 data. The same combiner is used for SRE 2004 1side and 8side conditions, but different combiners are used for SRE 2005, 1side and 8side conditions.

Results for e04 show that the combined SVM system (C2) performs comparably to the baseline for the 1side training condition. The combination improves on the individual performances both in DCF and in EER. For the 8side training condition, there is a different trend. The combined SVM system does significantly better than the baseline. Further, combination with the baseline does not give any improvement.

These observations are consistent on e05 data, except that combination of combined SVM and baseline for 8side training condition results in a small improvement in DCF. This difference can be attributed to the fact that the combiner for e04-8s was trained using fsh-1s-1 but the combiner for e05-8s was trained using e04-8s. Therefore, the training data was better matched to e05 than to e04.

Results for the combined SVM system and baseline are analyzed further in Table 7. The results are presented on the SRE 2005 8side training condition because the most improvement is observed for 8side training condition and the SRE 2005 8side condition has the highest number of trials across all 8side conditions. Note that results reported as e05-8s are on the common condition, where the training and test language is English and a certain restriction is placed on the type of handset. For this analysis, we include all English language trials. That gives us 20856 trials, which is more than what is shown in Table 1.

Table 7 shows the results for different sets – for each gender, for same channel type[4], and for a mix of channels. Results show a consistent improvement of 29% for both genders using the SVM system. For different channel combinations, the SVM system gives around 22% improvement for the same channel condition and 29% improvement for mismatched channel condition.

6. SUMMARY AND CONCLUSIONS

We have described a new approach for using cepstral features in an SVM framework. It consists of training four SVMs using four different features derived from cepstral features. The final output is generated by combining their outputs with equal weights.

[4] Channel type and gender were determined by an automatic gender and handset detector. The detector is similar to the baseline GMM system.

Table 2 Baseline system performance

System	%EER									
	e03-1s-1	e03-1s-2	e03-8s-1	e03-8s-2	fsh-1s-1	fsh-1s-2	e04-1s	e04-8s	e05-1s	e05-8s
Baseline	4.63	4.74	1.92	2.05	4.57	3.99	**7.77**	**4.95**	**7.48**	**4.83**

Table 3 Performance of the systems and combinations, using mean polynomial vector

System		%EER									
Index	Using mean polynomial vector	e03-1s-1	e03-1s-2	e03-8s-1	e03-8s-2	fsh-1s-1	fsh-1s-2	e04-1s	e04-8s	e05-1s	e05-8s
A0	original	5.92	6.00	1.44	1.62	5.37	4.22	**9.84**	**4.37**	**x**	**x**
A1	after background transform	6.97	6.83	1.28	1.42	5.50	4.52	**9.84**	**4.52**	**10.2**	**4.25**
A2	after background complement transform	6.73	7.06	1.12	1.32	5.97	5.50	**11.52**	**3.94**	**11.08**	**3.83**
A3	A1+A2	*6.61*	*6.68*	*1.23*	*1.32*	*5.50*	*4.52*	***9.77***	***4.37***	***9.22***	***3.59***
A4	A0+0.5*(A1+A2)	*5.19*	*5.15*	*1.06*	*1.22*	*4.84*	*4.14*	***8.57***	***3.49***	***x***	***x***

Table 4 Performance of the systems and combinations, using mean divided by standard deviation polynomial vector

System		%EER									
Index	Using mean_div_std polynomial vector	e03-1s-1	e03-1s-2	e03-8s-1	e03-8s-2	fsh-1s-1	fsh-1s-2	e04-1s	e04-8s	e05-1s	e05-8s
B0	original	5.32	5.57	1.55	1.52	5.24	4.22	**9.14**	**4.08**	**x**	**x**
B1	after background transform	5.38	6.06	1.71	1.76	4.77	4.14	**9.28**	**4.32**	**8.05**	**4.25**
B2	after background complement transform	5.89	6.33	1.39	1.42	5.77	4.82	**10.61**	**3.94**	**11.08**	**3.83**
B3	B1+B2	*4.45*	*4.95*	*1.23*	*1.47*	*4.58*	*3.77*	***8.29***	***3.35***	***7.49***	***3.29***
B4	B0+0.5*(B1+B2)	*4.50*	*4.86*	*1.38*	*1.32*	*4.44*	*3.84*	***8.22***	***3.35***	***x***	***x***

Table 5 Performance of the combination with systems using both types of polynomial vectors

Systems		%EER									
		e03-1s-1	e03-1s-2	e03-8s-1	e03-8s-2	fsh-1s-1	fsh-1s-2	e04-1s	e04-8s	e05-1s	e05-8s
C0	A0+B0	5.56	5.62	1.49	1.51	5.24	4.06	**9.14**	**4.08**	**x**	**x**
C1	A0+B3	4.96	4.98	1.17	1.27	4.64	4.14	**8.71**	**3.35**	**x**	**x**
C2	A3+ B3	4.38	4.51	1.06	1.17	4.31	3.99	**8.01**	**3.35**	**7.26**	**3.05**

Table 6 Combination results with the baseline system

Training Split	Test Split	Systems	%EER	DCF (x10)
x	x	Baseline	7.73	0.311
		C2	8.01	0.313
fsh-1-1	e04-1s	Baseline + C2	**6.54**	**0.281**
x	x	Baseline	4.96	0.211
		C2	3.35	0.126
fsh-1-1	e04-8s	Baseline + C2	**3.94**	**0.143**

Training Split	Test Split	Systems	%EER	DCF (x10)
x	x	Baseline	7.48	0.253
		C2	7.26	0.272
e04-1s	e05-1s	Baseline + C2	**5.73**	**0.215**
x	x	Baseline	4.83	0.164
		C2	3.05	0.112
e04-8s	e05-8s	Baseline + C2	**3.11**	**0.102**

Table 7 %EER for different conditions from SRE 2005 8side training condition. The results are for all English trials.

