Available online at www.sciencedirect.com

ScienceDirect

Computer Speech & Language 47 (2018) 333–350

www.elsevier.com/locate/csl

Sparse coding based features for speech units classification☆

Pulkit Sharma*, Vinayak Abrol, A.D. Dileep, Anil Kumar Sao

School of Computing and Electrical Engineering, Indian Institute of Technology, Mandi, India

Received 26 May 2016; received in revised form 15 June 2017; accepted 17 August 2017
Available online 1 September 2017

Abstract

In this work, we propose sparse representation based features for speech units classification tasks. In order to effectively capture the variations in a speech unit, the proposed method employs multiple class specific dictionaries. Here, the training data belonging to each class is clustered into multiple clusters, and a principal component analysis (PCA) based dictionary is learnt for each cluster. It has been observed that coefficients corresponding to middle principal components can effectively discriminate among different speech units. Exploiting this observation, we propose to use a transformation function known as weighted decomposition (WD) of principal components, which is used to emphasize the discriminative information present in the PCA-based dictionary. In this paper, both raw speech samples and mel frequency cepstral coefficients (MFCC) are used as an initial representation for feature extraction. For comparison, various popular dictionary learning techniques such as K-singular value decomposition (KSVD), simultaneous codeword optimization (SimCO) and greedy adaptive dictionary (GAD) are also employed in the proposed framework. The effectiveness of the proposed features is demonstrated using continuous density hidden Markov model (CDHMM) based classifiers for (i) classification of isolated utterances of E-set of English alphabet, (ii) classification of consonant-vowel (CV) segments in Hindi language and (iii) classification of phoneme from TIMIT phonetic corpus.

Keywords: Sparse representation; Dictionary learning; Speech recognition

1. Introduction

Feature extraction is an important step for speech recognition, as it involves conversion of the speech signal into a sequence of acoustic features (frame by frame basis) (Rabiner and Schafer, 2010). Here, the extracted features should capture characteristics contributing to the phonetic differences among different speech units, so as to enable discrimination among them. Available features for the tasks in speech recognition are either motivated by speech production or by speech perception mechanisms. Features belonging to the former category are also known as articulatory based features (Saenko et al., 2009; Mitra et al., 2011), and have been demonstrated to give good speech recognition performance, but their estimation from speech signal is difficult (Mitra et al., 2011). Hence, features based on speech perception, such as mel frequency cepstral coefficients (MFCC) (Davis and Mermelstein, 1980),

☆ This paper has been recommended for acceptance by L. ten Bosch.

* Corresponding author.
E-mail address: pulkit_s@students.iitmandi.ac.in (P. Sharma), vinayak_abrol@students.iitmandi.ac.in (V. Abrol), addileep@iitmandi.ac.in (A.D. Dileep), anil@iitmandi.ac.in (A.K. Sao).

http://dx.doi.org/10.1016/j.csl.2017.08.004

perceptual linear predictive (PLP) coefficients (Hermansky, 1990) etc. are very popular for the tasks in speech recognition. Further, few approaches have performed transformations, e.g., neural networks based transformations (Hermansky et al., 2000) and feature-space minimum phone error (fMPE) (Povey et al., 2005), on the standard perception based features to improve the performance of speech recognition systems. Mitra et al. (2011) have demonstrated that MFCC in conjunction with articulatory representations provide better results as compared to individual features, at the expense of increased computational complexity.

Features for speech recognition should highlight the discriminative information among the speech units. Although the speech signal corresponds to a high dimensional data captured using sensors i.e., microphone (Tosic and Frossard, 2011), the number of generating causes is very small as compared to recorded observations. Thus, the information relevant to the underlying process of generating signal (speech in our case) is generally of reduced dimensionality as compared to the recorded observations (Tosic and Frossard, 2011), and can be exploited for estimating efficient representations of the speech signal. Such representations can be achieved either using a compact code or sparse code (Shashanka et al., 2007). In general, compact code ($\mathbf{x}_c \in \mathbb{R}^{N_1}$) for any signal $\mathbf{x} \in \mathbb{R}^N$, is of fewer dimension i.e., $N_1 < N$. On the contrary, the number of elements in sparse distributed code ($\boldsymbol{\alpha} \in \mathbb{R}^N$), are equal to the number of elements in the input, but most of those elements are zero, i.e., only K ($K \ll N$) elements are needed to represent the given input faithfully. However, in sparse code, location of K significant coefficients representing the generating causes may vary for different speech units (Shashanka et al., 2007). This helps in capturing the distinct causes responsible for different speech units. This work is focused on demonstrating the use of sparse code in capturing discriminative information for speech units classification.

In recent years sparse coding based signal processing has been applied to various speech processing applications such as audio classification (Zubair et al., 2013), speaker verification (Haris and Sinha, 2012), speech enhancement (Abrol et al., 2013; Low et al., 2013), speech recognition (Sivaram et al., 2010), speech separation (Xu et al., 2013), speaker tracking (Barnard et al., 2014) and speech coding (Giacobello et al., 2010). In sparse representations (SR) of speech signal, a speech frame is written as a linear combination of atoms of a resource, known as dictionary. The sparse vector obtained for each speech frame, given a dictionary, is used as a feature. The behavior of sparse vector is very much influenced by the choice of dictionary, which could be either analytical or learnt. Analytical dictionaries are easy to implement and have fast transform properties. The learnt dictionaries are derived from the data itself and thus adapt to the variations in data effectively (Tosic and Frossard, 2011).

In this work, novel SR of speech signal, computed using principal component analysis (PCA) based dictionary, is proposed for tasks in speech recognition. The approach presented in this work is similar to the one proposed in Dong et al. (2011), in the context of images. In order to capture the variations present in the speech signal, we have shown that it is preferable to use multiple dictionaries. This is done by first clustering speech frames corresponding to a speech unit into Q different clusters. Eigenvalues and corresponding eigenvectors are computed for each cluster. All the eigenvectors (in decreasing order of eigenvalues) are arranged column-wise to construct dictionary (specifically sub-dictionary) for a cluster. It is studied in the literature of image processing, that the eigenvectors corresponding to the largest eigenvalues give information common to all the training samples, and the least significant information is present in the eigenvectors corresponding to small eigenvalues (Shejin and Sao, 2012; O'Toole et al., 1994, 1997). The eigenvectors corresponding to intermediate principal directions include the discriminative information among the training examples and are demonstrated in the context of face recognition. We have proposed to use a transformation function known as weighted decomposition (WD) (Shejin and Sao, 2012) to suppress the most and least significant components in the proposed PCA-based dictionary.

In the proposed approach, dictionaries belonging to all the clusters along with their centroids are stored for each speech unit (class). For any speech frame, a minimum distance criteria is used to select a suitable dictionary and then the corresponding sparse vector is used as a feature. The extracted features are employed in a continuous density hidden Markov model (CDHMM) based classifier to demonstrate the usefulness of these features for speech units classification tasks. As a comparison, we have also explored K-singular value decomposition (KSVD) dictionary (Aharon et al., 2006), simultaneous codeword optimization (SimCO) dictionary (Dai et al., 2012) and greedy adaptive dictionary (GAD) (Jafari and Plumbley, 2011) to obtain SR for a speech unit in the proposed approach.

This paper is an extension of our existing work published in Sharma et al. (2015), and the contributions of this work are: (i) PCA-based multiple dictionaries to extract SR based feature from speech signals, (ii) emphasizing discriminative information using a dictionary based on weighted decomposition of principal components (PCs)

(iii) sparse nature of proposed feature vector is exploited for noise robustness, (iv) obtaining SR from raw speech samples, and (v) extensive experimentation using three datasets for the tasks in speech recognition.

The organization of the paper is as follows: Section 2 describes sparse coding for speech signals and various SR based approaches for speech recognition. The proposed dictionary learning technique and feature extraction method is explained in Section 3. Detailed description about the databases used in this work is given in Section 4. Experimental results are presented in Section 5, and the work is concluded in Section 6.

