I-vector with Sparse Representation Classification for Speaker Verification

Jia Min Karen Kua*, Julien Epps, Eliathamby Ambikairajah

School of Electrical Engineering and Telecommunications,
The University of New South Wales, UNSW Sydney, NSW 2052, Australia
j.kua@unswalumni.com, j.epps@unsw.edu.au, ambi@ee.unsw.edu.au,

Abstract − Sparse representation-based methods have very lately shown promise for speaker recognition systems. This paper investigates and develops an *i*-vectorbased sparse representation classification (SRC) as an alternative classifier to Support Vector Machine (SVM) and Cosine Distance Scoring (CDS) classifier, producing an approach we term *i*-vector−Sparse Representation Classification (*i*-SRC). Unlike SVM which fixes the support vector for each target example, SRC allows the supports, which we term sparse coefficient vectors, to be adapted to the test signal being characterized. Furthermore, similar to CDS, SRC does not require a training phase. We also analyze different types of sparseness methods and dictionary composition to determine the best configuration for speaker recognition. We observe that including an identity matrix in the dictionary helps to remove sensitivity to outliers and that sparseness methods based on ℓ_1 and ℓ_2 norm, offer the best performance. A combination of both techniques achieves a 18% relative reduction in EER over a SRC system based on ℓ_1 norm and without identity matrix. Experimental results on NIST 2010 SRE show that the *i*-SRC consistently outperform *i*-SVM and *i*-CDS in EER in the range of 0.14–0.81% and the fusion of *i*-CDS and *i*-SRC achieves a relative EER reduction of 8–19% over *i*-SRC alone.

Index Terms – Speaker recognition, sparse representation classification, ℓ_1-minimization, *i*-vectors, support vector machine, cosine distance scoring

1. Introduction

Automatic speaker verification is the task of authenticating a speaker's claimed identity. There are two fundamental research issues in automatic speaker verification, which are the exploration of discriminative information in speech in the form of features (e.g. spectral, prosodic, phonetic and dialogic) and how to effectively organize and exploit the speaker cues in the classifier design for the best performance.

Addressing the latter issue, some of the conventional methods include support vector machines (SVM) [1, 2] and Gaussian mixture model – universal background models (GMM-UBM) [3, 4]. When using GMM-UBM, each speaker is modelled as a probabilistic source. Each speaker is represented by the means (μ), covariance (typically diagonal) (σ) and weights (ω) of a mixture of n multivariate Gaussian densities defined in some continuous feature space of dimension f. These Gaussian mixture models are adapted from a suitable UBM using maximum a posterior (MAP) adaptation [4]. Matching is then performed by evaluating the likelihood of the test utterance with respect to the model.

SVMs have proven their effectiveness for speaker recognition tasks, reliably classifying input speech that has been mapped into a high-dimensional space, using a hyperplane to separate two classes [1, 2]. A critical aspect of using SVMs successfully is the design of the kernel, which is an inner product in the SVM feature space that induces distance metrics. Generalised linear discriminant sequence (GLDS) kernels and GMM supervectors are two such kernels [1, 5, 6] and the latter is employed in this paper. GMM supervectors are formed by concatenating the MAP-adapted mean vector elements ($\mu_{i,j}$) normalized using the weights (w_i) and the diagonal covariance elements ($\Sigma_{i,j}$) as shown in (1) where i is the index of the mixture, j is the index of the dimension of the feature vector, n is the total number of mixtures and f is the number of dimensions of the feature vector. Since SVMs are not invariant to linear transformations in feature space, variance normalization is performed so that some supervector dimensions do not dominate the inner product computations.

$$\mathbf{M} = \left[\frac{\sqrt{w_1}\mu_{1,1}}{\sqrt{\Sigma_{1,1}}} \cdots \frac{\sqrt{w_1}\mu_{1,f}}{\sqrt{\Sigma_{1,f}}} \cdots \frac{\sqrt{w_i}\mu_{i,1}}{\sqrt{\Sigma_{i,1}}} \cdots \frac{\sqrt{w_i}\mu_{i,f}}{\sqrt{\Sigma_{i,f}}} \cdots \frac{\sqrt{w_n}\mu_{n,1}}{\sqrt{\Sigma_{n,1}}} \cdots \frac{\sqrt{w_n}\mu_{n,f}}{\sqrt{\Sigma_{n,f}}}\right]^T \quad (1)$$

Although SVMs are capable of pattern classification in a high dimensional space using kernels, their performance is determined by three main factors: kernel selection, the SVM cost parameter and kernel parameters [7-9]. Many researchers have committed considerable time to finding the optimum kernel functions for speaker recognition [10-12] due to the diverse sets of kernel functions available. Once a suitable kernel function has been selected, attention turns to the cost parameter and kernel parameter settings [13]. Moreover, besides the factors as discussed above, the composition of speakers in the SVM background dataset has recently shown to have a significant impact on the speaker verification performance [14-17]. This is because the hyperplane that is trained using the target and background speakers' data tends to be biased towards the background dataset in a speaker verification task since the number of utterance from the target speaker (normally only one utterance) is usually much less than the background speaker (thousands of utterances). Therefore effective selection of the background dataset is required to improve the performance of an SVM-based speaker verification system. In [15], the support vector frequency was used to rank and select negative examples by evaluating the examples using the target SVM model, and then selecting the closest negative examples to the enrolment speaker as the background dataset. Their proposed technique results in an improvement of 10% in EER on NIST 2006 SRE over a heuristically chosen background speaker set.

Currently, one of the main challenge in speaker modelling is channel variability between the testing and training data [18, 19]. In [20], Kenny et al. introduced Joint Factor Analysis (JFA) as a technique for modelling inter-speaker variability and to compensate for channel/session variability in the context of GMMs, and more recently the *i*-vectors [21, 22], which have collectively amounted to a new de facto standard in state-of-the-art speaker recognition systems. In the *i*-vector framework, the speaker and channel-dependent supervector **M** is represented as

$$\mathbf{M} = \mathbf{m} + \mathbf{Tq} \tag{2}$$

where **T** is the total variability matrix (containing the speaker and channel variability simultaneously) and **q** is the identity vector (*i*-vector) of dimension typically around 400. Channel compensation is then applied based on within-class covariance normalization (WCCN) [26] and/or linear discriminant analysis

(LDA) [21]. WCCN was introduced in [27] for minimizing the expected error rate of false acceptances and false rejections during the SVM training step. The WCC matrix is computed as

$$\boldsymbol{W} = \frac{\mathbf{1}}{\boldsymbol{C}}\sum_{\boldsymbol{c=1}}^{\boldsymbol{C}}\frac{\mathbf{1}}{\boldsymbol{n_c}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n_c}}(\mathbf{q}_i^c - \overline{\mathbf{q}_c})(\mathbf{q}_i^c - \overline{\mathbf{q}_c})^t, \tag{3}$$

where $\overline{\mathbf{q}_c} = \frac{1}{n_c}\sum_{i=1}^{n_c}\mathbf{q}_i^c$ is the mean of the *i*-vectors of each speaker, *C* is the number of speakers and n_c is the number of utterances for each speaker *c*. Then a feature-mapping function φ_{wccn} is defined as

$$\varphi_{wccn}(\mathbf{q}) = \mathbf{B}^t\mathbf{q}, \tag{4}$$

where B is obtained through Cholesky decomposition of matrix $\mathbf{W}^{-1} = \mathbf{B}\mathbf{B}^t$. In the case of LDA, similarly to WCCN, the speaker factors are then submitted to the projection matrix A obtained from LDA[21] as follows

$$\varphi_{LDA}(\mathbf{q}) = \mathbf{A}^t\mathbf{q}. \tag{5}$$

In the total variability space, Dehak et al. [21] introduce a new classification method based on cosine distance, termed the Cosine Distance Scoring (CDS) classifier as an alternative to SVM as shown in equation (6) where $\mathbf{q}_{test}$ and $\mathbf{q}_{target}$ are the test and target speaker's *i*-vectors respectively. The CDS classifier allows a much simplified speaker recognition system since the test and target *i*-vectors are scored directly, as opposed to SVM which requires the training of a target model before scoring.

$$\text{score}(\mathbf{q}_{test}, \mathbf{q}_{target}) = \frac{\langle \mathbf{q}_{test}, \mathbf{q}_{target}\rangle}{\|\mathbf{q}_{test}\|\|\mathbf{q}_{target}\|} \tag{6}$$

Widespread interest in sparse signal representations is a recent development in digital signal processing [28-31]. The sparse representation paradigm, when it was originally developed, was not intended for classification purposes but instead for an efficient representation and compression of signals at a greatly reduced rate than the standard Shannon-Nyquist rate with respect to an overcomplete dictionary of base elements [32, 33]. Nevertheless, the sparsest representation is naturally discriminative because among the set of base vectors, the subset which most compactly represent the input signal will be chosen [31]. In compressive sensing, the familiar least squares optimization is inadequate for signal

decomposition, and other types of convex optimization are used [28]. This is because the least square optimization usually results in solutions which are typically non-sparse (involving all the dictionary vectors) [34] and the largest coefficients are often not associated with the class of the test sample when used for classification as illustrated in [31].

In recent years, sparse representation based classifiers have begun to emerge for various applications, and experimental results indicate that they can achieve comparable or better performance to that of other classifiers [31, 35-37]. In the case of face recognition, Wright et al. cast the problem in terms of finding a sparse representation of the test image features with respect to the training set, whereby the sparse representation are computed by ℓ_1-minimization [31]. They exploit the following simple observation: if sufficient training data are available for each class, a test sample is represented only as a linear combination of the training sample from the same class, wherein the representation is sparse by excluding samples from other classes. They have shown an absolute accuracy gain of 0.4% and 7% over linear SVM and nearest neighbour methods respectively on the Extended Yale B database [38]. Further, in [35], Naseem et al. showed classification based on sparse representation to be a promising method for speaker identification. Although the initial investigations were encouraging, the relatively small TIMIT database characterizes an ideal speech acquisition environment and does not include e.g. reverberant noise and session variability. Recently we exploited the discriminative nature of sparse representation classification using supervectors and NAP [35] for speaker verification as an alternative and/or complementary classifier to SVM on the NIST 2006 SRE database [39].

