Fusing deep speaker specific features and MFCC for robust speaker verification

Ryan Price[1,a)] Sangeeta Biswas[1] Koichi Shinoda[1]

Abstract: Acoustic representations typically used in speaker recognition are general and carry mixed information, including information that is irrelevant to the specific task of speaker recognition. Extracting specific information components from the speech signal for a desired task, such as extracting the speaker information component for speaker verification, is challenging. In this study, a nonlinear feature transformation discriminatively trained to extract speaker specific features from MFCCs is combined with a Gaussian mixture model support vector machine (GMM-SVM) system. Separation of the speaker information component and non-speaker related information in the speech signal is accomplished using a regularized siamese deep network (RSDN). RSDN learns a hidden representation that well characterizes speaker information by training a subset of the hidden units using pairs of speech segments. The hybrid RSDN GMM-SVM system achieves about 5% relative improvement over the baseline GMM-SVM system when applied to text-independent speaker verification using a subset of the NIST SRE 2006 1conv4w-1conv4w task. Speaker verification systems that fuse information typically provide better performance than those based on a single input modality. Score level fusion, in which scores from several classifiers are combined, is commonly employed as a fusion method for speaker verification. This study explores several fusion methods for RSDN and MFCC information, including score fusion, and the much less widely utilized fusion methods of GMM supervector fusion, and feature fusion. Score fusion and GMM supervector fusion offered further performance improvement, both achieving a 6.6% relative improvement over the baseline GMM-SVM system.

Keywords: speaker verification, neural networks, feature extraction, GMM-SVM, score fusion, feature fusion

1. Introduction

The task of speaker verification is concerned with verifying an individual's claimed identity based on a sample of their speech. Discrimination between speakers is made based on speaker-related differences in the speech signal. The speech signal conveys information about the speaker's identity resulting from a combination of anatomical differences and the learned speaking habits of different speakers, though this information is thought to be secondary to the linguistic information which is the primary information component in the speech signal. Thus, performance of speaker verification systems depends on extracting and modeling speaker specific characteristics of the speech signal which distinguish speakers from one another [1].

GMM-SVM with mel-frequency cepstral coefficients (MFCC) input features are one of the most prevalent approaches for speaker verification [2], [3], [4]. In the GMM-SVM approach, MAP adapted means of the mixture components are stacked to form supervectors. The supervectors are input to SVMs that model the boundary between a speaker and a set of imposters, rather than modeling their probability distributions. While this approach has provided good performance with demonstrated robustness and scalability, there are drawbacks. For example, the generative approach of GMM speaker modeling lacks the ability to extract speaker specific information by discriminative means. The MFCC input features typically used in these systems are a general spectral representation, widely used in various speech tasks, such as speech and speaker recognition. However, a task-specific representation designed with the objective of maximizing speaker verification performance may be more suitable for the problem of speaker verification.

An interesting area of recent work in speech information processing (e.g. [5], [6], [7]) uses regularized siamese deep networks (RSDN) to extract speaker specific information from a spectral representation, which can then be used in speaker modeling instead of the general spectral representations of speech that are typically employed. RSDN consist of two identical multilayer feed-forward networks, the middlemost layers of which are associated via a contrastive loss function for learning a speaker specific representation from spectral input features. As of yet, speaker specific features extracted using RSDN have not widely been combined with a robust approach to speaker modeling and decision-making for speaker verification, though some preliminary work has been done [8].

In this study we combine a GMM-SVM system with speaker specific input features extracted from a discriminatively trained RSDN, forming a hybrid RSDN GMM-SVM, and demonstrate the potential of this approach by applying it to a subset of the NIST SRE 2006 task [9]. We focus on exploring several information fusion methods including score fusion, GMM supervector fusion, and feature fusion. The main contribution of this paper is the study of multiple fusion methods for the new, not yet well

[1] Tokyo Institute of Technology
[a)] ryan@ks.cs.titech.ac.jp

studied, hybrid RSDN GMM-SVM using standard benchmark datasets for speaker verification. The hybrid RSDN GMM-SVM alone improves on an MFCC based GMM-SVM baseline and we demonstrate that score combination and supervector fusion both offer further performance improvements.

The outline of this paper is as follows. In Section 2 we review some related studies on neural network based feature extraction for speaker recognition and information fusion methods. Then we describe the RSDN architecture we use for extracting speaker specific features from MFCC inputs. Section 4 reviews key concepts in the GMM-SVM approach to speaker verification and describes the hybrid RSDN GMM-SVM system. Section 5 provides the details of our experimental setup and results are presented in Section 6. We finish the paper with some conclusions and possibilities for future work.