2. Sparse coding for speech signals

The basic idea in SR based signal processing is supported by an observation that signal has a sparse representation with respect to a suitable dictionary. Assuming that a speech frame $\mathbf{s} \in \mathbb{R}^N$ is represented using a dictionary $\boldsymbol{\Psi} \in \mathbb{R}^{N \times N}$ as $\mathbf{s} = \boldsymbol{\Psi}\boldsymbol{\alpha}$, such that $\boldsymbol{\alpha} \in \mathbb{R}^N$ is K ($K \ll N$) sparse, i.e., $\boldsymbol{\alpha}$ has only K significant coefficient. Given $\mathbf{s}$ and $\boldsymbol{\Psi}$, the estimate of sparse vector $\boldsymbol{\alpha}$ can be obtained as

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}}\ f(\boldsymbol{\alpha}) \quad \text{s.t.} \quad \| \mathbf{s} - \boldsymbol{\Psi}\boldsymbol{\alpha} \|_2^2 < \epsilon, \tag{1}$$

where ϵ is a constant known as tolerance error, and $f(.)$ is used to promote sparsity and can be l_0 or l_1-norm (Elad, 2010). The l_0-norm counts the number of non zero elements in $\boldsymbol{\alpha}$. The computation of l_0-norm is a combinatorial search and is a NP hard problem (Donoho and Stark, 1989). Hence, a greedy approach known as orthogonal matching pursuit (OMP) is used to solve l_0-norm in this work (Tropp and Gilbert, 2007). Alternatively convex relaxations to l_0-norm e.g., l_1-norm can also be used.[1] An estimate of $\widehat{\boldsymbol{\alpha}}$ obtained using equation (1) is used to estimate speech signal as $\widehat{\mathbf{s}} = \boldsymbol{\Psi}\widehat{\boldsymbol{\alpha}}$.

Most of the existing SR based features in speech recognition use estimate of speech signal ($\widehat{\mathbf{s}}$) as a feature for acoustic modeling (Sainath et al., 2010a, 2010b, 2011). On the contrary, method proposed in Sivaram et al. (2010) uses estimate of sparse vector ($\widehat{\boldsymbol{\alpha}}$) as a feature. Similar to Sivaram et al. (2010), the method proposed in this paper also employs the estimated sparse vector as a feature for speech units classification. Approaches for SR based speech recognition can be categorized into two categories: (1) exemplar based approaches, and (2) feature based approaches.

2.1. Exemplar based approaches

In exemplar based approaches, a test speech signal is modeled as superposition of training speech exemplars (Gemmeke et al., 2011; Yilmaz et al., 2014; Baby et al., 2015), where the dictionary atoms are speech exemplars from the training set.[2] Approaches in Gemmeke et al. (2011) and Baby et al. (2015) uses a single overcomplete dictionary with spectro-temporal representations of speech (labeled at frame level) as exemplars. On the contrary, Yilmaz et al. (2014) use multiple dictionaries (including exemplars of same speech unit) corresponding to different speech units.

During testing, for a given exemplar $\mathbf{y} \in \mathbb{R}^N$, the sparse vector is obtained using a dictionary $\mathbf{D} \in \mathbb{R}^{N \times C}$ ($N < C$), where C is the total number of atoms in the dictionary. The estimated vector will have larger values for the coefficients corresponding to the atoms of true class as compared to rest of the atoms, and hence Gemmeke et al. (2011) and Baby et al. (2015) uses these atom activations for speech recognition. On the other hand, in Yilmaz et al. (2014) speech recognition is performed by finding the class sequence yielding the minimum reconstruction error between the test exemplar and its estimate.

For real time speech recognition using exemplar based approaches, the speech data is labeled using a dynamic programming approach (e.g., hidden Markov models (HMM)) to obtain the class boundaries first. Any errors in labeling of class boundaries leads to poor performance of speech recognition system. In addition, the obtained dictionaries are highly overcomplete as the number of atoms are more than the dimensionality of exemplar i.e., $N \ll C$. Thus, the search space for efficient sparse solution is very large, which in general leads to an unstable solution and hence increases the chances of misclassification (Elad, 2010).

[1] The l_1-norm, also known as Manhattan norm of a vector $\boldsymbol{\alpha}$ is defined as $\| \boldsymbol{\alpha} \|_1 = \sum_i |\boldsymbol{\alpha}_i|$ (Elad, 2010).

[2] For exemplar based approaches, speech exemplars are spectrographic representations of speech spanning over multiple speech frames (Gemmeke et al., 2011).

2.2. Feature based approaches

In feature based approaches, sparse coding is used to derive features for speech recognition. Here, either the sparse vector (Sivaram et al., 2010) or the estimate of speech signal (obtained using the sparse vector) (Sainath et al., 2010a, 2010b, 2011) is used as the feature representation for acoustic modeling. Existing approaches employ both single overcomplete (Sivaram et al., 2010) and multiple dictionaries (Sainath et al., 2010a, 2010b, 2011) to derive these features. Sivaram et al. (2010) used a spectro-temporal representation to learn a single overcomplete dictionary using a gradient descent approach. Sainath et al. (2010a, 2010b, 2011) used various methods e.g., the nearest neighbors and trigram language model based methods to select dictionary atoms. Since different dictionary atoms (from the training data) are selected for different speech frames in Sainath et al. (2010b, 2011), the computational complexity of resulting feature extraction method is high. In addition, here all the training samples are required to be stored which increases the memory requirements also.

However, the method proposed in this work require only the learnt dictionaries along with corresponding centroids to be stored, thus reducing the required storage space. In addition, the proposed method uses raw speech samples to derive SR based features as compared MFCC and spectro-temporal patterns employed in the literature.

The major differences of the proposed approach with respect to the existing approaches are as follows: (i) the proposed approach uses multiple dictionaries as opposed to a single overcomplete dictionary in Sivaram et al. (2010), (ii) the proposed feature extraction method is computationally efficient, as it does not uses any initial representation like spectro-temporal (Sivaram et al., 2010) or MFCC (Sainath et al., 2010b, 2011), and (iii) the proposed method employs a WD-PCA based dictionary to emphasizes the discriminative information among speech units. Detailed description of the approach proposed to obtain the sparse feature for speech signal is presented in the next section.

3. Proposed approach for dictionary learning and sparse features for speech signal

The proposed approach uses sparse coding framework to derive features (from speech signal) for the tasks in speech recognition. Here, the sparse representation obtained using a given dictionary is used as a feature for speech units classification. Dictionary ($\boldsymbol{\Psi}$) used to estimate sparse feature can be either analytical like discrete cosine transform (DCT), wavelet, discrete Fourier transform (DFT) etc. or it can be learnt from speech frames of training set e.g., KSVD dictionary (Aharon et al., 2006) , GAD (Jafari and Plumbley, 2011) etc. In this work, we propose to use PCA-based dictionary to obtain sparse features for speech units classification.

Consider n_i training speech frames from a speech unit/class $i, \{\mathbf{s}_{ij}\}_{j=1}^{n_i}$, arranged in a data matrix $\mathbf{S}_i \in \mathbb{R}^{N\times n_i}$ as columns such that $\mathbf{S}_i = [\mathbf{s}_{i1}, \mathbf{s}_{i2}, \ldots, \mathbf{s}_{in_i}]$. In this work, speech unit/class represent CV unit or phoneme, and are used interchangeably. In order to learn multiple dictionaries, these frames are first clustered into Q clusters using K-means clustering algorithm (Duda et al., 2001), and data matrix corresponding to qth cluster is denoted by $\mathbf{S}_{iq} \in \mathbb{R}^{N\times m_{iq}}$, $q = 1, 2, \ldots, Q$. Here m_{iq} denotes total number of frames assigned to cluster q (with centroid $\boldsymbol{\mu}_{iq}$).

A dictionary $\boldsymbol{\Psi}_{iq}, \{q = 1, 2, \ldots, Q\}$ corresponding to each cluster $\mathbf{S}_{iq}$ is learnt such that the representation over this dictionary is as sparse as possible. $\boldsymbol{\Psi}_{iq}$ can be learnt using the following objective function:

$$\left(\widehat{\boldsymbol{\Psi}}_{iq}, \widehat{\boldsymbol{\Lambda}}_{iq}\right) = \underset{\boldsymbol{\Psi}_{iq},\ \boldsymbol{\Lambda}_{iq}}{\operatorname{argmin}} f_1(\boldsymbol{\Lambda}_{iq}) \ \text{ s.t. } \ \| \mathbf{S}_{iq} - \boldsymbol{\Psi}_{iq}\boldsymbol{\Lambda}_{iq} \|_F^2 < \epsilon, \tag{2}$$

where $\|.\|_F$ is the Frobenius norm and $f_1(.)$ defines how sparseness is measured over columns of $\boldsymbol{\Lambda}_{iq}$, which is the representation coefficient matrix of $\mathbf{S}_{iq}$ over sub-dictionary $\boldsymbol{\Psi}_{iq}$. Eq. (2) is a joint optimization problem of solving $\boldsymbol{\Psi}_{iq}$ and $\boldsymbol{\Lambda}_{iq}$, which can be solved by alternatively optimizing $\boldsymbol{\Psi}_{iq}$ and $\boldsymbol{\Lambda}_{iq}$ (Dong et al., 2011). The joint minimization of Eq. (2) is computationally expensive and is generally used to learn an overcomplete dictionary. A computationally efficient method, proposed in this work, to learn dictionary for each cluster is discussed in the Section 3.1.