Recently, a discriminative SRC, which focuses on achieving high discrimination between classes as opposed to the standard sparse representation that focuses on achieving small reconstruction error, was proposed specifically for classification tasks [30]. The results in [30] demonstrated that discriminative SRC is more robust to noise and occlusion than the standard SRC for signal classification. The discriminative approach works by incorporating an additional Fisher's discrimination power to the sparsity property in the standard sparse representation. Our initial investigation was unsuccessful since the discriminative SRC requires the computation of the Fisher F-ratio (ratio of between-class and within-class

variances) [40] with multiple samples per class. However for the task of speaker verification (which is a two class problem) with only one sample for the target class, the within-class scatter for the target class always goes to zero.

This paper is motivated by our previous work on sparse representation using supervectors [39] and recent work by Li et al. [41] using *i*-vectors as features for SRC. Li et al [41] focus on enhancing the robustness and performance of speaker verification through the concatenation of a redundant identity matrix at the end of the original over-complete dictionary, new scoring measures termed as background normalised (Bnorm) ℓ_2-residual and a simplified TNorm procedure for SRC system by replacing the dictionary with TNorm i-vectors. However, two factors that can have a significant impact on classification performance, the choice of sparsity regularization constraints and background set used in the SRC dictionary are not explored. As discussed earlier, ever since SVMs were introduced to the field of speaker recognition by Campbell et al. [1], various extensive investigations have been conducted in each individual component of SVM (e.g type of kernel, SVM cost parameter, kernel parameters and background dataset) with the hope of improving the system performance and/or increasing the computational efficiency of SVM training. Similarly in this work and building on the work of Li et al. [41], we extend our analysis to different types of sparseness constraints, dictionary composition and ways to improve the robustness of SRC against corruption as recommended in [31, 41] to determine the best configuration for speaker recognition using SRC. Furthermore, a comparison in terms of classification performance between CDS and SRC will be conducted since both classifiers have the common property of not requiring a training phase.

2. Sparse Representation Classification

2.1. Sparse Representation

The sparse representation of a signal with respect to an overcomplete dictionary is formulated as follows. Given a $K \times N$ matrix $\mathbf{D}$, where each column represents an individual vector from the overcomplete

dictionary, with $N > K$ and usually $N >> K$, then for the sparse representation of a signal $\mathbf{S} \in \mathbb{R}^K$, the problem is to find an $N \times 1$ coefficient vector $\boldsymbol{\gamma}$, such that $\mathbf{S} = \mathbf{D}\boldsymbol{\gamma}$ and $\|\boldsymbol{\gamma}\|_0$ is minimized as follows

$$\boldsymbol{\gamma} = \arg\min_{\boldsymbol{\gamma}'} \|\boldsymbol{\gamma}'\|_0 \qquad s.t. \qquad \mathbf{S} = \mathbf{D}\boldsymbol{\gamma} \tag{7}$$

where $\|.\|_0$ denotes the ℓ_0-norm, which counts the number of nonzero entries in a vector. However finding the solution to a underdetermined system of linear equations is NP-hard [42]. Recent developments in sparse representation and compressive sensing [43, 44] indicate that if the solution $\boldsymbol{\gamma}$ sought is sparse enough, the ℓ_0-norm in (7) can be replaced with an ℓ_1-norm as shown in (8), which can be efficiently solved by linear programming.

$$\boldsymbol{\gamma} = \arg\min_{\boldsymbol{\gamma}'} \|\boldsymbol{\gamma}'\|_1 \qquad s.t. \qquad \mathbf{S} = \mathbf{D}\boldsymbol{\gamma} \tag{8}$$

2.2. Classification based on Sparse Representation

In classification problems, the main objective is to determine correctly the class of a test sample (**S**) given a set of labelled training samples from L distinct classes. First, the l_i training samples from the ith class are arranged as the columns of a matrix $\mathbf{D}_i = [\mathbf{v}_{i,1}, \mathbf{v}_{i,2}, \dots, \mathbf{v}_{i,l_i}]$. If **S** is from class i, then **S** will approximately lie in the linear span of the training samples in $\mathbf{D}_i$ [31]

$$\mathbf{S} \approx \alpha_{i,1}\mathbf{v}_{i,1} + \alpha_{i,2}\mathbf{v}_{i,2} + \cdots + \alpha_{i,l_i}\mathbf{v}_{i,l_i} \tag{9}$$

for some scalars, $\alpha_{i,j} \in \mathbb{R}, j = 1,2, \dots, l_i$.

Since the correct class identity of the test sample is unknown during classification, a new matrix **D** is defined as the concatenation of all the training samples of all L classes:

$$\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \dots, \boldsymbol{D}_L] = [\mathbf{v}_{1,1}, \mathbf{v}_{1,2}, \dots, \mathbf{v}_{L,l_L}] \tag{10}$$

Then, **S** can be rewritten as a linear combination of all training samples as

$$\mathbf{S} = \mathbf{D}\boldsymbol{\gamma} \tag{11}$$

where the coefficient vector, termed the sparse coefficients [45], $\boldsymbol{\gamma} = [0, \dots, 0, \alpha_{i,1}, \alpha_{i,2}, \dots \alpha_{i,l_i}, 0, \dots, 0]^T$ has entries that are mostly zero except those associated with the ith class after solving the linear system of

equations $\mathbf{S} = \mathbf{D}\boldsymbol{\gamma}$ using (8). In this case, the indices of the sparse coefficients encode the identity of the test sample **S**, and these form the non-zero entries of what we term the 'sparse coefficient vector', $\boldsymbol{\psi}$.

In order to demonstrate sparse representation classification using ℓ_1-norm minimization (equation (8)), an example matrix **D** was created using a small number of synthetic 3-dimensional data[1] (K = 3), where the columns of **D** represent 6 different classes with 1 samples for each class in our previous work (L = 6 , N = 6) [39]. A test vector **S** was chosen near to class 4 (C4). Solving equation (8)[2] produces the vector $\boldsymbol{\gamma} \approx [0, 0, -0.2499, 0.8408, 0, 0.2136]^T$, where the largest value (0.8408) corresponds to the correct class (C4), but $\boldsymbol{\psi}$ also has entries from training samples of classes 3 and 6. Ideally, the entries in $\boldsymbol{\psi}$ would only be associated with samples from a single class *i* where we can easily assign the test sample **S** to class *i*. However, noise may lead to small nonzero entries associated with other classes (as shown in the example discussed above) [31].

For more realistic classification problems, or problems with more than one training samples per class, **S** can be classified based on how well the coefficients associated with all training samples of each class reproduce **S,** instead of simply assigning **S** to the object class with the single largest entry in $\boldsymbol{\gamma}$ [31]. For each class *i*, let $\delta_i: \mathbb{R}^N \longrightarrow \mathbb{R}^N$ be the characteristic function that selects the coefficients associated with the *i*th class as shown in (12).

$$\delta_i(\boldsymbol{\gamma}) = \begin{bmatrix} \sigma_{1,1} \\ \sigma_{1,2} \\ \vdots \\ \sigma_{1,l_1} \\ \vdots \\ \sigma_{N,1} \\ \sigma_{N,2} \\ \vdots \\ \sigma_{N,l_N} \end{bmatrix} \; where \; \sigma_{j,k} = \begin{cases} 0\,, j \neq i \\ \alpha_{i,k}\,, j = i \end{cases} \qquad (12)$$

Hence for the above example, the characteristic function for class 4 would be $\delta_4(\boldsymbol{\gamma}) = [0, 0, 0, 0.8408, 0, 0]^T$. Using only the coefficients associated with the *i*th class, the given test

[1] Please refer to [37] for details.
[2] This example is solved using the MATLAB implementation of Gradient Projection for Sparse Reconstruction (GPSR) which is available online on http://www.lx.it.pt/~mtf/GPSR/.

sample $\mathbf{S}$ is approximated as $\hat{\mathbf{S}}_i = \mathbf{D}\delta_i(\boldsymbol{\gamma})$. $\mathbf{S}$ is then assigned to the object class, $\mathbb{C}_{\mathbf{S}}$, that gave the smallest residual between $\mathbf{S}$ and $\hat{\mathbf{S}}_i$:

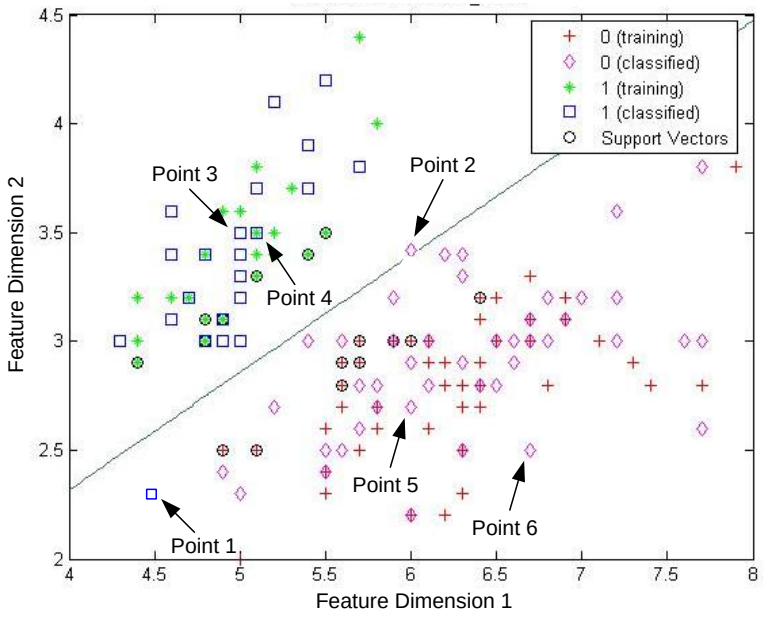
$$\mathbb{C}_{\mathbf{S}} = \arg\min_i r_i(\mathbf{S}) \quad where \quad r_i(\mathbf{S}) = \left\|\mathbf{S} - \hat{\mathbf{S}}_i\right\|_2 \tag{13}$$