2. Related Studies

Considerable research has been done on extracting speaker discriminative features from cepstral features using multilayer perceptrons (MLP) (e.g. [10], [11], [12], [13]). A common approach was to train an MLP to discriminate between a set of selected speakers, with a window of several frames of MFCC features input to the neural network. After sufficient training of the MLP, weights are fixed and features are extracted from a hidden layer, usually a bottleneck layer, and are used as inputs to train another classifier for speaker identification or speaker verification. In [11], hidden layer activations from MLPs trained to distinguish between a subset of speakers selected through a speaker clustering process were used as input features for a SVM speaker recognition system. Stoll *et al.* [11] also examined the use of features generated by an MLP that was trained to distinguish between phones as input to a GMM speaker recognition system. [13], [14] focused on extracting features that were robust to mismatched training and testing conditions of speaker verification systems by training MLPs with data for the same speakers under different conditions. Morris *et al.* [12] dealt with the question of which subset of speakers would be most effective for MLP training when data is available for a large number of speakers.

There are significant difficulties in applying these techniques on a larger scale with the more challenging speaker verification tasks studied today, even though previous attempts at using neural network hidden representations as features for speaker recognition demonstrated some success. These approaches typically select a small number of speakers to form a small speaker basis in order to prevent the classification problem from becoming too difficult to train the MLP. Modern speaker recognition evaluations contain a diverse set of speakers, languages, channel effects and handset types that would be difficult to represent well with such a small speaker basis set. Since the RSDN based feature extraction approach allows for any number of speakers in the training data, selecting a compact but representative basis set is not necessary.

Examples of utilizing information fusion in speaker verification for improved performance, though not always emphasized, are numerous in the literature. [15] obtained excellent speaker verification performance through score level fusion of 11 subsystems. [16] also illustrates the importance of score calibration and score fusion for speaker verification. GMM supervector fusion is an interesting alternative to score fusion, though it is much less frequently applied. [17] fused GMM supervectors from MFCC and LPCC and found supervector fusion performed better than score fusion in a small speaker verification task. [18] provides a broader discussion of fusion methods for multimodal biometric systems in general.

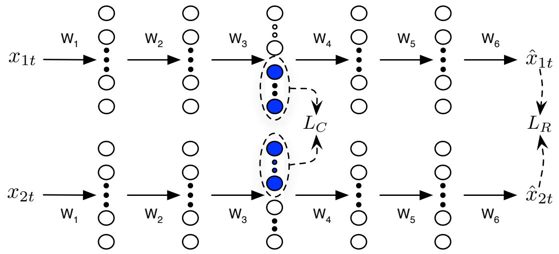

Fig. 1 Regularized siamese deep network shown with contrastive loss L_C and reconstruction loss L_R. Contrastive loss is applied to speaker related units which learn a speaker specific representation of the speech signal. Reconstruction loss and non-speaker related units aid in normalizing non-speaker related information in the speech signal.

Fusion has also been useful in neural network feature extraction based approaches. For example, [11] found that feature level fusion of MLP features and MFCCs, as well as score level combination of GMMs trained with those features, offered significant improvements over a basic, 256 mixture component, GMM baseline. However, score level fusion of MLP features and a state-of-the-art GMM system offered no performance improvement. [13] found consistent improvement in performance when taking a weighted combination of scores from MLP feature based GMMs and cepstrum based GMMs, even when the MLP features performed poorly alone.

3. Regularized siamese deep network

3.1 Overview

RSDN can be thought of as a type of regularized siamese network discriminatively trained using pairs of speech segments (**Fig. 1**). The code layer units (we refer to the middlemost hidden layer as the "code layer") are split into two parts - speaker related and non-speaker related. The speaker related code layer units learn a stable representation of the speech from a given speaker. The representation learned for each speaker should also be dissimilar to the representations of other speakers. In other words, the representation learned by these units is speaker specific. The non-speaker related units, along with input reconstruction constraints added to the network, aid in the normalization of non-speaker related information present in the speech signal. Following an unsupervised initialization step, RSDN discriminative training is accomplished using 2 types of loss functions described in Subsection 3.3. In addition to speaker verification tasks, RSDN extracted features have been applied to speaker comparison and speaker segmentation [5], [6], [19]. The unsupervised initialization step is described next.

3.2 Pretraining

The pretraining phase serves as a means of initializing the weights and biases of a deep autoencoder using unlabeled data, which can then be converted to a siamese network and discriminatively trained using labeled pairs of speech segments. One method for initializing a deep autoencoder is to stack denoising autoencoders [20]. Denoising autoencoders, like classical autoencoders, map an input to a hidden representation, which is then mapped back to a reconstruction of the input. However, denoising autoencoders reconstruct the input from a *corrupted* version of it and allow for an overcomplete hidden layer while still learning a useful intermediate representation. Mean squared error loss between the clean input and the reconstructed input is minimized with respect to the weights and biases.

During the RSDN pretraining phase, unlabeled speech data, in this case MFCC features, are used for training a stacked denoising autoencoder. MFCC features are continuous valued so Gaussian noise in the form of $\mathcal{N}(0, \sigma_i)$, where σ_i is the standard deviation of feature i estimated from the training data, is added to give a corrupted version of the inputs during layerwise pretraining. After the pretraining phase, the deep autoencoder is duplicated to form a siamese network with two identical subnets (see Fig. 1). The weights and biases of the the two halves of the siamese network are shared and any subsequent weight update is applied to both halves, keeping their values the same.