Fig. 1 illustrates the proposed approach of dictionary learning and selection for *ith* speech unit using two dimensional representation of all the frames available in training speech signals. Symbolically six clusters are shown in the Fig. 1 (a). $\boldsymbol{\mu}_{iq}$ is the mean vector corresponding to cluster q. The process of dictionary selection for a speech signal is shown in Fig. 1 (b). For the illustration, speech signal $\mathbf{s}_i$ is divided into 8 frames $\mathbf{s}_{i1}, \mathbf{s}_{i2}, \ldots, \mathbf{s}_{i8}$ as shown in Fig. 1 (b)[3].

[3] Extracted frames, illustrated in Fig. 1, of speech signal are non-overlapping, but in our experiments overlapping frames are used.

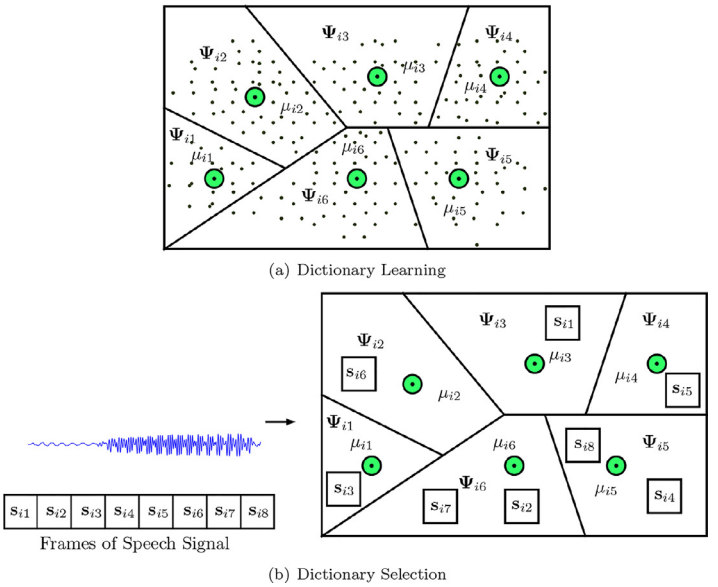

Fig. 1. Illustration of proposed (a) dictionary learning approach, and (b) dictionary selection method, for a speech unit/class i. The waveform shown here correspond to speech unit */ba/*.

A suitable dictionary, chosen based on the minimum distance from the cluster centers with given speech frame, is used to compute the sparse vector for each frame, which is then used as a feature for tasks in speech recognition.

3.1. Adaptive dictionary for speech signals

In order to effectively discriminate among different speech units, the learnt dictionary should be able to capture the discriminative information present in different speech units. PCA-based dictionaries are one of the dictionaries known to capture the discriminative information (among different classes) well (Shejin and Sao, 2012) and hence are employed in this work. Our approach of dictionary learning is explained in Procedure 1, and builds upon the idea of Dong et al. (2011) in the context of image super resolution. However, in this work dictionaries are learnt for the classification task, as opposed to the reconstruction (super-resolution) task in Dong et al. (2011). Here, class dependent multiple dictionaries are learnt, as compared to class independent multiple dictionaries learnt in Dong et al. (2011). In addition, the proposed method also employs weighted decomposition of principal components to emphasize the discriminative information among confusing speech units, which will be explained in Section 3.1.1.

Procedure 1. Proposed method for dictionary learning.

Inputs: Matrix $\mathbf{S}_i = [\mathbf{s}_{i1}, \mathbf{s}_{i2}, \ldots, \mathbf{s}_{in_i}]$ with speech frames from class i as columns.
Outputs: Q dictionary pairs $\{\boldsymbol{\Psi}_{iq}, \boldsymbol{\mu}_{iq}\}$.

1: Cluster $\mathbf{S}_i$ into Q clusters using K - means clustering algorithm with mean $\boldsymbol{\mu}_{iq}$ being the center of q^{th} cluster.
2: $\mathbf{S}_{iq}$ represents data belonging to q^{th} cluster.
3: Dictionary $\boldsymbol{\Psi}_{iq}$ can be obtained after applying PCA to $\mathbf{S}_{iq}$.
$\boldsymbol{\Psi}_{iq} = [\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, ..., \boldsymbol{\psi}_N]$, where $\boldsymbol{\psi}_e$ corresponds to e^{th} eigenvector.
4: Repeat above steps to obtain dictionaries for all the M speech units (classes).
5: Save all the dictionaries $\boldsymbol{\Psi}_{iq}$ for all the speech unit along with centroid $\boldsymbol{\mu}_{iq}$ corresponding to each cluster $\mathbf{S}_{iq}$.

Repeat above steps for M classes to get QM pairs
$\{\boldsymbol{\Psi}_{iq}, \boldsymbol{\mu}_{iq}\}$.

For a given dictionary, the representation can be obtained using linear approximation (LA) or nonlinear approximation (NLA) methods (Sezer et al., 2015). Let us denote the dictionary corresponding to *qth* cluster using $\boldsymbol{\Psi}_{iq} = [\boldsymbol{\psi_1}, \boldsymbol{\psi_2}, \ldots, \boldsymbol{\psi_N}]$, where *eth* column $\boldsymbol{\psi}_e$ is *eth* principal component of $\mathbf{S}_{iq}$. It has to be noted that $[\boldsymbol{\psi_1}, \boldsymbol{\psi_2}, \ldots, \boldsymbol{\psi_N}]$ are arranged in decreasing order of corresponding eigenvalues. This is done for the convenience purpose and the order of PCs in dictionary will not affect the results. The LA of $\mathbf{s} \in \mathbb{R}^{N\times 1}$ using dictionary $\boldsymbol{\Psi}_{iq} \in \mathbb{R}^{N\times N}$ with *eth* column $\boldsymbol{\psi}_e \in \mathbb{R}^N$ such that $\boldsymbol{\psi}_e^T \boldsymbol{\psi}_e = 1$ is defined as:

$$\widehat{\mathbf{s}}_L(\boldsymbol{\Psi}_{iq}, n) = \sum_{e=1}^{n} \boldsymbol{\psi}_e \widehat{\boldsymbol{\alpha}}_e, \tag{3}$$

where $1 \leq n \leq N$, and

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \parallel s - \sum_{e=1}^{n} \boldsymbol{\psi}_e \boldsymbol{\alpha}_e \parallel^2, \tag{4}$$

$\widehat{\boldsymbol{\alpha}} = [\widehat{\boldsymbol{\alpha}}_1, \widehat{\boldsymbol{\alpha}}_2, \ldots, \widehat{\boldsymbol{\alpha}}_n]^T$, $\widehat{\boldsymbol{\alpha}}_i$ is the coefficient associated with basis vector $\boldsymbol{\psi}_i$. On the contrary, the NLA of $\mathbf{s}$ with $\boldsymbol{\Psi}_{iq}$ is defined as:

$$\widehat{\mathbf{s}}_N(\boldsymbol{\Psi}_{iq}, n) = \boldsymbol{\Psi}_{iq} \widehat{\boldsymbol{\alpha}}, \quad \text{s.t.} \quad \parallel \widehat{\boldsymbol{\alpha}} \parallel_0 = n, \tag{5}$$

where $\|.\|_0$ denotes the ℓ_0-norm, and the weight vector is estimated as:

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \parallel s - \boldsymbol{\Psi}_{iq} \boldsymbol{\alpha} \parallel^2 \ \text{s.t.} \ \parallel \boldsymbol{\alpha} \parallel_0 = n. \tag{6}$$

In LA, the approximating coefficient ($\widehat{\boldsymbol{\alpha}}$) is computed by minimizing the error between signal and its estimate using first n dictionary atoms (PCs). On the contrary, in NLA, the approximation coefficients are computed using an additional constraint of minimizing the l_0-norm, i.e., $\parallel \boldsymbol{\alpha} \parallel_0 = n$. Under the assumption that the speech frames under analysis are generated from a Gaussian process, for approximation using n PCs as dictionary, NLA performs better as compared to LA (Sezer et al., 2015).

The objective of the proposed work is to compute the discriminative feature from a given speech frame. Therefore, we have chosen to use the NLA using the selected PCs for a given speech frame. In the selected PCs first few and last few PCs are not included as they do not contain much discriminative information for pattern recognition tasks, where patterns have subtle minute differences. For example, in face recognition, all the face images have one nose, two eyes but the difference lies in the subtle variations, emphasized using selected middle PCs (Shejin and Sao, 2012). This analogy holds true for the sound units such as */ba/* and */bA/*, where subtle variations between the units has to be emphasized for better discrimination.