2.3. Comparison of SVM and SRC classification

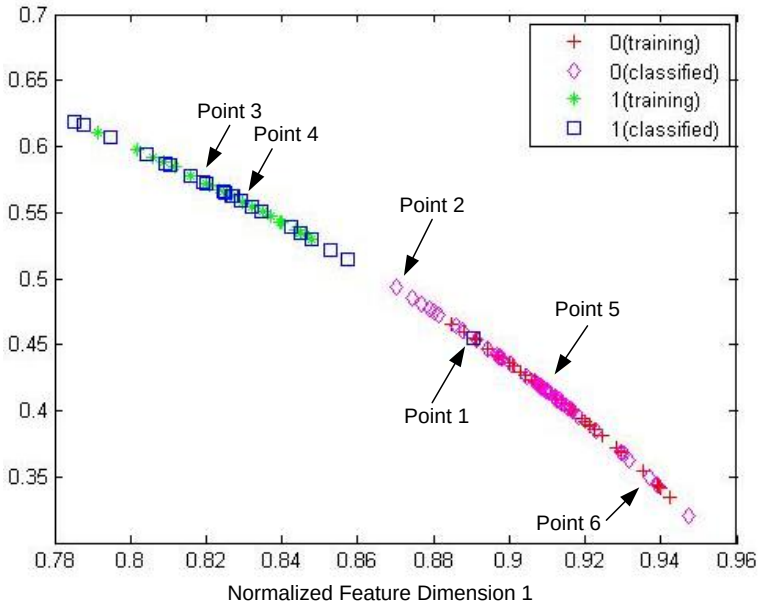
A comparison of SVM and SRC in terms of recognition performance was conducted with the aim of understanding the similarities and differences between the classifiers. We considered simple 2-dimensional data for easy visualization, as shown in Fig. 1. For sparse representation-based classification, all the samples are normalised to have unit ℓ_2-norm, which matches the length normalization in the SVM kernel as shown in Fig. 1 (b). This experiment is conducted on the Fisher iris data [46] using the sepal length and width for classifying data into two groups: Setosa and non-Setosa shown as "Class 1" and "Class 0" respectively on Fig. 1. The experiment was repeated 20 times, with the training and testing sets selected randomly.

Notably, the performance of SRC matches that of the SVM in 19 out of the 20 trials. Similarly to SVM, the sparse representation approach also finds it difficult to classify the same test point indicated as "point 1" in Fig. 1 (a) for SVM and (b) for SRC, since it is in the subspace of class 0 for both classifiers. However "point 2" (shown in Fig. 1) is correctly classified as class 0 for SRC and misclassified as class 1 by SVM. This could be because SVM does not adapt the number and type of supports to each test example. It selects a sparse subset of relevant training data, known as support vectors (shown as circles in Fig. 1 (a)) which correspond to the data points from the training set lying on the boundaries of the trained hyperplane, and uses these supports to characterize "all" data in the test set. Although visually "point 2" is closer to the training subset of class 0, it is misclassified since it is on the left hand side of the hyperplane, corresponding to class 1. SRC allows a more adaptive classification with respect to the test sample by changing the number and type of support training samples for each test sample [47] as shown in the sparse coefficients of four test samples (Fig. 1 (c) – (f)) chosen from Fig. 1 (b), indicated as "point 3" to "point 6" respectively, whereas the SVM classifies with the same support vector weights as shown in Fig.

1 (c) – (f) across all test data in the test set. In addition, Fig. 1 supports the concept that test samples can be represented as a linear combination of the training samples from the same class since it can be observed from Fig. 1 (c) – (d) that for test samples from Class 1 (indicated as Point 3 and 4 on Fig. 1(b)), the sparse coefficients have larger values for the dictionary indices belonging to class 1 and the same applies to Point 5 and 6 from Class 0 (shown in Fig. 1(e) – (f)).



(a)



(b)

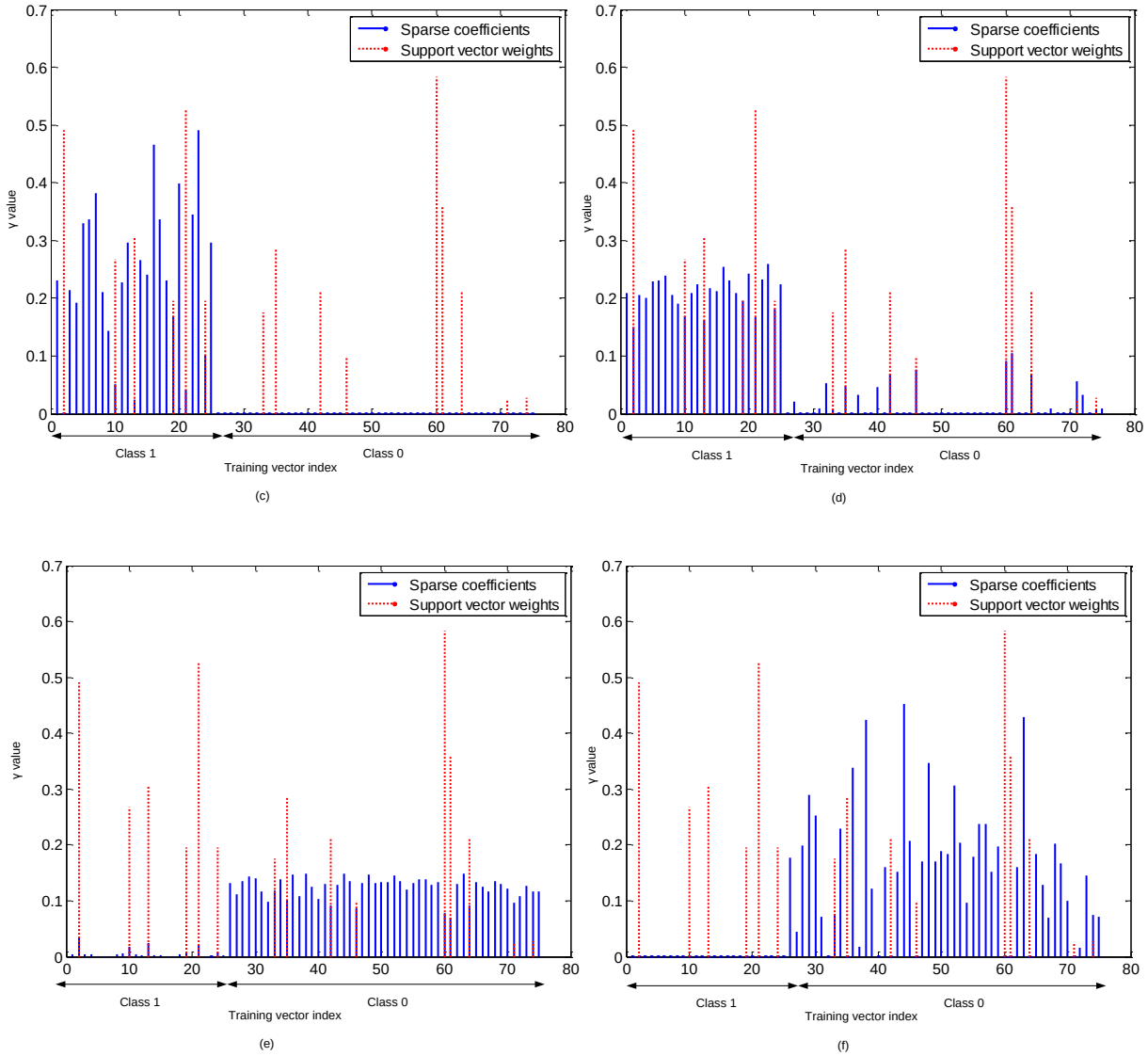


Fig. 1 Comparison between (a) SVM and (b) SRC for a two-class problem (class 0 and class 1) where '+' and '*' correspond to the training set instances for class 0 and class 1 respectively. ◊ and □ correspond to the test points for class 0 and class 1 respectively. ○ are the support vectors chosen from the training data sets of each class for SVM. (c) – (f) The values of the sparse coefficients and weights of the support vectors (shown in Fig. 1 (a)) for test points 3 – 6 respectively

3. *i*-vector-based SRC

In this work we explore the use of SRC for speaker verification since many experimental results reported in the literature indicate that SRC can achieve a generalization performance that is better than or equal to other classifiers [31, 35-37].

In [35], Naseem et al proposed the use of the GMM mean supervector, **M**, to develop an over-complete dictionary using all the training utterances of speakers in a database for speaker identification. Likewise, we employed a similar approach termed GMM-Sparse Representation Classification (GMM-SRC) in the context of speaker verification in our previous work [39]. However the sparse representation of large dimension supervectors requires a large amount of memory due to the over-complete dictionary, which can limit the training sample numbers and could slow down the recognition process. Motivated by [41], where the authors proposed the use of *i*-vectors as features for the SRC, we adopt the same approach with the use of *i*-vectors as feature vectors for the SRC.