3.3 Discriminative training

During the discriminative training phase, pairs of short speech segments (X_1, X_2), T frames in length, coming from either the same speaker (genuine pairs) or from different speakers (imposter pairs), are presented to the network. The contrastive loss function (Eq. (2)) is applied to a subset of the code layer units in order to learn a speaker specific representation and is actually a combination of two cost functions - one minimizes the objective with respect to genuine pairs, and the other minimizes with respect to imposter pairs. A second loss function, reconstruction loss (Eq. (3)), provides a form of weight regularization during discriminative training by requiring the network still be good at reconstructing the input. Description of the loss functions is facilitated by first defining a compatibility measure between two speech segments, (X_1, X_2):

$$C(X_1, X_2) = C_m(X_1, X_2) + C_s(X_1, X_2), \tag{1a}$$

$$C_m(X_1, X_2) = |\boldsymbol{\mu}_{S1} - \boldsymbol{\mu}_{S2}|_2^2, \tag{1b}$$

$$C_s(X_1, X_2) = |\Sigma_{S1} - \Sigma_{S2}|_F^2, \tag{1c}$$

with $\boldsymbol{\mu}_{S1}$, $\boldsymbol{\mu}_{S2}$, Σ_{S1} and Σ_{S2} being the means and covariance matrices of the outputs of speaker specific code layer units corresponding to the segment pair (X_1, X_2), respectively, and $| \cdot |_F$ is the Frobenius norm. Intuitively, we can see that if speech segments are from the same speaker the value of Eq. (1a) should be small. In contrast, when the speech segments are from different speakers, the value of Eq. (1a) should be large. Minimizing the contrastive loss function based on the compatibility measure ensures that the RSDN representation of speech from the same speaker is similar while being dissimilar to the representation of speech from other speakers. The contrastive loss L_C is defined as:

$$L_C(X_1, X_2) = \mathcal{I}C + (1 - \mathcal{I})[e^{-\frac{C_m}{\lambda_\mu}} + e^{-\frac{C_s}{\lambda_{\mathrm{cov}}}}], \tag{2}$$

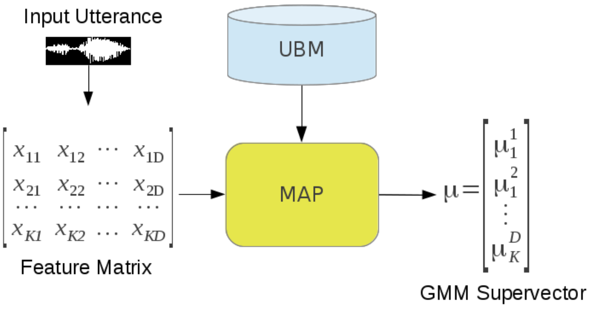
Input Utterance
UBM
x_{11} x_{12} ··· x_{1D}
x_{21} x_{22} ··· x_{2D}
x_{K1} x_{K2} ··· x_{KD}
Feature Matrix
MAP
$\mu = [\mu_1^1, \mu_1^2, \vdots, \mu_K^D]$
GMM Supervector

Fig. 2 A GMM supervector is generated for a speaker through MAP adaptation of the UBM given an utterance from the speaker.

with $\mathcal{I} = 1$ for genuine pairs and 0 for imposter pairs, and λ_μ and λ_{cov} are hyperparameters set to approximately balance the value of the contrastive loss function for genuine pairs and imposter pairs. The reconstruction loss L_R is defined as:

$$L_R(X_1, X_2) = \frac{1}{T} \sum_{t=1}^{T} \left[|x_{1t} - \hat{x}_{1t}|_2^2 + |x_{2t} - \hat{x}_{2t}|_2^2 \right], \tag{3}$$

The contrastive loss and reconstruction loss are combined in the overall loss function (Eq. (4)) which is minimized during discriminative training.

$$L(X_1, X_2) = \alpha L_R + (1 - \alpha) L_C, \tag{4}$$

where α determines the trade-off between L_R and L_C.

4. Hybrid RDSN GMM-SVM

4.1 GMM-SVM for Speaker Verification

Before describing the combination of the RSDN speaker specific feature extractor and GMM-SVM, we will briefly review the key concepts of GMM supervectors, SVM, and a linear kernel function for GMM supervectors. Additional details can be found in [21]. A GMM universal background model (UBM) is defined as

$$g(\boldsymbol{x}) = \sum_{i=1}^{M} \omega_i \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \tag{5}$$

where ω_i, $\boldsymbol{\mu}_i$, and Σ_i are the mixture weight, means and covariance of the ith mixture component, respectively. A speaker's utterance is used to obtain a speaker model through MAP adaptation of the UBM means $\boldsymbol{\mu}_i$. The means from this adapted speaker model are concatenated to form a GMM supervector, which is later used as a positive sample for training an SVM model for the given speaker. Negative training samples are generated using utterances from speakers who make up the UBM. The process of generating a supervector from a given input utterance and UBM is illustrated in **Fig. 2**.