3.1.1. Significance of principal components in the proposed dictionary

In order to demonstrate the significance of principal components (PCs) in the dictionary to emphasize the discriminative information among the speech units, the following experiments were performed. Speech frames of 25 ms duration with 10 ms overlap are extracted from all the examples of speech units, */ba/* provided in the Hindi CV segments (Dileep and Sekhar, 2013). It has to be noted that speech signal is sampled at 16 kHz, thus each frame of speech signal can be seen as a point in 400-dimension space. The frames, extracted from all the examples of */ba/* are divided into five cluster (the reason for five cluster will be explained later) in 400 dimensions and PCs are computed for each cluster. The distribution of frames can be visualized in 2-dimensions (2-d) using t-Distributed Stochastic Neighbor Embedding (t-SNE) visualizations (van der Maaten and Hinton, 2008) and is shown in Fig. 2(a). We have also marked the mean of the five clusters, estimated using *K*-means algorithm, in the same figure using numeral digits. The above mentioned experiments were repeated with the speech signals of two more sound units namely: */bA/* and */no/* and plots are shown in Fig. 2(b) and (c), respectively. It can be observed that the distribution of speech frames of sound units */ba/* and */bA/* are more overlapped as compared to the frames of sound unit of */no/*. Reason for this could be the similarity in production mechanism for */ba/* and */bA/* as compared to */no/* (Rabiner and Schafer, 2010). Please note that numeral representation of clusters has no order, it is just a symbol to a cluster.

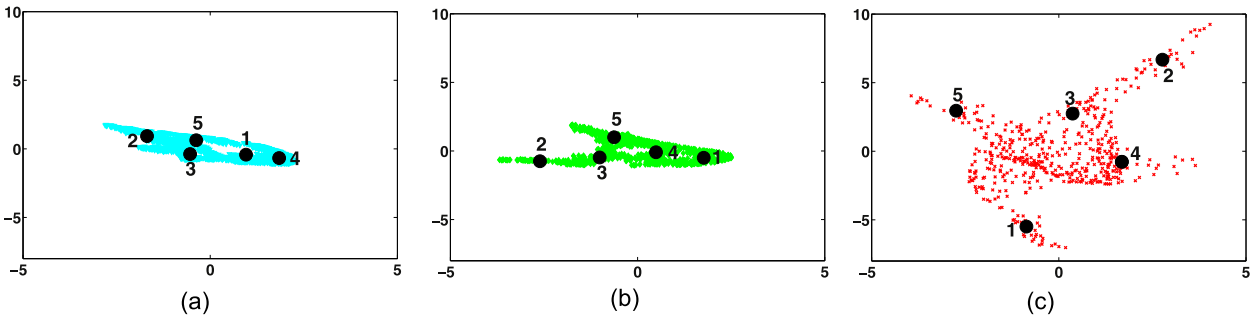

Fig. 2. Two-dimensional t-SNE visualization of data along with cluster centroids for (a) /ba/, (b) /bA/ and (c) /no/ class. These visualizations are corresponding to 400-dimensional raw speech samples.

In order to investigate further, we have measured similarity among pairs of temporal sequences $/ba/ - /bA/$ and $/ba/ - /no/$ using dynamic time warping (DTW) algorithm (Rath and Manmatha, 2003) as shown in Fig. 3. Ideally, DTW path should be along diagonal if the two sequences are from same sound units and will go away from the diagonal path for the different sound units. Here /ba/ is considered as reference template and representation of each frame is computed using selected PCs in the dictionary learnt from the cluster, nearby to the respective frames. We have also marked (see in bracket) the cluster number, whose PCs are employed to compute the representation, for each frame of the two speech signals. In this experiment, five dictionaries learnt on training data of /ba/ are used to derive feature representations for examples of different classes. Since the space spanned by dictionaries learnt for individual classes may overlap, initially we analyze the behavior of the SR obtained using dictionaries of only one class. However, the behavior obtained using a single dictionary will be magnified using class specific dictionaries. It can be observed from Fig. 3(a) that the sequence of clusters is approximately same for the pair $/ba/ - /bA/$, as their acoustic characteristic are similar with only subtle differences. These differences are emphasized if the representation is computed using only selected PCs (10–120) as shown in the figure. The reason could be that first few PCs give average information of the clusters and last few PCs contain least significant information of the cluster. Hence, the representation vector corresponding to middle PCs (10–120) help in emphasizing subtle discriminative information of two similar sound units. In contrast, for the pair $/ba/ - /no/$ from distinct classes, top coefficients (1–10) result in more deviation in the DTW path, as shown in Fig. 3(b). Although not the best, middle PC still achieve reasonable deviation in the DTW path for distinct speech units. Thus, an attempts is made to emphasize middle coefficients for classification among speech sounds (distinct as well as confusing).

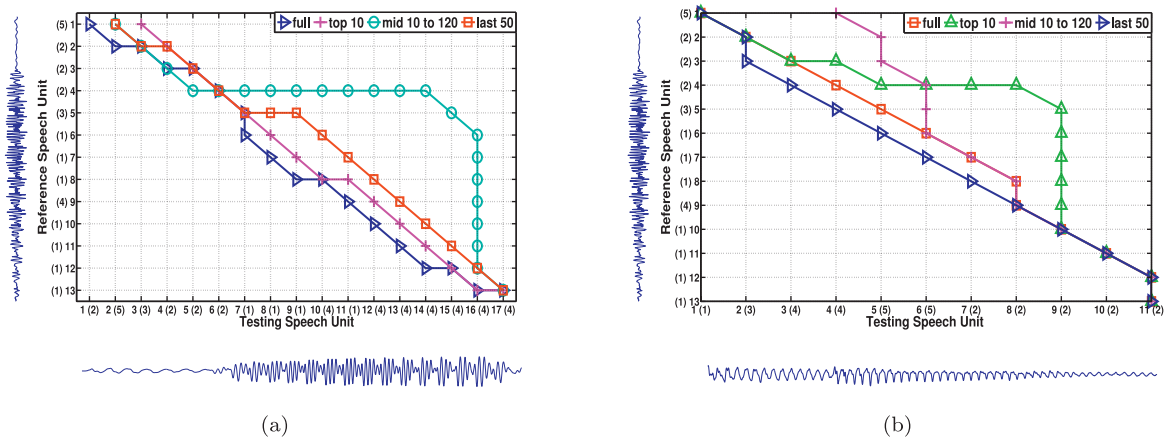

Fig. 3. DTW path for sparse vector obtained using selected principal components in the dictionary for two speech units. (a) confusing classes /ba/ vs /bA/ and (b) different classes /ba/ vs /no/ when 5 dictionaries learnt on /ba/ data are used to derive feature vectors for all the classes. Numbers in () on both the axis indicate the cluster index corresponding to dictionary used for that particular speech frame.

Table 1
Average difference in DTW distance (in %) of sparse vector obtained using all PCs in dictionary with sparse vector obtained using selected number of PCs in the dictionary.

Hindi CV segments		PCs used		
		Top 10	Middle 10 − 120	Last 50
Confusing classes	/ba/ **vs** /bA/	−2.31	**−8.07**	4.59
	/ri/ **vs** /rI/	−1.97	**−7.38**	−2.76
	/ve/ **vs** /vi/	−1.83	**−7.52**	−1.04

The observations for confusing speech units are statistically verified by calculating the percentage difference of DTW distances for different choices of PCs in the dictionary from the DTW distance obtained using all the PCs in the dictionary. The averaged percentage difference in DTW distance for three sets of confusing classes, having 200 different speech utterance pairs in each set, is shown in Table 1. Negative/positive percentage in this table means the corresponding DTW distances are increasing/decreasing. The negative percentage means that the resultant sparse/feature vector is more discriminative.λ

Nonlinear approximations over dictionary consisting of middle PCs i.e., $\boldsymbol{\Psi}_{iq_p} = [\boldsymbol{\psi}_l, \ldots \boldsymbol{\psi}_u]$ can be performed to emphasize the discrimination among the confusing classes. However, it is difficult to come up with the best choice of start (l) and end (u) of middle PCs. In addition, this will result in decreased discrimination among distinct classes. This issue is addressed by modifying the dictionary as:

$$\boldsymbol{\Psi}^t_{iq} = \boldsymbol{\Psi}_{iq}\mathbf{W}_{iq}, \tag{7}$$

where $\mathbf{W}_{iq}$ is a diagonal weight matrix with $\left\{\frac{1}{\sqrt{\lambda_i q1}}, \frac{1}{\sqrt{\lambda_i q2}}, \ldots, \frac{1}{\sqrt{\lambda_i qN}}\right\}$ as diagonal elements and $\lambda_{iq1}, \lambda_{iq2}, \ldots, \lambda_{iqN}$ are the eigenvalues (magnitude only) corresponding to the eigenvectors. $\boldsymbol{\Psi}_{iq}$ is the dictionary matrix consisting of PCs as its columns. The transformed dictionary obtained from Eq. (7) is denoted as WD of PCs. The WD transformation will allow the scaling of each principal component with the corresponding eigenvalues and hence results in de-emphasizing the most significant PCs (corresponding to the largest eigenvalues) and emphasizing the middle and last PCs.