The underlying structure and detailed architecture of the i-vector-based SRC, which we term *i*-vector− Sparse Representation Classification (*i*-SRC) is shown in (14) and Fig. 2 respectively.

$$\mathbf{D} = \left[\mathbf{D}_{tar}\mathbf{D}_{bg}\right] \tag{14a}$$

$$\mathbf{D}_{tar} = \left[\mathbf{q}_{tar,1}, \dots, \mathbf{q}_{tar,l_{tar}}\right] \tag{14b}$$

$$\mathbf{D}_{bg} = \left[\mathbf{q}_{bg,1}, \dots, \mathbf{q}_{bg,l_{bg}}\right] \tag{14c}$$

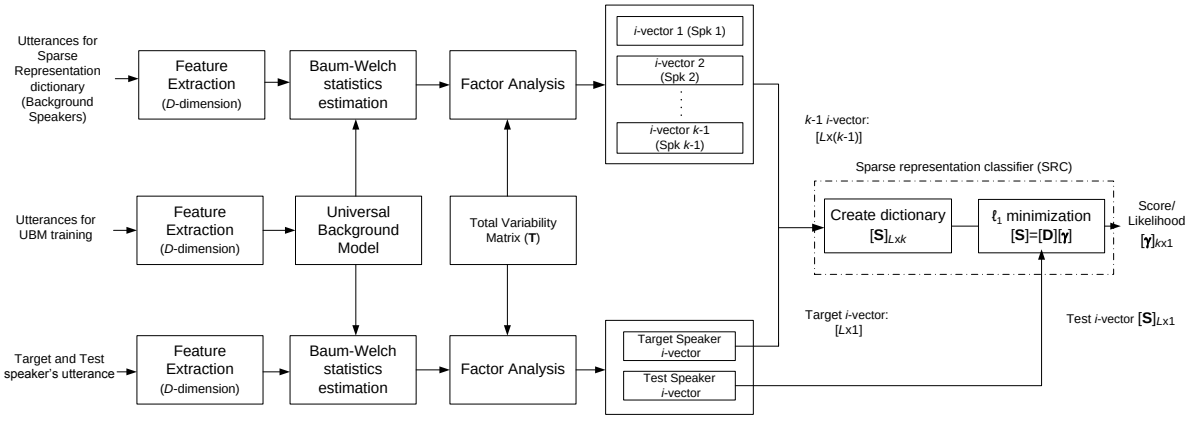


Fig. 2 Architecture of the *i*-SRC system.

The over-complete dictionary (**D**) is composed of the normalized i-vectors (with unit ℓ_2 norm) of training utterances from the target speaker ($\mathbf{D}_{tar}$) and the background speakers ($\mathbf{D}_{bg}$). The normalization process is analogous to the length normalization in the SVM kernel and in this paper the dictionary data

composition is the same as the kernel training data for SVM unless otherwise specified. In the context of speaker verification, usually $l_{bg} \gg l_{tar}$, with l_{tar} equal to 1, where l_{bg} and l_{tar} represent the number of utterances from the background and target speakers respectively.

Following this, the *i*-vector of a test utterance (**S**) from an unknown speaker are represented as a linear combination of this over-complete dictionary, a process referred to as sparse representation classification for speaker recognition, as follows

$$\mathbf{S} = \mathbf{D}\boldsymbol{\gamma} \tag{15}$$

Throughout the testing process, the background samples $\mathbf{D}_{bg}$ are fixed and only the target samples $\mathbf{D}_{tar}$ are replaced with respect to the claimed target identity in the test trial.

In the context of speaker verification, $\boldsymbol{\gamma}$ is sparse since the test utterance corresponds to only a very small fraction of the dictionary. As a result, $\boldsymbol{\gamma}$ will have large $\boldsymbol{\psi}$ corresponding to the correct target speaker of the test utterance as shown in Fig. 3(a), where the dictionary index *k*=1 corresponds to the true target speaker. On the other hand, if the test utterance is from a false target speaker, the coefficients will be sparsely distributed across multiple speakers in the dictionary [36, 39], as shown in Fig. 3(b). As shown in Fig. 3, the membership of the sparse representation in the over-complete dictionary itself captures the discriminative information since it adaptively selects the relevant vectors from the dictionary with the fundamental assumption that test samples from a class lie in the linear span of the dictionary entries corresponding to the class of the test samples [31, 37]. Therefore, given sufficient training samples from each speaker, any new sample **S** from the same speaker can be expressed as a linear combination of the corresponding training samples. This assumption is valid in the context of speaker recognition since it has been shown by Ariki et al. that each individual speaker has their own subspace [48, 49]. In addition, even though the number of background examples significantly outweighs that of target speaker examples, the SRC framework is not affected by the unbalanced training set which is in contrast to an SVM system which requires tuning of the SVM cost values. This is because for SVM, a hyperplane trained by an unbalanced training set will be biased toward the class with more training samples [50, 51], but this is not

the case for SRC. On the other hand, SRC utilizes the highly unbalanced nature of the training example to form a sparse representation problem [41].

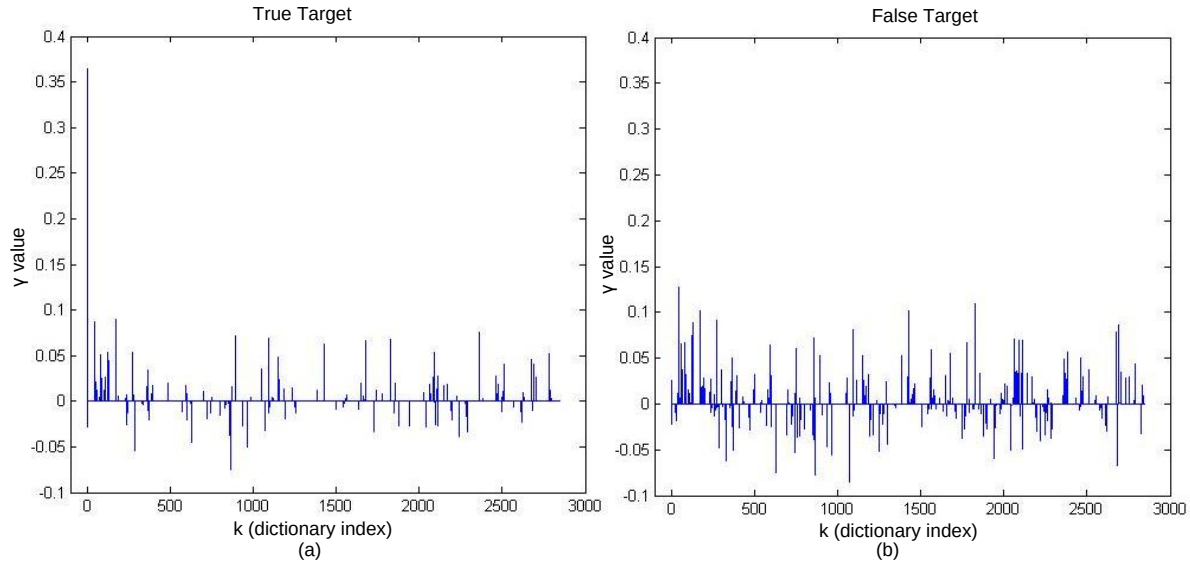


Fig. 3 The sparse solution γ of two example speaker verification trials (a) True target (k = 1) (b) False target

Then the ℓ_1-norm ratio, $\wp$ shown in (16) is used as the decision criterion for verification, where the operator δ_{target} selects only the coefficients associated with the target class [41]. The example shown in Fig. 3 has target ℓ_1-norm of 0.1828 and 0.0537 for the true target (a) and false target (b) respectively. Although three different decision criteria are proposed in [41], our experiments showed that using the ℓ_1-norm ratio gave the best performance.

$$\wp = \left\|\delta_{target}(\boldsymbol{\gamma})\right\|_1 / \|\boldsymbol{\gamma}\|_1 \qquad (16)$$

4. System Development Using SRC

4.1. Database

All experiments reported in this section were carried out on the female subset of the core condition of the NIST 2006 speaker recognition evaluation (SRE) as development dataset for model parameter tuning which will be evaluated on NIST 2010 SRE in section 5. For each target speaker model, a five-minute telephone conversation recording is available containing roughly two minutes of speech for a given

speaker. In the NIST evaluation protocol, all previous NIST evaluation data and other corpora can be used in system training, and we also adopt this protocol.

4.2. Experimental Setup

The front-end of the recognition system includes an energy based speech detector [52] which was applied to discard silence and noise frames. A Hamming window of 20ms (overlap of 10ms) was used to extract 19 mel frequency cepstral coefficients (MFCCs) together with log energy. This 20-dimensional feature vector was subjected to feature warping using a 3s sliding window, before computing delta coefficients that were appended to the static features.

Three current state of the art systems, namely GMM-SVM [53], *i*-vector based SVM (*i*-SVM) [22] and *i*-vector based CDS (*i*-CDS) [22] were implemented as baseline systems. They are all based on the universal background model (UBM) paradigm [4], so we have used gender-dependent UBMs of 2048 Gaussians trained using NIST 2004. In our SVM system, we took 2843 female SVM background impostor models from NIST 2004 to train the SVM. In addition, for the GMM-SVM system, NAP (rank 40) trained using NIST 2004 and 2005 SRE corpus was incorporated to remove unwanted channel or intersession variability [53]. On the other hand for *i*-SVM and *i*-CDS, LDA (trained using Switchboard II, NIST 2004 and 2005 SRE) with dimensionality reduction (dim = 200) followed by WCCN (trained using NIST 2004 and 2005 SRE) were used for session compensation[3] [21]. For i-vector based systems, the total variability space matrix was trained using LDC releases of Switchboard II, Phases 2 and 3; switchboard Cellular, Parts 1 and 2 and NIST 2004-2005 SRE. The total variability matrix was composed of 400 total factors. Finally, the decision scores were normalized using zt-norm (z-norm followed by t-norm) using 367 female t-norm models and 274 female z-norm utterances from NIST 2004 and 2005 SRE respectively. Note that any utterances from speakers in NIST 2005 that appear in NIST 2006 have been

[3] The combination/configuration of LDA and WCCN was determined experimentally through development on NIST 2006 SRE and the best results were reported.

excluded from the training set. The speaker verification results for all the baseline systems are shown in Table 1.