SVM applies linear classification techniques after performing a nonlinear mapping from the input feature space to a very high dimensional feature space and is a very effective method for the binary classification problem in speaker verification. The discriminant function for an SVM is

$$f(\boldsymbol{y}) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{y}, z) + b, \tag{6}$$

where z are support vectors and parameters α_i and b are found during optimization, and the kernel function $k(\boldsymbol{y}, \boldsymbol{y'})$ is defined as

$$k(\boldsymbol{y}, \boldsymbol{y'}) = \phi(\boldsymbol{y})^T \phi(\boldsymbol{y'}), \qquad (7)$$

where $\phi(\cdot)$ is a mapping from the input space to the high dimensional SVM feature space. Whether $f(\boldsymbol{y})$ is above or below a threshold determines the classification decision.

Since the kernel has an effect on the SVM decision boundary, selection of the kernel is generally data dependent [22]. Supervectors, which map an utterance to a high dimensional vector, require an appropriate kernel for the associated classification problem. Given supervectors $\boldsymbol{s}^a$ and $\boldsymbol{s}^b$ resulting from GMM training and MAP adaptation of two utterances a and b, [21] derives a kernel function based on an approximation to the KL divergence between the two utterances,

$$k(\boldsymbol{s}^a, \boldsymbol{s}^b) = \sum_{i=1}^{M} (\sqrt{\omega_i}\Sigma_i^{-\frac{1}{2}}\boldsymbol{\mu}_i^a)^T (\sqrt{\omega_i}\Sigma_i^{-\frac{1}{2}}\boldsymbol{\mu}_i^b), \qquad (8)$$

where $\boldsymbol{\mu}_i^a$ and $\boldsymbol{\mu}_i^b$ are the means corresponding to mixture component i from supervectors $\boldsymbol{s}^a$ and $\boldsymbol{s}^b$, respectively. This can be recognized as a linear kernel of the familiar form given by

$$k(\boldsymbol{y}, \boldsymbol{y'}) = \boldsymbol{y}^T \boldsymbol{y'}, \qquad (9)$$

where y_i corresponds to supervector mean μ_i from mixture component m after normalizing by the corresponding mixture weight and variance parameters from the UBM,

$$y_i = \sqrt{\frac{\omega_m}{\sigma_i^2}}\mu_i. \qquad (10)$$

4.2 Combination of RSDN feature extractor and GMM-SVM

In [6], [7], [19], RSDN code layer outputs were used to derive single Gaussian speaker models from short speech segments for speaker verification tasks. Scores for binary classification of a given test trial were calculated using a simplified form of the divergence metric for two normal distributions from [23], or by symmetric Gaussian log likelihood measure [24]. Performance based on equal error rate (EER) and minimum detection cost (MDC) metrics was compared with features extracted from autoencoders and convolutional neural networks, as well as a GMM trained with 19 dimensional MFCC features and RSDN features and offered promising results in the speaker verification tasks studied. [8] takes a straightforward approach to combining speaker specific features extracted from the RSDN code layer with a GMM-SVM system for speaker verification. We briefly review the hybrid RSDN GMM-SVM approach below.

Since we are only interested in using the RSDN as a nonlinear feature extractor, reconstructions of the input are not needed after pretraining and discriminative training. Thus, only the encoder portion of one half of the siamese network is kept (**Fig. 3**) which allows for more efficient feature extraction since much of the computation associated with the full siamese network is eliminated.

The remaining nonlinear feature extractor (Fig. 3) is used to extract speaker specific features from MFCC inputs for each utterance in the UBM, enrollment and evaluation data. The feature extractor from Fig. 3 replaces the input utterance with speaker specific features in the supervector generation process illustrated in Fig. 2. GMM-SVM training follows the typical GMM-SVM training procedure [2], starting with using the extracted features to derive a UBM. RSDN code layer outputs are then extracted for all frames in the enrollment and evaluation data. Next, supervectors are created on a per utterance basis using MAP adaptation of the means with a relevance factor of 1. Supervectors extracted from the utterances used to train the UBM are used as imposter examples to train an SVM model with the linear kernel in Eq. (8) for each target speaker in the enrollment set. Finally, scores are calculated for each target and nontarget trial using the target speaker's SVM model and the supervector extracted from the test utterance using the discriminant function in Eq. (6).

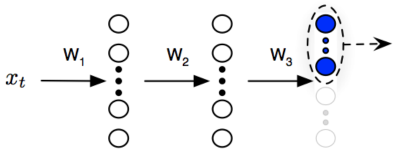

Fig. 3 Given a frame of MFCC input features x_t, outputs from the speaker specific code layer units are extracted for input into the GMM-SVM. For efficient nonlinear feature extraction, only the encoder portion of one half of the RSDN is retained after training.

5. Experiments

In this section we describe pre-processing of the raw inputs and the network training protocol. We then describe the copora used in our text-independent speaker verification experiments and introduce the fusion methods we explored for improving performance.