Cepstral System	ALL	Gender		Channel	
		M	F	Same	Different
GMM	5.03	4.73	5.18	1.07	7.08
Combined SVM	3.54	3.34	3.63	0.83	4.99

The four SVM systems use a linear kernel and differ in the types of features. Two of the systems use transformed features from the mean polynomial vectors and the other two use transformed features from the mean polynomial vector divided by the standard deviation polynomial vector. The two transformations are obtained using features for the set of background speakers. The idea behind the transformations is based on the observation that the number of speakers is much smaller than the number of features. Therefore the classification in original feature space can be divided into two subspaces: (1) a subspace modeling the variation across background speakers, and (2) the subspace orthogonal to it.

In the case of cepstral features, these two subspaces have different properties as seen in the combination of these systems. Although the systems using subspace projections of the mean polynomial vector give similar performance, they provide improved performance both in their combination and in combination with the original system. The systems using subspace projections of the mean polynomial vector divided by the standard deviation polynomial vector are also complementary to each other, and their combination performs significantly better than the original system. Finally, the four systems using the subspace transformations produced complementary output so that their equally weighted combination gave the best overall performance.

An interesting result is obtained in the background-complement subspace. As mentioned earlier, the feature vectors for background speakers all map to the origin. The classification is performed with one impostor data point and 1 or 8 target data points. Considering the simplistic features used in this system, it is remarkable that it performs comparably to the system using a subspace spanned by features from background speakers. This shows the importance of the characteristics not modeled by background speakers. More work is needed to understand these characteristics and to exploit them for speaker recognition.

The combination of the final result with a state-of-the-art baseline system showed different trends for 1side and 8side training conditions. For the 1side training, the combined SVM system performed comparably to the baseline gave significant improvements in combination. For 8side training, the combined SVM system performed significantly better than the baseline and gave a small improvement in combination with the baseline.

The combined SVM system was tested extensively using various databases recorded in a variety of channel conditions. This gave five results for 1side and four results for 8side training. The consistent performance of the combined SVM system across all datasets shows that this is a promising approach for speaker recognition.

7. ACKNOWLEDGMENTS

Author thanks SRI colleagues for useful discussions. This work was funded by a DoD KDD award via NSF IRI-9619921. The views herein are those of the authors and do not reflect the views of the funding agencies.

REFERENCES

[1] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker Verification Using Adapted Mixture Models," in *Digital Signal Processing*, vol. 10: Academic Press, 2000, pp. 181--202.

[2] E. Shriberg, L. Ferrer, A. Venkataraman, and S. Kajarekar, "SVM Modeling of SNERF-grams for Speaker Recognition," presented at ICSLP, Jeju Island, South Korea, 2004.

[3] W. M. Campbell, J. P. Campbell, D. A. Reynolds, D. A. Jones, and T. R. Leek, "High-Level Speaker Verification with Support Vector Machines," presented at ICASSP, Montreal, 2004.

[4] A. Hatch, B. Peskin, and A. Stolcke, "Improved Phonetic Speaker Recognition using Lattice Decoding," presented at ICASSP, Philadenphia, 2005.

[5] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman, "MLLR Transforms as Features in Speaker Recognition," presented at Eurospeech, Lisbon, Portugal, 2005.

[6] W. M. Campbell, "Generalized Linear Discriminant Sequence Kernels for Speaker Recognition," presented at ICASSP, Orlando, 2002.

[7] A. Solomonoff, C. Quillen, and W. M. Campbell, "Channel Compensation for SVM Speaker Recognition," presented at Odyssey: The Speaker and Language Recognition Workshop, Toledo, Spain, 2004.

[8] O. Thyes, R. Kuhn, P. Nguyen, and J. C. Junqua, "Speaker Idenitification and Verification using Eigenvoices," presented at ICSLP, Beijing, China, 2000.

[9] NIST, "www.nist.gov/speech."

[10] LDC, "www.ldc.upenn.edu."

[11] R. Auckenthaler, E. S. Parris, and M. J. Carey, "Improving a GMM Speaker Verification System by Phonetic Weighting," presented at Proc. of ICASSP, Phoenix, Arizona, 1999.

[12] A. Martin, D. Miller, M. Przybocki, J. Campbell, and H. Nakasone, "Conversational Telephone Speech Corpus Collection for the NIST Speaker Recognition Evaluation 2004," presented at IAD, 2004.

[13] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. Wiley, 1973.

[14] "LNKNet, http://www.ll.mit.edu/IST/lnknet," Massachusetts Institute of Technology, Lincoln Labs.