In the following subsections, results for various SRC systems will be presented, unless specified all optimization was performed by the Gradient Projection for Sparse Reconstruction (GPSR) [54] MATLAB toolbox[4] and no score normalisation are performed. Alternatively, other freely available MATLAB toolbox including ℓ_1-magic [55], SparseLab [56] and l1_ls [57] can be used. During initial investigations, all toolboxes gave similar performance so GPRS was chosen as it is significantly faster, especially in large-scale settings [54]. Score normalisation (i.e TNorm) has been excluded from the SRC system because the conventional way of score normalisation (individual scoring against each TNorm model) slows down the verification process significantly (by a factor of three to six depending on the number of TNorm model and dictionary size) as compared with other systems (i.e SVM, CDS). Although a novel SRC-based TNorm has been proposed in [41] through the replacement of the Tnorm data as the background samples in the over-complete dictionary, no performance improvement were observed in the -proposed method over the conventional Tnorm as reported in [41]. In addition, the direct replacement of the background samples in the over-complete dictionary using TNorm data seems somewhat heuristic.

Table 1: Baseline speaker verification results on the NIST 2006 Female Subset database

Systems	EER (%)	minDCF
GMM-SVM	14.79	0.0760
GMM-SVM + NAP	5.78	0.0285
i-SVM + LDA + WCCN	4.40	0.0230
i-CDS + LDA + WCCN	4.31	0.0222

[4] Gradient Projection for Sparse Reconstruction (GPSR) MATLAB toolbox is available online on http://www.lx.it.pt/~mtf/GPSR/

4.3. i-vector-based SRC

In this section, we evaluate the *i*-SRC system in comparison with *i*-SVM and *i*-CDS. The dictionary $\mathbf{D}_{bg}$ matrix of SRC was composed of 2843 utterances from NIST 2004 SRE database, which was the same as the background training speaker database for SVM. Furthermore, we tried various channel compensation steps in the total variability space that are reported in [21] and the best performance for *i*-SRC was found to be based on LDA (*i*-SRC-LDA) with an EER of 5.03%. This result shows that the initial performance of the *i*-SRC is slightly worse than that of *i*-SVM and *i*-CDS. In the following sub-sections, we investigate some techniques presented in [21, 36, 41, 58] with a view to improving the system performance.

4.4. Robustness to corruption

In many practical recognition scenarios, the test sample **S** can be partially corrupted due to large session variability. Thus it has been suggested in [31, 36, 41] to introduce an error vector ***e*** into the linear model in (17) as follows

$$\mathbf{S} = \mathbf{D}\boldsymbol{\gamma} + \mathbf{e} = [\mathbf{D}\ \mathbf{I}]\begin{bmatrix}\boldsymbol{\gamma}\\ \mathbf{e}\end{bmatrix} \doteq \mathbf{B}\mathbf{w} \tag{17}$$

Here, $\mathbf{B} = [\mathbf{D}, \mathbf{I}] \in \mathbb{R}^{K\times(N+K)}$ so the system is always underdetermined. As before, the sparsest solution **w** is recovered by solving the following extended ℓ_1-minimization problem

$$\begin{aligned}\widehat{\mathbf{w}} &= \min\|\mathbf{w}\|_1 \ \ subject\ to\ \mathbf{S} = \mathbf{B}\mathbf{w}\\ \widehat{\mathbf{w}} &= [\hat{\boldsymbol{\gamma}}\ \ \hat{\mathbf{e}}]^t \in \mathbb{R}^{N+K}\end{aligned} \tag{18}$$

If the error vector **e** is sparse and has no more than $\frac{K+l_{tar}}{2}$ nonzero entries, the new sparse solution $\widehat{\mathbf{w}}$ is the true generator [31]. Finally, the same decision criterion in (1) is used for verification.

Here we briefly illustrate the effect of including the identity matrix in the overcomplete dictionary and show the incremental improvement in accuracy for purposes of completeness. An example speaker from NIST 2006 database was chosen, such that the test speaker's *i*-vector had a large outlier in the third dimension relative to its training*i*-vector, as shown in Fig. 4(a) and (b) respectively. It has been reported

in [31, 59] that the identity matrix will capture any redundancy between the test sample and dictionary, hence the outlier is captured by the identity matrix at the location corresponding to the third dimension in this example, for an original dictionary size of k = 2844 as shown in Fig. 4(c). The inclusion of the identity matrix in the dictionary improves the recognition performance from 5.03% to 4.73% EER. The improvement supports the claim in [31, 36, 41] that by adding a redundant identity matrix at the end of the original over-complete dictionary, the sparse representation is more robust to variability.

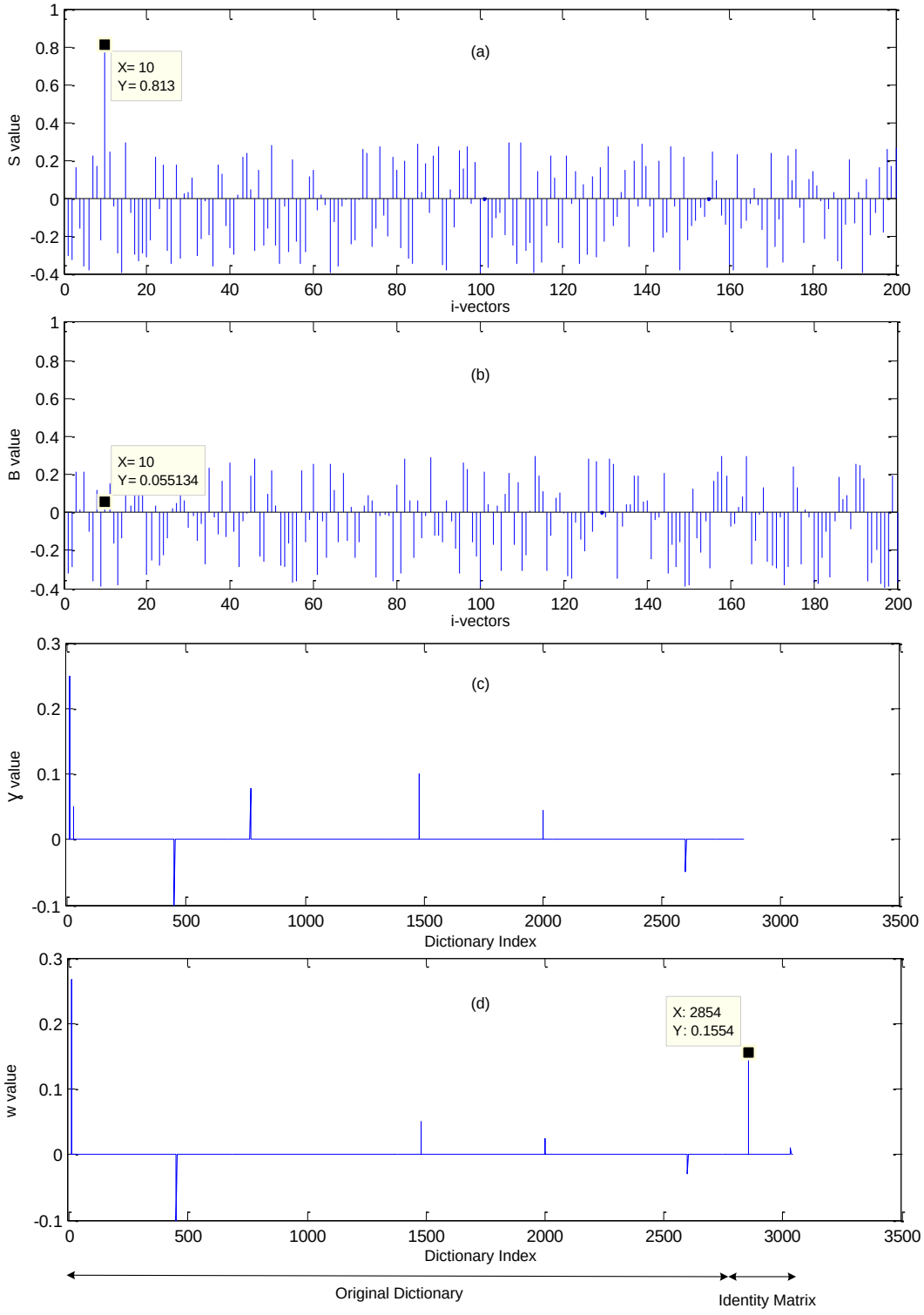


Fig. 4 Illustration of inclusion of identity matrix (a) Test speaker's *i*-vector (b) Target speaker's *i*-vector (for dictionary index = 1) (c) Sparse solution γ without identity matrix (d) Sparse solution $\hat{w}$ with identity matrix included

4.5. Sparseness constraint

The use of exemplar-based techniques for both speech classification and recognition tasks has become increasingly popular in recent years. In [58], the appropriateness of different types of sparsity regularization constraints on $\boldsymbol{\gamma}$ in speech processing applications was analysed. Sparseness methods such as LASSO [60] and Bayesian Compressive Sensing (BCS) [61], using an ℓ_1 sparseness constraint, Elastic Net [62], which uses a combination of an ℓ_1 and ℓ_2 constraint and Approximate Bayesian Compressive Sensing (ABCS) [37], which uses an ℓ_1^2 constraint, were compared. Since the results reported in [58] for the various techniques for sparsity constraint coupled with an ℓ_2 norm show almost similar results among the above techniques, Elastic Net (which gave the best performance reported in [58]) was selected for comparison in this section. It can be formulated as follows:

$$\min_{\boldsymbol{\gamma}} \|\mathbf{S} - \mathbf{D}\boldsymbol{\gamma}\|_2 + \lambda\|\boldsymbol{\gamma}\|_1 + (1-\lambda)\|\boldsymbol{\gamma}\|_2^2 , \ \ where\ \lambda \in [0,1) \tag{19}$$

where $\lambda\|\boldsymbol{\gamma}\|_1 + (1-\lambda)\|\boldsymbol{\gamma}\|_2^2$ is termed the elastic net penalty, which is a convex combination of the LASSO and ridge regression [63]. Ridge regression is an exemplar-based technique that uses information about all training examples in the dictionary to make a classification decision about the test example, in contrast to sparse representation techniques that constrain $\boldsymbol{\gamma}$ to be sparse. When $\lambda = 0$, the naïve elastic net penalty becomes simple ridge regression and when $\lambda = 1$, it becomes LASSO. In this section, Elastic Net is implemented using the Glmnet MATLAB package[5] [64] with $\lambda = 0.6$ since it gave the best EER as shown in Fig. 5.