However, the last PCs does not contain any discriminative information and should be discarded. The estimation of the number of PCs required for efficient discrimination is important. The rank of the covariance matrix, estimated for each cluster, could be used to decide the effective number of PCs needed to be kept in the dictionary. However, the computation of rank adds an extra computational burden and may vary over different clusters. Thus, all the PCs, computed using covariance method are used as dictionary atoms. After experimentation, it has been observed that the least $N/2$ eigenvalues of covariance matrix are very small (close to zero), when raw speech samples are used as initial representation of speech. The inverse of least $N/2$ eigenvalues will be vary large, and thus the WD transformation is over-emphasizing the respective PCs. Hence, to prevent over-emphasis of least $N/2$ PCs, these PCs are scaled with a very small constant. Thus, it can be concluded that transformed dictionary obtained from Eq. (7) denoted as WD followed by scaling of least significant PCs, emphasizes the discriminative information present in the middle PCs.

3.2. Estimation of feature vector

After dictionary learning, we have Q dictroids (dictionary + centroid) pairs $\{\boldsymbol{\Psi}^t_{iq}, \boldsymbol{\mu}_{iq}\}$ for each speech unit. For a new speech frame $\mathbf{s}$, the best fitted dictionary is selected based on the following criterion:

$$q^* = \underset{q}{\operatorname{argmin}} \ \| \mathbf{s} - \boldsymbol{\mu}_{iq} \|_2 \tag{8}$$

The corresponding sub-dictionary $\boldsymbol{\Psi}^t_{iq^*}$ is used to solve the following optimization problem to obtain estimate of sparse vector $\boldsymbol{\alpha}$ corresponding to each speech frame $\mathbf{s}$

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \ \| \mathbf{r} - \boldsymbol{\Psi}^t_{iq^*}\boldsymbol{\alpha} \|_2^2 \quad \text{s.t.} \quad \| \boldsymbol{\alpha} \|_0 \leq k, \tag{9}$$

where $\mathbf{r} = \mathbf{s} - \boldsymbol{\mu}_{iq^*}$ is used to normalize the speech data. Sparse vector $\widehat{\boldsymbol{\alpha}}$ obtained after solving Eq. (9) (for different frames **s**) is used as a feature for tasks in speech recognition. During training, the labeled data is available and hence the class index i in Eqs. (8) and (9) is available. During testing, for a given test signal the Eqs. (8) and (9) are solved for all i, and the corresponding features are presented to the respective CDHMM. The test speech signal is then assigned to a class with maximum likelihood.

As a comparison, the performance of the proposed approach is also evaluated using recently proposed dictionaries such as KSVD, SimCO and GAD. KSVD, SimCO and GAD gives a sparser representation than PCA, but does not decorrelate training data as in case of PCA. The learnt PCA-based dictionary is always complete, but the KSVD dictionary, SimCO dictionary and GAD can be both complete and overcomplete. Overcomplete dictionaries are used to capture most of the variations in the training data, and thus obtain efficient SR of unseen data. In the proposed approach multiple dictionaries are learnt for each class by clustering the data into different clusters, thus reducing the need of a single overcomplete (highly) dictionary. In order to have a fair comparison, the same approach is followed i.e., training frames from a class i are first clustered into Q clusters and then for each cluster KSVD, GAD and SimCO dictionaries are learnt.

4. Experimental setup

The proposed features are used to build the CDHMM based classifier for different tasks in speech recognition and results are demonstrated using classification of (i) isolated utterance of E-set of English alphabet, (ii) consonant-vowel (CV) segments in Hindi, and (iii) phonemes in TIMIT phonetic corpus.

4.1. Database for E-set classification

In the study on classification of E-set, the Oregon Graduate Institute (OGI) spoken letter database is used (ISOLET Corpus, 2010). E-set consists of the highly confusing subset of spoken letters in English alphabet. The E-set database used in this work (ISOLET Corpus, 2010) includes the 9 letters: B, C, D, E, G, P, T, V, and Z. Here the training data set consists of 240 utterances for each letter from 120 speakers, and the test data set consists of 60 utterances for each letter from 30 speakers. Thus, this data set has a total of 2160 training examples and a total of 540 test examples. The E-set classification accuracy (CA) obtained for 540 test examples is reported in classification results.

4.2. Database for CV segments classification

The experiments on CV segments classification are performed on continuous speech corpus of broadcast news in Hindi (Dileep and Sekhar, 2013). We considered total 103 CV classes with minimum 50 examples in the training data set. A total of 19,458 CV segments are there in the training data set and 4866 CV segments are there in test data set. The CV segment CA presented is the accuracy along with 95% confidence interval obtained for 5-fold stratified cross-validation.

4.3. Database for phoneme classification

TIMIT phonetic corpus (Garofolo et al., 1993) contains 61 phoneme classes of English. In accordance with standard experimentation on TIMIT, the 61 phonetic labels are mapped to a standard set of 48 phonemes for acoustic model training and decoding (Lee and Hon, 1989). The TIMIT training set consists of utterances of speech from 375 speakers while testing set consists of utterances of 87 speakers. A total of 171,531 phoneme segments are there in the training data set and 62,605 phoneme segments are there in test data set. The phoneme CA obtained for 62,605 test segments is reported in classification results.

4.4. Initial representation for a speech frame

Speech signal used for experimentation is sampled at a rate of 16 kHz and features are extracted at a frame size of 25 ms with an overlap of 10 ms. We consider both 400-dimensional raw speech sample and 39-dimensional MFCC

as two different initial representations for a speech frame. For MFCC feature vector, first 12 features are MFCC and the 13*th* feature is log energy. The remaining 26 features are the delta and acceleration coefficients. These initial representations are used to learn dictionaries for each speech unit.

The proposed sparse feature vector is labeled as $\mathbf{f}_{\{b\}}^{\{a\}}$ where

- {a} is either
  1. R when the sparse feature is derived from raw speech samples, or
  2. M when the sparse feature is derived from MFCC as initial representation for a speech frame.
- {b} is
  1. KSVD : when the sparse feature is derived from KSVD dictionary.
  2. PCA : when the sparse feature is derived from PCA-based dictionary.
  3. GAD : when the sparse feature is derived from GAD.
  4. WD-PCA : when the sparse feature is derived from weighted decomposition of PCA-based dictionary.
  5. S-PCA : when the sparse feature is derived from experimentally selected middle components of PCA-based dictionary.
  6. SimCO : when the sparse feature is derived from SimCO dictionary.

Thus, $\mathbf{f}_{\text{PCA}}^{\text{R}}, \mathbf{f}_{\text{KSVD}}^{\text{R}}$ and $\mathbf{f}_{\text{GAD}}^{\text{R}}$ correspond to 400-dimensional sparse feature vector derived using raw speech samples with PCA-based dictionary, KSVD dictionary and GAD, respectively. Sparse feature vector derived using 39-dimensional MFCC as initial representation for a speech frame with PCA-based dictionary, KSVD dictionary and GAD is labeled as $\mathbf{f}_{\text{PCA}}^{\text{M}}, \mathbf{f}_{\text{KSVD}}^{\text{M}}$ and $\mathbf{f}_{\text{GAD}}^{\text{M}}$, respectively. The sparse feature obtained using MFCC as initial representation when WD method is used to emphasize the discriminative information in a PCA-based dictionary is labeled as $\mathbf{f}_{\text{WD-PCA}}^{\text{M}}$. Similarly, when raw speech sample is used as initial representation, corresponding sparse feature is labeled as $\mathbf{f}_{\text{WD-PCA}}^{\text{R}}$. The sparse feature obtained using experimentally selected middle principal components is labeled as $\mathbf{f}_{\text{S-PCA}}^{\text{M}}$ and $\mathbf{f}_{\text{S-PCA}}^{\text{R}}$ for MFCC and raw speech samples as initial representation, respectively. The sparse feature obtained using SimCO dictionary is labeled as $\mathbf{f}_{\text{SimCO}}^{\text{M}}$ and $\mathbf{f}_{\text{SimCO}}^{\text{R}}$ for MFCC and raw speech samples as initial representation, respectively.