5.1 Data Processing

We use the following procedure for extracting MFCC features for use as input features. Silence was removed using an energy based VAD. The speech signal was pre-emphasized by applying the first order difference equation $s'_n = s_n - 0.95 s_{n-1}$. A 25 msec Hamming window and 10 ms frame rate are used to extract a 19 dimensional MFCC vector (zero order coefficient is excluded) from 24 filterbank channels. Cepstral mean normalization is applied to account for long-term spectral effects caused by channel differences.

5.2 Training Protocol

Details of RSDN training are as follows. All frames in the training data were used for pretraining. For discriminative training, we created approximately 6000 pairs of segments (X_1, X_2) that are 500 frames in length by splitting up the training data into a set, S, of non-overlapping segments and randomly selecting another a segment, X_2, for each segment $X_1 \in S$, with $X_2 \neq X_1$. We ensured the data set is "balanced" by generating approximately the same number of genuine pairs as imposter pairs.

Mini-batch sizes were 100 frames and 500 frames for pretraining and discriminative training, respectively. Note that number of frames in a minibatch used for discriminative training must be the same as the number of frames in the speech segments since Eq.

(2) is defined using 1st and 2nd order statistics of code layer outputs generated from a speech segment that is T frames in length. We used a network with 5 hidden layers having sizes of 100, 100, 200, 100, and 100 hidden units, respectively. We used 100 speaker specific units in the code layer, which is half the number of hidden units in that layer, as this was found to be advantageous in [5], [19]. The network was pretrained for 40, 20, 20 epochs at a learning rate of 0.01. The learning rate for discriminative training was 0.001. The trade-off parameter α was set to 0.2. In order to prevent overfitting, early stopping and learning rate annealing were used while checking EER on the development data, as well as monitoring the contrastive and reconstruction losses.

5.3 Text-independent speaker verification

We evaluate the performance of the hybrid RSDN GMM-SVM system, as well as several strategies for fusing with MFCC based scores and features, on subsets of NIST SRE 2004 and 2006 [9]. The 242 male speakers from NIST SRE 2004 1-side were selected for pretraining and discriminative training of the RSDN. Utterances from 50 randomly selected male speakers who did not appear in the training data were taken from NIST SRE 2004 8-side training files and used for RSDN development. The same utterances from NIST SRE 2004 1-side that we used for RSDN training were used to train the UBM. For evaluation, we randomly selected 100 male speakers from the NIST SRE 2006 1conv4w-1conv4w task and used all trials associated with those speakers. In total, 453 genuine trials and 6057 imposter trials were used for evaluation.

The training and test utterances used in this experiment are 2 minutes on average after silence removal and are considerably longer than those used in previous studies of RSDN extracted features for speaker verification, such as [6], [7], which mainly focused on short test utterances (5s or less) without channel and environmental mismatch. Most of the segments in the training and test data are in English (but include non-native speakers) and are recorded over a telephone line, but different languages, types of telephone handsets and transmission channels are included.

5.4 Fusion

Two broad categories of fusion in biometric systems exist: pre-classification fusion combines information prior to the application of a classifier and post-classification fusion combines information after the decisions of the classifiers have been obtained [18]. Some common approaches to fusion are feature level fusion (a form of pre-classification fusion) and score level fusion (a form of post-classification fusion). In feature level fusion different feature vectors that are obtained by applying multiple feature extraction algorithms on the same raw data are combined. We try two different types of feature level fusion - fusing RSDN extracted features and MFCCs, and fusing supervectors from MAP adapted GMMs trained with RSDN extracted features and MFCCs. Score level fusion refers to combining the scores obtained from several classifiers. We take the score combination approach in which the individual scores from SVMs trained on supervectors resulting from RSDN extracted features and SVMs trained on supervectors resulting from MFCCs are combined to generate a single scalar score that is used to make the decision.

Table 1 *EER(%) for MFCC based GMM-SVM and Hybrid RSDN GMM-SVM with varying number of mixture components.*

	Mix Comp			
System	32	64	128	256
GMM-SVM (MFCC)	14.77	13.72	**13.24**	13.65
Hybrid RSDN GMM-SVM	12.59	**12.58**	13.90	15.00

6. Results and Discussion

6.1 Comparison with MFCC based GMM-SVM

The baseline results for the hybrid RSDN GMM-SVM system and MFCC based GMM-SVM system prior to fusion are presented in this subsection and results for several fusion methods are presented in subsequent subsections. EER is used as an evaluation metric in all experiments. The number of GMM mixture components was varied from 32 to 256 and the EER for both systems is shown in table 1. The best performing hybrid system achieved 12.58% EER with 64 mixture components which is about a 5% relative reduction in EER compared to the best performing MFCC based GMM-SVM which achieved 13.24% with 128 mixture components.