[5] MATLAB implementation of Glmnet is available online on http://www-stat.stanford.edu/~tibs/glmnet-matlab/.

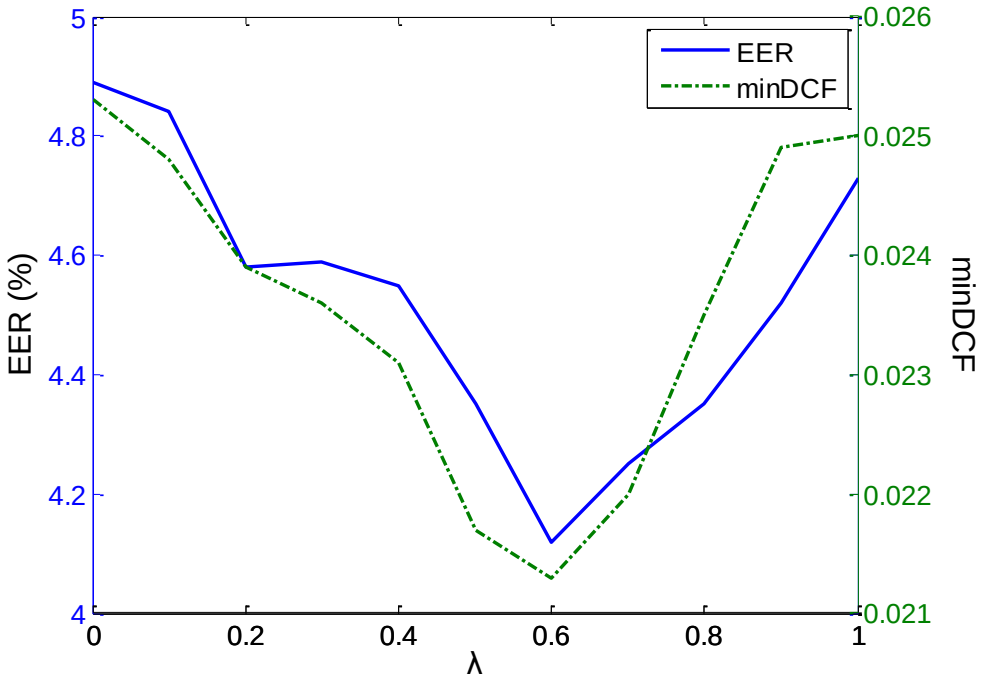


Fig. 5 Speaker recognition performance (EER: left y-axis, solid line and minDCF: right y-axis, dash-dot line) on NIST 2006 as the elastic net penalty, λ, is refined.

Table 2: Speaker verification results on the NIST 2006 SRE Female Subset database

Systems	EER (%)	minDCF
i-SRC-LDA (DIM = 200) with ℓ_1-constraint	4.73	0.025
i-SRC-LDA (DIM = 200) with ℓ_2-constraint	4.89	0.0253
i-SRC-LDA (DIM = 200) with ℓ_1 and ℓ_2 -constraint	4.12	0.0213
i-SRC-LDA (DIM = 200) with quadratic constraints [36, 41]	4.40	0.0233

As shown in Fig. 5 and Table 2, the method using only ℓ_1 norm or ℓ_2 norm has slightly lower accuracy, showing the decrease in accuracy when a high or low degree of sparseness is enforced respectively (similar results are observed in [58]). Thus, it appears that using a combination of a sparsity constraint on $\boldsymbol{\gamma}$, coupled with an ℓ_2 norm, does not force unnecessary sparseness and offers the best performance. Furthermore, the ℓ_1-minimization with quadratic constraints system as proposed in [36, 41]

has been included in Table 2 for comparisons. From the results, we could observe that the Elastic Net performs slightly better than the ℓ_1-minimization with quadratic constraints system.

4.6. Proposed dictionary design

In recent years, apart from the study of different pursuit algorithms for sparse representation, the design of dictionaries to better fit a set of given signals has attracted growing attention [65-68]. As mentioned previously, McLaren et al. [15] proposed SVM background speaker selection algorithms for speaker verification. In this section, a similar idea, which we termed column vector frequency, is considered for choosing the dictionary of SRC based on the total number of times each individual column of the background dictionary $(\mathbf{D}_{bg})$ is chosen, as shown in (20)

$$\mathbf{D}_{bg} = \left[\mathbf{y}_{bg,1}\, \mathbf{y}_{bg,2}\, \cdots\, \mathbf{y}_{bg,l_{bg}}\right]$$
$$\mathfrak{V}(\mathbf{y}_{bg,t}) = \sum_{c=1}^{P} \mathfrak{M}(\alpha_{bg,t}^{c}) \;\; where\; \mathfrak{M}(x) = \begin{cases} 1, x \neq 0 \\ 0, x = 0 \end{cases} \tag{20}$$

where t is the column index of the background dictionary with values from 1 to l_{bg}, P is the number of test trials, $\alpha_{bg,t}$ is the sparse coefficient for the t^{th} column of the background dictionary and $\mathfrak{V}$ is the frequency counter for the corresponding t^{th} column.

Table 3: Results from NIST 2006 SRE using different dictionary datasets

Dictionary	EER (%)	minDCF
NIST 2004	4.12	0.0213
NIST 2005	4.53	0.0245
NIST 2004 + NIST 2005	4.33	0.0237

First, the results using a number of different dictionary dataset configurations without any background speaker selection (with ℓ_1+ℓ_2 constraint, $\lambda = 0.6$) are detailed in Table 3. It has be observed that using the NIST 2004 dataset alone gave the best performance, which is the same as the results

reported for SVM in [16]. Combining the NIST 2004 dataset with NIST 2005 resulted in the degradation of EER performance despite the significant increase in the number of impostor examples.

Table 4: Performance on NIST 2006 female trials when using SRC background datasets refined by impostor column vector frequency.

Dictionary	EER (%)	minDCF
Full Dataset	4.33	0.0237
500 highest ranked frequency	3.99	0.0212
500 lowest ranked frequency	5.65	0.0371

As an initial indicator of whether the column vector frequency is an adequate metric to represent the suitability of a background speaker, the 500 highest ranked and 500 lowest ranked background speakers from the NIST 2004 (2843 speakers) and NIST 2005 (673 speakers) datasets based on column vector frequency were selected on gender-dependent basis and the evaluation results are detailed in Table 4. The performance demonstrates that the dictionary chosen based on a column vector frequency basis is an appropriate measure of the impostor example. Furthermore, to determine an optimal size for the dictionary, the experiment was repeated using only the highest R column vector frequencies with R varying from 300 to 3516 in steps of 200. The resulting EER and minDCF were approximately 3.99% and 0.0212 respectively for values of R in the range of 500 to 2500 as shown in Fig. 6(a), indicating that a smaller size dictionary can be used. In addition, a 79% relative reduction in computation time is achieved using the refined dictionary over the full dictionary (as shown in Fig. 6(b)), allowing a faster verification process. The refined dictionary with R=500 will be used for all subsequent experiments and will be shown to generalize well to the NIST 2010 dataset in Section 5. On the other hand, despite the significant improvement in time, the SRC is still somewhat slower than the *i*-SVM (1800s) and significantly slower than *i*-CDS scoring (244s) for scoring on the full database.

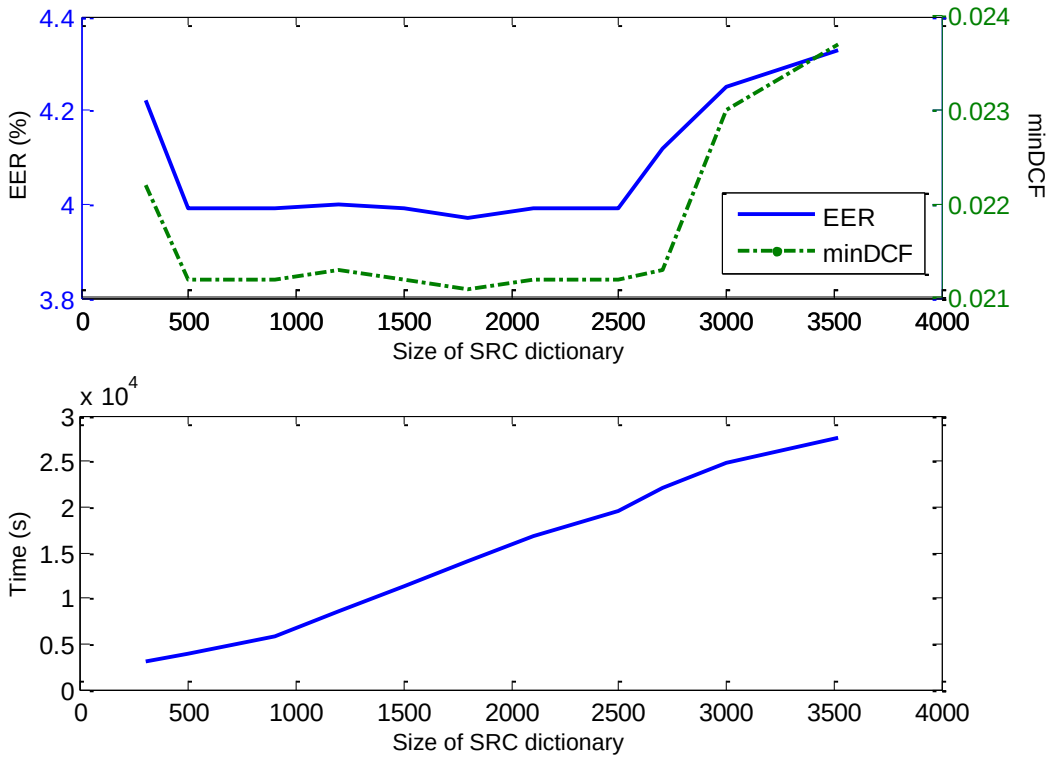


Fig. 6 Speaker recognition performance on NIST 2006 as the SRC dictionary is refined. (a) EER (left y-axis, solid line) and minDCF (right y-axis, dash-dot line) (b) Total time taken (in seconds) for computing the ℓ_1-norm score across all test utterances.