4.5. Size of dictionary

Learnt dictionaries can be both complete with total number of atoms equal to the size of each atom or overcomplete where total number of atoms are more than the size of individual atom. For the classification of speech units, it is more important for the respective dictionary to capture the discriminative information. Thus, the experiments are performed using both complete and overcomplete dictionaries. In case of overcomplete dictionaries, the number of atoms in the dictionary are more than the signal dimension. The learnt PCA dictionaries are always complete (of size $N \times N$) while learnt KSVD dictionary, SimCO dictionary and GAD could be both complete (of size $N \times N$) and overcomplete (of size $N \times 3N$ in our case). It is observed that in case of overcomplete dictionary, increasing the overcompleteness factor beyond $3N$ doesn't improve the CA. The dimension of the sparse vector obtained from complete dictionary is N and from overcomplete dictionary is $3N$. The value of N is 39 for dictionary derived using MFCC and 400 for dictionary derived using raw speech samples. The experiments were carried out using dictionaries learnt with different number of clusters. Number of clusters are also decided empirically.

5. Experimental observations

The classification results for proposed features are compared with that of the CDHMM-based classifier and SVM-based classifier with HMM-based intermediate matching kernel (HMM-IMK) using standard MFCC features (Dileep and Sekhar, 2013). In all the experiments, a left-to-right CDHMM is built for each class with varying number of states (N_s) and the number of components (Q_G) for the state specific Gaussian mixture model (GMM). The values of N_s and Q_G are varied over a set {3, 4, 5, 6, 7, 8} and the best results are obtained for $N_s = 5$

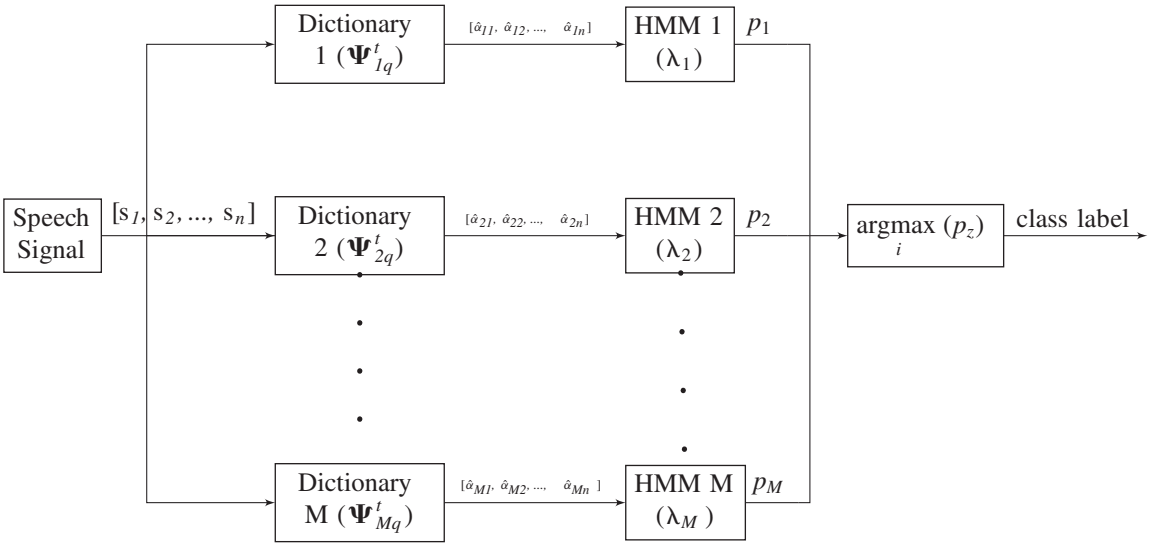

Fig. 4. Block diagram representation of testing in the proposed approach. $[\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n]$ represents n speech frames of a speech unit and $[\hat{\boldsymbol{\alpha}}_{i1}, \hat{\boldsymbol{\alpha}}_{i2}, \ldots, \hat{\boldsymbol{\alpha}}_{in}]$ represents feature vector derived using dictionaries of i^{th} class with corresponding likelihood p_i.

and $Q_G = 3$, respectively. Thus, the results reported in this paper are for a CDHMM with $N_s = 5$ and $Q_G = 3$. In addition, results presented using the proposed features are the average results over ten trials. The WD transformation of PCA dictionary emphasize the least significant PCs. These least significant PCs does not contribute to discrimination among speech units and hence they are scaled down by a factor of 10^{-3}. The number of least significant PCs scaled in WD-PCA based dictionary is $N/3$ and $N/2$, when MFCC and raw speech samples are used as initial representation, respectively. Orthogonal matching pursuit (OMP) (Rubinstein et al., 2008) is used to solve equation (9) with a fixed value of sparsity (K) as $N/3$. In the CDHMM-based systems, we have considered diagonal covariance matrices for the state-specific GMM. Testing strategy employed in this work is shown in Fig. 4. A test speech signal is presented to the CDHMM system built for each class after converting its sequence of frames to sequence of sparse feature vectors using the dictionaries specific to each class. A test speech signal is then assigned to a class for which the likelihood is maximum. In this work, the effect of number of clusters (Q) in obtaining the proposed sparse feature is examined first. Then the effectiveness of the proposed features for the classification tasks is compared with that of the standard MFCC and PLP features. Finally the performance of the proposed features is evaluated in noisy conditions.

5.1. Effect of number of clusters on performance

In this experiment, we have evaluated the performance of proposed features with respect to the number of clusters (Q). Here the raw speech samples are used as initial representation and the dictionary used is a PCA-based dictionary. Initially, training data from all the classes is pooled together to learn class-independent (CI) PCA-based dictionaries. Classification results for the proposed features obtained using CI PCA-based dictionaries (for different number of clusters, Q) are shown in Table 2. These results indicate that the performance of CI PCA-based dictionaries is not good, possibly because CI dictionaries are not able to capture the inter-class variations effectively. Thus, in order to effectively model the variations in each class, class-dependent (CD) dictionaries are employed. In CD dictionaries, multiple PCA-based dictionaries (depending on number of clusters) are learnt for each class. Classification results when the proposed features are derived using CD PCA-based dictionaries, with varying number of clusters Q are shown in Fig. 5. It can be observed that features obtained using a single CD PCA-based dictionary result in poor classification. This is because it is difficult to model all the variations present in a speech class/unit using a single dictionary. In addition, it can also be observed that the CA is high when the number of clusters is between three and seven with best at five clusters. One possible reason for this is the nature of speech units in the datasets used. In these datasets, each speech signal contains two prominent sounds (consonant and vowel), uttered either by a male or

Table 2
CA (in %) of CDHMM-based classifier (with 5 states and 3 Gaussian mixtures in each state) using the proposed features derived using class-independent PCA-based dictionaries and MFCC features.

		Proposed features with initial representation		
Dataset	Q	MFCC	Raw speech samples	MFCC
E-set	30	63.73	58.49	87.95
Hindi CV	200	35.91 ± 0.61	28.74 ± 0.58	48.87 ± 0.85
TIMIT	100	47.72	35.92	68.5

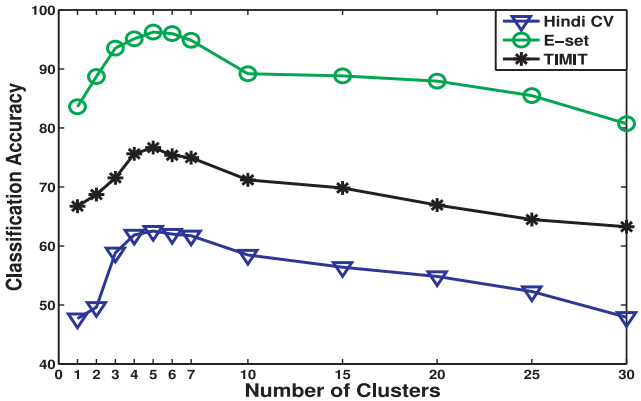

Fig. 5. CA (in %) for phoneme classification, Hindi CV units classification and E-set classification when PCA-based dictionary learnt using different number of clusters to obtain sparse features.

female speaker. There are variations in the way of pronouncing those sounds too. These variations can be effectively captured when the number of clusters are between three and seven.

5.2. *Classification results using the proposed sparse features*

Classification results of CDHMM-based classifier using different sparse features obtained on raw as well as MFCC as initial representation of speech are presented in Table 3. It is observed that the CA obtained with overcomplete KSVD dictionary, SimCO dictionary and GAD are better than those obtained using complete KSVD dictionary, SimCO dictionary and GAD, respectively. This is mainly because, more signal variations are captured in overcomplete dictionaries than that of complete dictionaries. It is also observed that KSVD and SimCO dictionaries performed better than complete PCA-based dictionary. It is possibly because signal variations are modeled well using KSVD and SimCO dictionaries. These experiments reveal that classification results obtained using KSVD and SimCO dictionaries are almost similar.