It should be noted that there are several differences in the GMM-SVM system evaluated here compared to those evaluated by others who have also used NIST'04 for background training data and NIST'06 for evaluation, such as [4], were all of NIST'04 was used (we use a subset) and concepts like score normalization, nuisance attribute projection (NAP), and factor analysis were also applied. Nonetheless, we believe this result demonstrates that combining speaker specific features extracted from RSDN with a GMM-SVM system can be an effective approach. The dimensionality of the RSDN extracted feature vectors used in the hybrid RSDN GMM-SVM is more than 5 times greater than that of the MFCC feature vectors used in the MFCC based GMM-SVM. It seems not surprising that the best performing hybrid system has fewer mixture components relative to the MFCC based GMM-SVM, given that the number of training vectors remains the same for both systems. We note that the performance of the hybrid system declines abruptly when the number of mixture components is increased beyond 64, possibly due to overfitting, while the performance of the MFCC based GMM-SVM system remains relatively unchanging for 64, 128, and 256 mixture components.

6.2 Score fusion

It is relatively easy to access and combine scores generated by different classifiers and thus score fusion is one of the most common approaches in biometric systems [18], including speaker verification. Speaker recognition systems utilizing neural network feature extraction methods (Section 2) have also shown improvement when scores are fused with MFCC based systems, even when the extracted features performed poorly alone. We performed score fusion by combining the scores from the 64 mixture component hybrid RSDN GMM-SVM and the 128 mixture component MFCC based GMM-SVM, giving scores from the 2 systems equal weighting when calculating the combined score. The Detection Error Tradeoff (DET) curves for the fused scores, hybrid RSDN GMM-SVM, and MFCC based GMM-SVM are

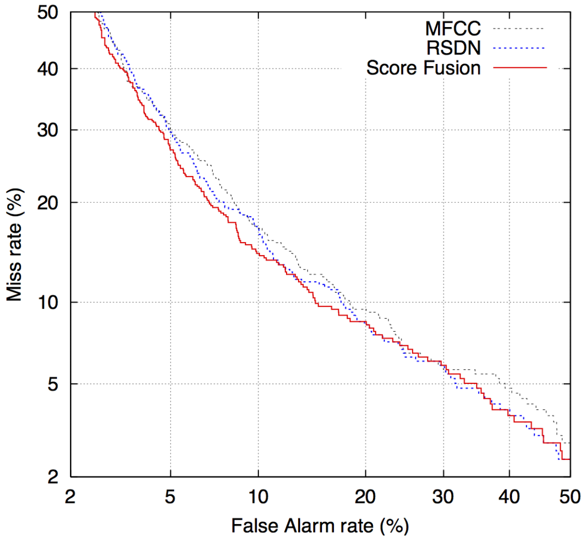

Fig. 4 DET curves for MFCC based GMM-SVM (MFCC), hybrid RSDN GMM-SVM (RSDN), and score fusion.

plotted in **Fig. 4**. DET curves for the 3 systems follow similar trends but the hybrid RSDN GMM-SVM and score fusion consistently outperform the MFCC based GMM-SVM. Score fusion of the hybrid RSDN GMM-SVM and MFCC based GMM-SVM provides some further reduction in EER, achieving an EER of 12.36%, which is a 6.6% relative improvement over the MFCC based GMM-SVM.

6.3 Supervector Fusion

Supervector fusion offers an interesting form of feature level fusion for GMM-SVM systems. We performed supervector fusion by concatenating the supervectors generated from MAP adapted GMMs trained with RSDN extracted features and those trained with MFCCs. Specifically, we fused supervectors from the 64 mixture component RSDN based GMM and the 128 mixture component MFCC based GMM, resulting in 8832 dimensional supervectors which were then used to train SVMs. The DET curves for the fused supervectors, hybrid RSDN GMM-SVM, and MFCC based GMM-SVM are plotted in **Fig. 5**. Like score fusion, supervector fusion performs favorably compared to the hybrid RSDN GMM-SVM and the MFCC based GMM-SVM. Supervector fusion achieves an EER of 12.36%, which is a 6.6% relative improvement over the MFCC based GMM-SVM. It is often useful (if not critical) to do input normalization for SVMs [22]. We considered that this might be the case for the fused supervectors and tried normalizing each supervector s by dividing s by its norm so that $||s|| = 1$ after normalization. However, we found that this degraded performance considerably and concluded that the normalization provided by Eq. (10) is already sufficient.

6.4 Fusion of RSDN extracted features and MFCC

While fusion at the score level has been extensively used in speaker verification literature, fusion at the feature level is relatively infrequently used. In addition to supervector fusion, we also performed another type of fusion at the feature level by directly concatenating the RSDN extracted features and MFCC features. This resulted in 119 dimensional feature vectors. A 64 mixture component GMM was then used to model the fused features and the MAP adapted supervectors were used to train SVMs following the usual GMM-SVM procedure. The DET curve for these fused features is shown in **Fig. 6**, along with curves for score fusion and supervector fusion for comparison. This feature fusion gave 13.69% EER, which is significantly worse than the other fusion methods.