Next, we compare the results reported in this paper with the best baseline system configuration reported in [41] which is based on ℓ_1–minimization with ℓ_1-constraint[6], inclusion of identity matrix, Bnorm-(ℓ_2-residual) scoring and TNorm (conventional). Using these configurations on NIST 2006 SRE database (female subset), an EER=4.55% and minDCF=0.0248 was achieved. It could be observed that similarly to other classifiers, incorporating TNorm does improve the EER performance (from 4.73%). Furthermore, comparing the result with Table 2 and Table 4, we observed that sparse representation based on a combination of ℓ_1 and ℓ_2 constraint on $\boldsymbol{\gamma}$ outperformed the proposed system in [41] significantly, with a relative EER reduction of 12.3%. This improvement seems to be mainly attributable to the degree of sparseness constraint on $\boldsymbol{\gamma}$**.** In addition, a faster verification process can be achieved with a smaller

[6] The ℓ_1-constraint refers to the constraint on $\boldsymbol{\gamma}$ (as discussed in section 4.5) and not the quadratic constraints on the error tolerance as indicated in [41] M. Li, X. Zhang, Y. Yan, and S. Narayanan, "Speaker Verification using Sparse Representations on Total Variability I-Vectors," in *Proc. of INTERSPEECH*, 2011..

dictionary refined based on column vector frequency, as opposed to the direct heuristic replacement of the dictionary with TNorm samples in [41].

5. Speaker Recognition Experiments on NIST 2010 SRE

In this section, the classifiers were evaluated using the larger and more contemporary *extended* NIST 2010 database, in order to see the database independency of the results. Results are reported for the five evaluation conditions with normal vocal effort, corresponding to det conditions 1-5 in the SRE'10 evaluation plan [71], which include *int-int, int-tel, int-mic* and *tel-tel*.

We used exactly the same UBM and total variability configuration as in Section 4. The only difference lay in the amount of data used to train the UBM, total variability parameters, WCCN, LDA and SVM impostor with respect to the evaluation conditions. We added the Mixer 5 and interview data taken from the follow-up corpus of the NIST 2008 SRE for interview (*int*) conditions, NIST 2005 and 2006 SRE microphone segments for microphone (*mic*) conditions and NIST 2006 SRE for telephone (*tel*) conditions. Table 5 summarises the datasets used to estimate our system parameters. Similarly to the previous setup (in Section 4.2), any common utterances from speakers in the NIST 2008 follow up and NIST 2010 databases have been excluded from the training set.

The performance of each classifier for each condition is given in Table 7. The results show that i-SRC ($\lambda = 0.6$) obtained the best performance in terms of EER, followed by i-CDS and i-SVM. Interestingly, the i-SRC approach performs better than all SVM variants in all conditions with just a single dictionary, designed according to the column vector frequency (X = 500) in Section 4.6, which indicates that the dictionary generalises well to different types of common conditions. On the other hand, for SVM-based systems, different background data sets need to be constructed separately for different conditions (i.e *int-int, int-tel, int-mic and tel-tel*) [72, 73] Table 6 shows the results with the best configuration. In addition, the i-SRC outperforms the i-CDS, which is of interest since both do not require a training phase and additionally do not require any form of score normalisation based on a set of impostor models, or cohort (i.e Z- or T-Norm) to achieve good performance.

Next, we explore whether SRC provides complementary information to the conventional baseline, since the study of systems which fuse well has held sustained interest in the speaker recognition community in recent times [69]. The fused results of the baseline system (i-CDS) with i-SVM or i-SRC are shown in Table 7. The fusion weights are estimated using the NIST 2008 evaluation data. The results demonstrated that the fusion of i-CDS and i-SRC is better than the fusion of i-CDS and i-SVM. In contrast, the fusion of i-SRC and i-SVM (shown in Table 7) results in minimal improvement in EER since both of the classifiers have very similar classification decisions for most of the test points, as explained in Section 2.3.

Table 5: Corpora used to estimate UBM, WCCN, LDA, SVM impostors, Z- and T-norm data for evaluation on NIST 2010 SRE.

	Switchboard II	Mixer 5	NIST 2004	NIST 2005	NIST 2006	NIST 2008 follow up
UBM			x	x	x	
t-norm			x			
z-norm				x		
T	x		x	x	x	x
WCCN		x	x	x	x	x
LDA	x	x	x	x	x	x

Table 6: Speaker verification performance on the *extended* NIST 2010 evaluation protocol. Note that DCF_{new} corresponds to the DCF with speaker detection cost model parameters of $C_{Miss} = 1$, $C_{FalseAlarm} = 1$, $P_{Target} = 0.001$

Common Condition	i-CDS		i-SRC		i-SVM	
	EER	DCF_{new}	EER	DCF_{new}	EER	DCF_{new}
1 (*int-int*)	3.05	0.557	**2.91**	**0.522**	3.40	0.591
2 (*int-int*)	4.51	0.654	**4.01**	**0.597**	4.81	0.690
3 (*int-tel*)	4.72	0.682	**4.32**	**0.628**	5.13	0.701
4 (*int-mic*)	4.12	0.599	**3.80**	**0.543**	4.44	0.651
5 (*tel-tel*)	3.35	0.568	**2.95**	**0.518**	3.71	0.598

Table 7: Fused speaker verification performance of JFA-SVM, JFA-CDS or JFA-SRC with JFA on *extended* NIST 2010 SRE database with speaker detection cost model parameters of C_{Miss} = 1, $C_{FalseAlarm}$ = 1, P_{Target} = 0.001 (EERx100, minDCFx1000)

System	Common Condition 1		Common Condition 2		Common Condition 3		Common Condition 4		Common Condition 5	
	EER	minDCF	EER	minDCF	EER	minDCF	EER	minDCF	EER	minDCF
i-CDS + i-SRC	**2.34**	**0.449**	**3.51**	**0.546**	**3.65**	**0.573**	**3.47**	**0.498**	**2.46**	**0.444**
i-CDS + i-SVM	2.63	0.507	4.17	0.591	4.44	0.630	3.78	0.554	2.92	0.513
i-SVM + i-SRC	2.85	0.510	3.81	0.580	4.01	0.601	3.65	0.516	2.73	0.485

6. Conclusion

In this paper, we investigated the different types of sparseness methods and dictionary composition of sparse representation classification (SRC) for speaker verification using i-vectors from the total variability model. Inspired by the principles of the sparse representation model and based on the intuitive hypothesis that a speaker can be represented by a linear combination of training samples from the same speaker, we first compute the sparse representation through ℓ_1-minimization, and classification is achieved based on an ℓ_1-norm ratio. Since SRC has only recently appeared in the context of speaker recognition, we evaluated a range of existing techniques for sparse representation classification and examined the effect on speaker recognition performance.

First, we observed that the inclusion of the identity matrix in the dictionary results in a relative reduction of 6% in EER on NIST 2006 SRE, and appear to be an essential aspect of the dictionary composition. Next, a sparseness method that uses a combination of ℓ_1 and ℓ_2 (Elastic net), offers better performance than one with only an ℓ_1 constraint, since the latter enforces a high degree of sparseness which leads to a decrease in accuracy. Finally, motivated by background speaker selection for the SVM-based system, we proposed the SRC background dataset selection based on column vector frequency. We demonstrated that a smaller dictionary refined by column vector frequency could be used, allowing a faster verification process. Furthermore, we showed that the dictionary chosen for development on NIST 2006 SRE generalised well to the evaluation on NIST 2010 SRE corpus for different evaluation condition,

as opposed to SVM background data, which require significant amounts of tuning based on the evaluation condition.

In addition, experiments on NIST 2010 database validated the findings that the sparse representation approach can outperform the best performance achieved by CDS or SVM. Finally, by fusing i-SRC with the conventional i-CDS system, we show that the overall system performance is improved, providing a relative reduction in EER of 8 – 19% over i-SRC alone, and the fusion of i-CDS with i-SRC outperformed the fusion of i-CDS with i-SVM in the range of 8-18% relative reduction in EER. Although care has been taken in this paper to investigate many aspects of SRC-based speaker recognition, it is highly possible that these results can be further improved with more research, for example into areas such as score normalization techniques for sparse representation, which remains an underexplored problem in the literature for SRC-based recognition applications.

ACKNOWLEDGMENT

The authors would like to thank Dr Kong Aik Lee and Dr Haizhou Li for their help with the implementation of the Joint Factor Analysis system.