It has been shown in Shejin and Sao (2012) that middle PCs capture discriminative information and there was an improvement in face recognition when these middle PCs are used as dictionary. We also used selected middle PCs as a dictionary. Table 3 also shows results of CDHMM-based classifier using sparse features derived from middle components of PCA-based dictionary for MFCC and raw speech samples as initial representation of speech. The middle PCs are selected empirically. We selected 24 middle components ranging from 6*th* to 29*th* component when MFCC is used as initial representation. We selected 111 middle components ranging from 10*th* to 110*th* component when raw samples are used as initial representation. Thus $\mathbf{f}^{\mathrm{M}}_{\mathrm{S-PCA}}$ correspond to a 24-dimensional feature vector and $\mathbf{f}^{\mathrm{R}}_{\mathrm{S-PCA}}$ correspond to a 111-dimensional feature vector. It is seen from Table 3 that the sparse vectors obtained using these selected middle PCs in the dictionary results in best classification accuracies. However selecting these middle PCs requires exhaustive experimentation. Hence we proposed to use WD decomposition of PCA-based dictionary as discussed in Section 3.1 to emphasize the middle PCs. Classification results of CDHMM-based classifier using

Table 3
CA (in %) of CDHMM-based classifier (using 5 states (N_s) and 3 Gaussian mixtures (Q_G) in each state) using the proposed features derived using MFCC and raw speech samples as initial representation of speech for E-set classification, Hindi CV segment classification and phoneme classification.

Initial representation of speech	Dictionary (Feature)	Dictionary type	E-set classification	Hindi CV segment classification	Phoneme classification
MFCC	**PCA** ($\mathbf{f}^{M}_{PCA}$)	Complete	95.83	61.52 ± 0.73	76.45
	KSVD ($\mathbf{f}^{M}_{KSVD}$)	Complete	96.74	62.41 ± 0.59	77.73
		Overcomplete	97.59	63.58 ± 0.79	78.52
	SimCO ($\mathbf{f}^{M}_{SimCO}$)	Complete	96.16	62.03 ± 0.63	77.28
		Overcomplete	97.03	63.27 ± 0.72	78.09
	GAD ($\mathbf{f}^{M}_{GAD}$)	Complete	94.07	59.65 ± 0.61	73.37
		Overcomplete	94.58	60.74 ± 0.68	77.42
	PCA ($\mathbf{f}^{M}_{S-PCA}$)	Selected Components	98.38	64.08 ± 0.87	79.92
	WD-PCA ($\mathbf{f}^{M}_{WD-PCA}$)	Complete	98.16	63.94 ± 0.64	79.03
Raw speech samples	**PCA** ($\mathbf{f}^{R}_{PCA}$)	Complete	96.25	62.37 ± 0.85	77.06
	KSVD ($\mathbf{f}^{R}_{KSVD}$)	Complete	97.93	63.74 ± 0.76	78.47
		Overcomplete	98.37	64.91 ± 0.86	79.28
	SimCO ($\mathbf{f}^{R}_{SimCO}$)	Complete	97.51	63.35 ± 0.72	78.01
		Overcomplete	97.92	64.47 ± 0.81	78.85
	GAD ($\mathbf{f}^{R}_{GAD}$)	Complete	95.14	60.73 ± 0.79	74.43
		Overcomplete	95.76	61.39 ± 0.81	75.39
	PCA ($\mathbf{f}^{R}_{S-PCA}$)	Selected Components	99.62	66.72 ± 0.73	80.83
	WD-PCA ($\mathbf{f}^{R}_{WD-PCA}$)	Complete	99.07	66.16 ± 0.71	80.15

sparse features derived WD-PCA based dictionary are also presented in Table 3. These results are similar to the best classification accuracies obtained using selected PCs.

It can be observed that the CDHMM-based classifier using sparse features obtained from raw speech performs better as compared to MFCC features. This may be because the proposed sparse feature when derived using raw speech samples capture the inherent variations in the speech signal. The learnt dictionaries used in this work are task dependent, i.e., a set of dictionaries learnt on one dataset doesn't generalize well to other datasets. The reduction in accuracy is more when raw speech samples are used as initial representation of speech, possibly because dictionary mismatch is more in case of raw speech samples. The E-set data consist of speech units which can be considered as a subset of TIMIT, thus the dictionaries learnt on latter are used to obtain features for former. Similarly the CA for nine classes of TIMIT phonemes similar to one in E-set is also obtained with dictionaries learnt on E-set dataset. These classification results obtained for E-set database, with the dictionaries corresponding to TIMIT dataset and vice versa, and are shown in Table 4 (for feature $\mathbf{f}^{R}_{WD-PCA}$). This leads to degradation in the performance, thus emphasizing the importance of data used to learn the dictionaries. However, the degradation in CA of TIMIT is more than the E-set dataset. One possible reason could be the limited training examples in E-set dataset, thus the corresponding dictionaries can't adapt to huge variation presents in TIMIT data. On the other hand TIMIT database has more examples (with enough variability) in its training set, thus the reduction in CA is not much. Hence, the proposed method is best suitable where training condition are similar to testing conditions. In addition, an efficient dictionary learning approach can be used where dictionaries can be updated online using the testing data, but such methods are out of scope of this work.

Table 4
CA (in %) for E-set with dictionary learnt on TIMIT dataset and vice versa. CA for TIMIT dataset presented in this table is the CA for a subset of TIMIT similar to E-set dataset.

Dataset	Dictionary learnt on	Classification accuracy (%)
E-set	**E-set**	99.07
	TIMIT	94.32
TIMIT	**E-set**	48.76
	TIMIT	80.15

5.3. Comparison with the existing features

In this section performance of CDHMM-based classifier using the proposed sparse features derived using raw speech samples is compared to CDHMM-based classifier using standard MFCC features and SVM-based classifier with HMM-IMK discussed in Dileep and Sekhar (2013). Proposed features are also compared with features proposed in Sainath et al. (2011), where dictionary atoms are seeded using N-nearest neighbors, with MFCC as initial representation labeled as SR_{NN}.[4] These results are presented in Table 5 for E-set classification and Hindi CV units classification tasks.

Similarly performance of CDHMM-based classifier for TIMIT phoneme classification using the proposed features and the CDHMM-based classifier using standard MFCC, perceptual linear prediction (PLP) features, SR_{NN} and MLP-based classifier for a SR based features proposed in Sivaram et al. (2010) (labeled as l_1) is given in Table 6. Best results using proposed features for TIMIT phoneme classification are comparable to state-of-the-art classification results in Karsmakers et al. (2007), where appropriate kernels are employed to improve the accuracy. Hence we also expect similar performance gain if appropriate kernel methods are employed while learning the dictionaries (Abrol et al., 2016a, 2016b), which is deferred for future work. It is observed from Tables 5 and 6 that CDHMM-based classifier using the sparse features obtained from the proposed approach performs significantly better as compared to CDHMM-based and SVM-based classifier using the standard features.

5.4. Performance under noise

In order to test the robustness against noise of the proposed features, experiments are conducted to find the CA for the three datasets used, in noisy conditions. The speech segments of the datasets used for the three tasks are corrupted by additive babble and volvo noises taken from NOISEX-92 database (Varga and Steeneken, 1993) at $0dB$ signal to noise ratio (SNR). Equation (9) used to solve for sparse vector $\widehat{\boldsymbol{\alpha}}$ is solved using fixed sparsity. However an alternative way to solve the same equation is by fixing the reconstruction error i.e.,

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \ \| \boldsymbol{\alpha} \|_0 \quad \text{s.t.} \quad \| \mathbf{r} - \boldsymbol{\Psi}_{iq}^{t} \boldsymbol{\alpha} \|_2^2 < \epsilon. \tag{10}$$

Table 5
CA (in %) of CDHMM-based classifier using the proposed features. N_s and Q_G indicates number of HMM states and number of GMM components in each state.

Classifier	Feature	(N_s, Q_G)	Classification accuracy	
			E-set	Hindi CV
CDHMM	MFCC	(5 , 3)	87.95	48.87 ± 0.77
SVM with HMM-IMK	MFCC	(5 , 3)	95.93	59.32 ± 0.85
CDHMM	SR_{NN}	(5 , 3)	97.58	64.03 ± 0.73
CDHMM	$\mathbf{f}^{\mathrm{R}}_{\mathrm{GAD}}$	(5 , 3)	95.76	61.39 ± 0.81
	$\mathbf{f}^{\mathrm{R}}_{\mathrm{KSVD}}$	(5 , 3)	98.37	64.91 ± 0.86
	$\mathbf{f}^{\mathrm{R}}_{\mathrm{SimCO}}$	(5 , 3)	97.92	64.47 ± 0.81
	$\mathbf{f}^{\mathrm{R}}_{\mathrm{WD-PCA}}$	(5 , 3)	**99.07**	$\mathbf{66.16 \pm 0.71}$

Table 6
CA (in %) of CDHMM-based classifier with 5 states and 3 mixtures in each state using the proposed features for phoneme (TIMIT) classification.