Results of the different fusion methods are summarized in **Table 2**. Score fusion and supervector fusion performed similarly and obtained the lowest EER. Somewhat surprisingly to us, fusion of RSDN extracted features and MFCC did not perform well, even though biometric systems that integrate information at an early stage of processing are believed to be more effective than

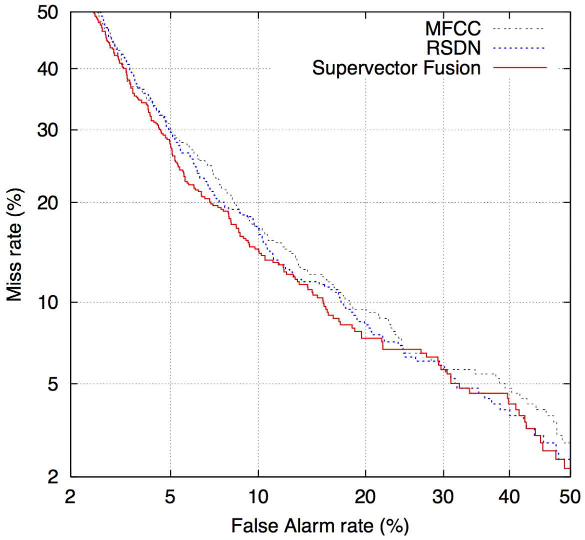

Fig. 5 DET curves for MFCC based GMM-SVM (MFCC), hybrid RSDN GMM-SVM (RSDN), and supervector fusion.

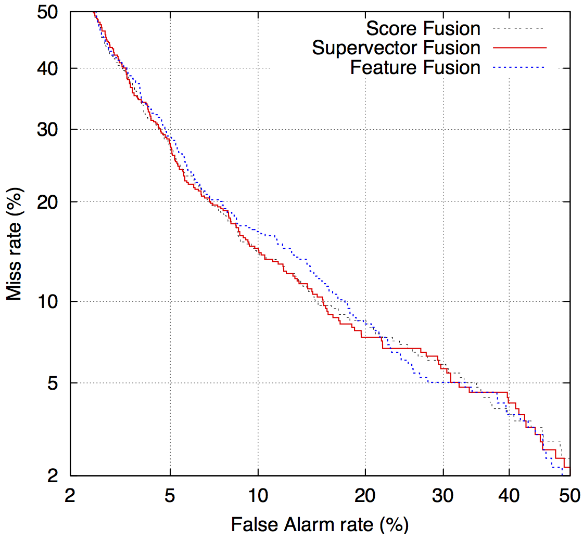

Fig. 6 DET curves for score fusion, supervector fusion and feature fusion.

Table 2 *EER(%) for various fusion methods.*

Fusion Method	EER
Score Fusion	12.36
Supervector Fusion	12.36
RSDN extracted features and MFCC fusion	13.69

those that integrate it at a later stage [18]. Potential reasons for the poor performance include the possibility that these features are too highly correlated and that 64 mixture components may be too many for the increased dimensionality of the fused feature vectors.

7. Conclusions and future work

We have explored several information fusion methods for a text-independent speaker verification system based on a hybrid RSDN GMM-SVM system. We first compared the baseline performance of the hybrid RSDN GMM-SVM system to a GMM-SVM system based on MFCC input features and found the hybrid RSDN GMM-SVM system yielded a modest performance gain of 5% relative reduction in EER for the subset of the NIST SRE 2006 1conv4w-1conv4w task studied. We then explored several methods for fusing information including score fusion and two types of feature fusion. Score fusion and supervector fusion yielded further improvement over the MFCC based GMM-SVM baseline system. Score fusion and supervector fusion gave comparable results, both achieving a 6.6% relative reduction in EER. In contrast, direct fusion of RSDN extracted features and MFCC features was not found to be helpful.

The results found here for the hybrid RSDN GMM-SVM approach along with several fusion methods are still preliminary and offer a lot of opportunity for future work. Several methods exist for score normalization of speaker verification systems, such as Z-norm and T-norm [25] and could potentially be applied prior to score fusion for improved performance. Application of channel compensation techniques, such as NAP [26] may also improve performance and could potentially be applied prior to fusing supervectors. More work needs to be done to find the optimal raw input features for RSDN feature extraction and determine if any benefit can be derived from including delta and delta-delta features as input features, as well as feature post-processing methods for increased robustness, such as feature warping [27]. The hybrid RSDN GMM-SVM system has a considerable number of parameters and the combination of two loss functions makes for a rather difficult training procedure. These factors may be critical to achieving good performance and could be obstacle to widespread adoption.

References

[1] Reynolds, D. A. and Rose, R. C.: Robust text-independent speaker identification using Gaussian mixture speaker models, *Speech and Audio Processing, IEEE Transactions on*, Vol. 3, No. 1, pp. 72–83 (1995).

[2] Campbell, W. M., Sturim, D. E. and Reynolds, D. A.: Support vector machines using GMM supervectors for speaker verification, *Signal Processing Letters, IEEE*, Vol. 13, No. 5, pp. 308–311 (2006).

[3] Reynolds, D. A. and Campbell, W.: Text-Independent Speaker Recognition, *Springer Handbook of Speech Processing and Communication* (Benesty, J., Sondhi, M. M. and Huang, Y. A., eds.), Springer-Verlag GMBH, Heidelberg, Germany (2007).