REFERENCES

[1] **W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Computer Speech & Language,* vol. 20, pp. 210-229, 2006.**

[2] **V. Wan and W. M. Campbell, "Support vector machines for speaker verification and identification," in *IEEE Workshop Neural Networks for Signal Processing*, 2000, pp. 775-784.**

[3] **D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication,* vol. 17, pp. 91-108, 1995.**

[4] **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.**

[5] **W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters,* vol. 13, pp. 308-11, 2006.**

[6] **B. G. B. Fauve, D. Matrouf, N. Scheffer, J. F. Bonastre, and J. S. D. Mason, "State-of-the-art performance in text-independent speaker verification through open-source software," *IEEE Transactions on Audio, Speech and Language Processing,* vol. 15, pp. 1960-8, 2007.**

[7] **N. A. Gunasekara, "Meta learning on string kernel SVMs for string categorization," Master of Computer and Information Sciences, AUT University, 2010.**

[8] **O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Machine Learning,* vol. 46, pp. 131-159, 2002.**

[9] **H. Frohlich and A. Zell, "Efficient parameter selection for support vector machines in classification and regression via model-based global optimization," in *International Joint Conference on Neural Networks*, 2005, pp. 1431-1436.**

[10] **P. J. Moreno and P. P. Ho, "A new SVM approach to speaker identification and verification using probabilistic distance kernels," in *Proc. of EUROSPEECH*, 2003, pp. 2965-2968.**

[11] **Z. N. Karam and W. M. Campbell, "A multi-class MLLR kernel for SVM speaker recognition," in *Proc. of ICASSP*, 2008, pp. 4117-4120.**

[12] **V. Wan and S. Renals, "Evaluation of kernel methods for speaker verification and identification," in *Proc. of ICASSP*, 2002, pp. 669-672.**

[13] **C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, pp. 121-167, 1998.**

[14] **Y. Lei, T. Hasan, J. W. Suh, A. Sangwan, H. Boril, G. Liu, K. Godin, C. Zhang, and J. H. L. Hansen, "The CRSS systems for the 2010 NIST speaker recognition evaluation."**

[15] **M. McLaren, R. Vogt, B. Baker, and S. Sridharan, "Data-driven impostor selection for T-norm score normalisation and the background dataset in SVM-based speaker verification," *Advances in Biometrics*, pp. 474-483, 2009.**

[16] **M. McLaren, B. Baker, R. Vogt, and S. Sridharan, "Improved SVM speaker verification through data-driven background dataset collection," in *Proc. of ICASSP*, 2009, pp. 4041-4044.**

[17] **J. W. Suh, Y. Lei, W. Kim, and J. H. L. Hansen, "Effective background data selection in SVM speaker recognition for unseen test environment: more is not always better," in *Proc. of ICASSP*, 2011.**

[18] **C. Q. Alex Solomonoff, and William M. Campbell, "Channel Compensation for SVM Speaker Recognition," in *Proc Odyssey: Speaker and Language Recognition Workshop*, 2004, pp. 57-62.**

[19] **A. Solomonoff, W. M. Campbell, and I. Boardman, "Advances in channel compensation for SVM speaker recognition," in *Proc. of ICASSP*, 2005, pp. 629-32.**

[20] **P. Kenny, "Joint factor analysis of speaker and session variability: theory and algorithms," *Tech Report Online: http://www.crim.ca/perso/patrick.kenny*, 2005.**

[21] **N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 788-798, 2011.**

[22] **N. Dehak, R. Dehak, P. Kenny, N. Brümmer, P. Ouellet, and P. Dumouchel, "Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification," in *Proc. of INTERSPEECH*, 2009.**

[23] **P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 1435-47, 2007.**

[24] **P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Speaker and session variability in GMM-based speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, pp. 1448-1460, 2007.**

[25] **N. Dehak, "Discriminative and generative approaches for long- and short-term speaker characteristics modeling: application to speaker verification," Doctoral Dissertation, Ecole de Technologie Superieure 2009.**

[26] **N. Dehak, P. Kenny, R. Dehak, O. Glembek, P. Dumouchel, L. Burget, V. Hubeika, and F. Castaldo, "Support vector machines and joint factor analysis for speaker verification," in *Proc. of ICASSP*, 2009, pp. 4237-4240.**

[27] **A. O. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Proc. of INTERSPEECH*, 2006, pp. 1471 - 1474.**

[28] **E. J. Candès, "Compressive sampling," in *Proc. Int'l Congress of Mathematicians*, 2006.**

[29] **A. Y. Yang, M. Gastpar, R. Bajcsy, and S. S. Sastry, "Distributed sensor perception via sparse representation," *Proceedings of the IEEE*, vol. 98, pp. 1077–1088.**

[30] **K. Huang and S. Aviyente, "Sparse representation for signal classification," *Advances in Neural Information Processing Systems*, vol. 19, p. 609, 2007.**

[31] **J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 210-227, 2008.**

[32] **E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, pp. 489-509, 2006.**

[33] **R. G. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, p. 118, 2007.**

[34] **K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation,"** ***Neural computation,*** **vol. 15, pp. 349-396, 2003.**

[35] **I. Naseem, R. Togneri, and M. Bennamoun, "Sparse Representation for Speaker Identification," in** ***Proc. of ICPR,*** **2010, pp. 4460-4463.**

[36] **Ming Li and S. Narayanan, "Robust Talking Face Video Verification Using Joint Factor Analysis And Sparse Representation On GMM Mean Shifted Supervectors," in** ***Proc. of ICASSP,*** **2011.**

[37] **T. N. Sainath, A. Carmi, D. Kanevsky, and B. Ramabhadran, "Bayesian compressive sensing for phonetic classification," in** ***Proc. of ICASSP,*** **2010, pp. 4370-4373.**

[38] **A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose,"** ***IEEE Transactions on Pattern Analysis and Machine Intelligence,*** **vol. 23, pp. 643-660, 2001.**

[39] **J. M. K. Kua, E. Ambikairajah, J. Epps, and R. Togneri, "Speaker verification using sparse representation classification," in** ***Proc. of ICASSP,*** **2011, pp. 4548-4551.**

[40] **A. R. Webb,** ***Statistical pattern recognition*****: John Wiley & Sons Inc, 2002.**

[41] **M. Li, X. Zhang, Y. Yan, and S. Narayanan, "Speaker Verification using Sparse Representations on Total Variability I-Vectors," in** ***Proc. of INTERSPEECH,*** **2011.**

[42] **E. Amaldi and V. Kann, "On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems,"** ***Theoretical Computer Science,*** **vol. 209, pp. 237-260, 1998.**

[43] **D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution,"** ***Communications on pure and applied mathematics,*** **vol. 59, pp. 797-829, 2006.**

[44] **E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?,"** ***IEEE Transactions on Information Theory,*** **vol. 52, pp. 5406-5425, 2006.**

[45] **A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Feature selection in face recognition: A sparse representation perspective,"** ***submitted to IEEE Transactions Pattern Analysis and Machine Intelligence,*** **2007.**

[46] **C. Blake and C. J. Merz, "UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California,"** ***Department of Information and Computer Science,*** **vol. 460, 1998.**

[47] **T. N. Sainath, D. Nahamoo, B. Ramabhadran, and D. Kanevsky, "Sparse representation phone identification features for speech recognition,"** ***Speech and Language Algorithms Group, IBM, Tech. Rep,*** **2010.**

[48] **Y. Ariki and K. Doi, "Speaker recognition based on subspace methods," in** ***lCSLP,*** **1994, pp. 1859 - 1862.**

[49] **Y. Ariki, S. Tagashira, and M. Nishijima, "Speaker recognition and speaker normalization by projection to speaker subspace," in** ***Proc. of ICASSP,*** **1996, pp. 319-322.**

[50] **D. Tao, X. Tang, X. Li, and X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval,"** ***IEEE Transactions on Pattern Analysis and Machine Intelligence,*** **pp. 1088-1099, 2006.**

[51] **P. Campadelli, E. Casiraghi, and G. Valentini, "Support vector machines for candidate nodules classification,"** ***Neurocomputing,*** **vol. 68, pp. 281-288, 2005.**

[52] **T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: from features to supervectors,"** ***Speech Communication,*** **vol. 52, pp. 12-40, 2010.**

[53] **W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in** ***Proc. of ICASSP,*** **2006, pp. 97-100.**

[54] **M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,"** ***IEEE Journal of Selected Topics in Signal Processing,*** **vol. 1, pp. 586-597, 2007.**

[55] **E. Candes and J. Romberg.** ***1-MAGIC: Recovery of Sparse Signals via Convex Programming, 2005*****. Available: http://www.acm.caltech.edu/l1magic.**

[56] **D. Donoho, V. Stodden, and Y. Tsaig, "Sparselab,"** ***Software: http://sparselab. stanford. edu,*** **vol. 25, 2005.**

[57] **K. Koh, S. Kim, and S. Boyd, "l1 ls: A matlab solver for large-scale l1-regularized least squares problems," ed: Stanford University, Mar, 2007.**

[58] **D. Kanevsky, T. N. Sainath, B. Ramabhadran, and D. Nahamoo, "An analysis of sparseness and regularization in exemplar-based methods for speech classification," in *Proc. of INTERSPEECH*, 2010.**

[59] **A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM review*, vol. 51, pp. 34–81, 2009.**

[60] **R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, pp. 267-288, 1996.**

[61] **S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, pp. 2346-2356, 2008.**

[62] **H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, pp. 301-320, 2005.**

[63] **A. N. Tikhonov and V. I. A. Arsenin, *Solutions of ill-posed problems*: Winston Washington, DC:, 1977.**

[64] **J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, p. 1, 2010.**

[65] **M. Aharon, M. Elad, and A. Bruckstein, "K SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Transactions on Signal Processing*, vol. 54, pp. 4311-4322, 2006.**

[66] **M. D. Plumbley, "Dictionary learning for l1-exact sparse coding," in *International Conference on Independent Component Analysis and Signal Separation*, 2007, pp. 406-413.**

[67] **J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," *Advances in Neural Information Processing Systems*, vol. 21, pp. 1033-1040, 2009b.**

[68] **D. Vainsencher, S. Mannor, and A. M. Bruckstein, "The Sample Complexity of Dictionary Learning," *Arxiv preprint arXiv:1011.5395*, 2010.**

[69] **F. Sedlák, T. Kinnunen, V. Hautamäki, K. A. Lee, and H. Li, "Classifier subset selection and fusion for speaker verification," in *Proc. of ICASSP*, 2011, pp. 4544 - 4547.**

[70] **N. Brummer and J. du Preez, "Application-independent evaluation of speaker detection," *Computer Speech & Language*, vol. 20, pp. 230-275, 2006.**

[71] **A. F. Martin and C. S. Greenberg, "The nist 2010 speaker recognition evaluation," 2010.**

[72] **N. Brummer, L. Burget, and P. Kenny, "ABC system description for NIST SRE 2010," *Proc. NIST 2010 Speaker Recognition Evaluation*, 2010.**

[73] **J. Villalba, C. Vaquero, E. Lleida, and A. Ortega, "I3A NIST SRE2010 System Description."**