[4] Fauve, B. G., Matrouf, D., Scheffer, N., Bonastre, J.-F. and Mason, J. S.: State-of-the-art performance in text-independent speaker verification through open-source software, *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 15, No. 7, pp. 1960–1968 (2007).

[5] Chen, K. and Salman, A.: Extracting Speaker-Specific Information with a Regularized Siamese Deep Network, *Advances in Neural Information Processing Systems (NIPS'11)* (2011).

[6] Chen, K. and Salman, A.: Learning Speaker-Specific Characteristics with a Deep Neural Architecture, *Neural Networks, IEEE Transactions on*, Vol. 22, No. 11, pp. 1744–1756 (2011).

[7] Salman, A. and Chen, K.: Exploring speaker-specific characteristics with deep learning, *Neural Networks (IJCNN), The 2011 International Joint Conference on*, IEEE, pp. 103–110 (2011).

[8] Price, R., Biswas, S. and Shinoda, K.: Combining Deep Speaker Specific Representations with GMM-SVM for Speaker Verification, *Proc. Interspeech*, to appear (2013).

[9] National Institute of Standards and Technology:
Speaker Recognition Evaluation, NIST (online), available from ⟨http://www.itl.nist.gov/iad/mig/tests/spk/⟩ (accessed June 21, 2013).

[10] Wu, D., Morris, A. and Koreman, J.: MLP internal representation as discriminative features for improved speaker recognition, *Nonlinear Analyses and Algorithms for Speech Processing*, pp. 72–80 (2005).

[11] Stoll, L., Frankel, J. and Mirghafori, N.: Speaker recognition via nonlinear discriminant features, *Proceedings of NOLISP07* (2007).

[12] Morris, A. C., Wu, D. and Koreman, J.: MLP trained to separate problem speakers provides improved features for speaker identification, *Security Technology, 2005. CCST'05. 39th Annual 2005 International Carnahan Conference on*, IEEE, pp. 325–328 (2005).

[13] Konig, Y., Heck, L., Weintraub, M., Sonmez, K. et al.: Nonlinear discriminant feature extraction for robust text-independent speaker recognition, *Proc. RLA2C, ESCA workshop on Speaker Recognition and its Commercial and Forensic Applications*, pp. 72–75 (1998).

[14] Heck, L. P., Konig, Y., Sönmez, M. K. and Weintraub, M.: Robustness to telephone handset distortion in speaker recognition by discriminative feature design, *Speech Communication*, Vol. 31, No. 2, pp. 181–192 (2000).

[15] Brummer, N., Burget, L., Cernocky, J., Glembek, O., Grezl, F., Karafiat, M., van Leeuwen, D., Matejka, P., Schwarz, P. and Strasheim, A.: Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006, *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 15, No. 7, pp. 2072–2084 (2007).

[16] Katz, M., Schafföner, M., Krüger, S. E. and Wendemuth, A.: Score calibrating for speaker recognition based on support vector machines and gaussian mixture models, *Proceedings of the Ninth IASTED International Conference on Signal and Image Processing*, ACTA Press, pp. 146–151 (2007).

[17] Liu, M. and Huang, Z.: Multi-Feature Fusion Using Multi-GMM Supervector for SVM Speaker Verification, *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, pp. 1–4 (2009).

[18] Jain, A., Nandakumar, K. and Ross, A.: Score normalization in multimodal biometric systems, *Pattern Recognition*, Vol. 38, No. 12, pp. 2270 – 2285 (2005).

[19] Salman, A.: Learning speaker-specific characteristics with deep neural architecture, PhD Thesis, University of Manchester (2012).

[20] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. and Manzagol, P.-A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *The Journal of Machine Learning Research*, Vol. 11, pp. 3371–3408 (2010).

[21] Campbell, W., Sturim, D., Reynolds, D. and Solomonoff, A.: SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation, *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, Vol. 1, pp. 97–100 (2006).

[22] Ben-Hur, A. and Weston, J.: A users guide to support vector machines, *Data mining techniques for the life sciences*, Springer, pp. 223–239 (2010).

[23] Campbell Jr, J. P.: Speaker recognition: A tutorial, *Proceedings of the IEEE*, Vol. 85, No. 9, pp. 1437–1462 (1997).

[24] Besacier, L., Bonastre, J. and Fredouille, C.: Localization and selection of speaker-specific information with statistical modeling, *Speech Communication*, Vol. 31, No. 2, pp. 89–106 (2000).

[25] Auckenthaler, R., Carey, M. and Lloyd-Thomas, H.: Score normalization for text-independent speaker verification systems, *Digital Signal Processing*, Vol. 10, No. 1, pp. 42–54 (2000).

[26] Solomonoff, A., Campbell, W. and Boardman, I.: Advances In Channel Compensation For SVM Speaker Recognition, *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, Vol. 1, pp. 629–632 (2005).

[27] Pelecanos, J. and Sridharan, S.: Feature Warping for Robust Speaker Verification, *2001: A Speaker Odyssey - The Speaker Recognition Workshop*, Crete, Greece, International Speech Communication Association (ISCA), pp. 213–218 (2001).