Feature	MFCC	PLP	SR_{NN}	l_1	$\mathbf{f}^{\mathrm{R}}_{\mathrm{GAD}}$	$\mathbf{f}^{\mathrm{R}}_{\mathrm{KSVD}}$	$\mathbf{f}^{\mathrm{R}}_{\mathrm{SimCO}}$	$\mathbf{f}^{\mathrm{R}}_{\mathrm{WD-PCA}}$
Accuracy	64.5	68.47	78.74	69.38	75.39	79.28	78.85	**80.15**

[4] To derive features labeled as SR_{NN}, size of dictionary is $N \times N$, where N is equal to the dimensionality of the MFCC feature for each frame.

Table 7
CA (in %) of CDHMM-based classifier (using 5 states (N_s) and 3 Gaussian mixtures (Q_G) in each state) built using the proposed features derived from speech corrupted with 0 dB babble and volvo noises for three databases used in this work.

Noise Type	Feature	Classification accuracy		
		E-Set	Hindi CV	TIMIT
Babble (0 dB)	**MFCC**	32.37	16.12 ± 0.78	26.94
	SR_{NN}	35.28	18.76 ± 0.81	28.47
	$\mathbf{f}^{M}_{WD-PCA}$	37.92	19.85 ± 0.72	29.94
	$\mathbf{f}^{R}_{WD-PCA}$	38.53	20.52 ± 0.73	30.07
Volvo (0db)	**MFCC**	59.72	34.26 ± 0.76	47.15
	SR_{NN}	63.51	37.03 ± 0.83	49.83
	$\mathbf{f}^{M}_{WD-PCA}$	65.13	38.72 ± 0.75	50.61
	$\mathbf{f}^{R}_{WD-PCA}$	67.43	39.83 ± 0.76	51.06

The results in noisy conditions for the three tasks are presented in Table 7 are results obtained by fixing the sparsity to $N/3$, where N is the feature dimension. It is seen that the proposed features outperforms other features in presence of noise also. The reasons for this could be as follows:

1. It has been shown in the literature that the sparsity constraint has an inherent noise removal property (Cai and Wang, 2011). It should be noted that the effect of noise can't be removed fully especially in case of babble noise. However using the inherent de-noising effect of sparse feature one can obtain better results in contrast to using a full feature vector (Abrol et al., 2013).
2. The proposed WD-based dictionary used in this work emphasizes the middle PCs which correspond to the inter-class variations. The initial and last PCs are suppressed in this dictionary and thus the intra class variations are also suppressed. Since noise doesn't have much inter-class variations, thus the dictionary construction is also helping us in obtaining features robust to noise.

Thus, both dictionary construction and fixed sparsity in feature generation step helps in obtaining noise robust features. The effect of dictionary construction and fixed sparsity can be seen by obtaining the features on noisy speech with PCA dictionary where reconstruction error is used as a criterion (Eq. (10) with $\epsilon = 10^{-3}$) to solve the sparse solver (instead of fixed sparsity). The results obtained for the PCA and WD-PCA based dictionaries are shown in Table 8. It has been observed that fixing the sparsity has the maximum contribution in reducing the effect of noise. This is the possible reason for good performance of the features obtained using other dictionaries (see Table 7).

5.5. Computational complexity

In this section, the computational complexity of the proposed SR based feature generation method is analyzed and compared with existing SR based feature generation methods (Sainath et al., 2010a, 2010b, 2011; Sivaram et al., 2010), both mathematically and empirically.

Table 8
CA (in %) of CDHMM-based classifier using the proposed features derived from speech corrupted with 0*dB* babble noise. Features used are derived using complete PCA and WD-PCA dictionary with different criterion used to solve the sparse solver.

Dictionary	Feature extraction criterion	Dataset		
		E-set	Hindi CV	TIMIT
PCA	Fixed sparsity	32.73	14.85 ± 0.85	23.73
	Fixed reconstruction error	23.49	10.72 ± 0.74	17.92
WD-PCA	Fixed sparsity	38.53	20.52 ± 0.73	30.07
	Fixed reconstruction error	29.84	13.63 ± 0.79	23.74

Table 9
Mathematical computational complexity for proposed method.

	Methods		
	proposed	*TAG*	l_1
Dictionary seeding/selection	$O(Q)$	$O(T\log T)$	$O(1)$
Sparse solver	$O(N^2)$	$O(N^3)$	$O(N^3)$

5.5.1. Mathematical complexity

In the proposed approach, dictionary is learnt during training phase and its computational complexity is not considered here. The computational complexity while feature generation is considered and it involves selecting suitable dictionary for each frame and then solving a sparse solver to obtain the respective feature. The computational complexity of the proposed approach is summarized as below:

1. First step is finding the suitable dictionary for a given frame from a set of Q dictionaries for each speech class. This is done by finding the minimum euclidean distance of the speech frame in consideration with the dictionary centroid. Thus this computational complexity is $O(Q)$.
2. After selecting the suitable dictionary a sparse solver is solved to obtain a sparse vector. Orthogonal matching pursuit (OMP) is used to solve this sparse solver and its computational complexity scales as $O(N^2)$ per frame (Mailhe et al., 2009), where N is the dimensionality of the frame.

On the other hand in Sivaram et al. (2010), a single overcomplete dictionary is used, making its computational complexity $O(1)$ and then l_1 penalty is used while estimating the sparse vector, for which computational complexity scales as $O(N^3)$ (Donoho and Tsaig, 2008; Anstreicher, 1999). The obtained sparse vector is used as new feature into a multi-layer perceptron (MLP), to estimate posterior probabilities, which are in turn used as emission likelihoods of HMM states. This step also increases the computational complexity in Sivaram et al. (2010), however still it is comparable to the computational complexity of proposed approach. A number of methods are used to seed dictionary atoms in Sainath et al. (2010a, 2010b, 2011), however the one used for comparing computational complexity is where dictionary is seeded using knowledge of the top aligned Gaussian at each frame (Sainath et al., 2011) (labeled as *TAG*). Dictionary seeding involves two steps: (i) Finding N random samples from training data to seed dictionary, which involves selecting N samples without replacement from a set of total T examples. Its computational complexity is $O(T\log T)$, and (ii) These samples are used to seed a complete dictionary, which is $O(N^2)$. Thus total computation of seeding dictionary is dominated by $O(T\log T)$. Sparse solver used in Sainath et al. (2011) is approximate Bayesian compressive sensing (ABCS) that has computational complexity $O(N^3)$, where N is the number of dictionary atoms. The computational complexity for the proposed method is compared with the existing methods in Table 9.

5.5.2. Empirical complexity

The empirical complexity of the proposed SR method is calculated by computing the average time per frame to select the dictionary and solve for sparse solver. These experiments are performed using *MATLAB* 2015*b* on a machine with $Intel^{®}Core^{TM}\ i7-4770CPU@3.40GHz$ processor and 16*GB* RAM running Ubuntu 12.04*LTS* operating system. Average computational time for all the frames in E-set database for the proposed WD-PCA method using raw speech samples as initial dictionary is compared with existing methods in Table 10. The proposed method is fast

Table 10
Comparison of computational time (in seconds) for proposed method averaged over all frames in E-set database.

	Methods		
	Proposed	*TAG*	l_1
Dictionary seeding/selection	0.0018	1.4182	–
Sparse solver	0.0868	0.1163	0.1184
Total	0.0886	1.5345	0.1184

compared to the one proposed in Sainath et al. (2011) as it required very less time for dictionary selection compared to time required for seeding dictionary in Sainath et al. (2011). Also the proposed method is comparable in terms of time required when compared to method proposed in Sivaram et al. (2010).

6. Conclusions

In this work a novel feature extraction technique, based on principles of SR, has been proposed. Proposed feature uses the SR to capture the discriminative information among different speech units. Both multiple signal adaptive dictionaries and single overcomplete dictionary are used to compute the sparse vector that is then used as a feature. It is observed that the feature vector corresponding to multiple dictionaries (for same speech unit) exhibit better CA. Here, four dictionary learning methods namely PCA, KSVD, SimCO and GAD are explored to obtain the feature for speech units classification. It has been observed that representations corresponding to the PCA-based dictionary are more discriminative as compared to GAD. In addition, we observe that selected middle components of PCA-based dictionary exhibit better discrimination compared to all the components. In order to emphasize this discriminative information we proposed a transformation on the PCA-based dictionary. Experimental results using three databases support the claim that the sparse vector can be an alternative to existing features. In this work, CA of individual speech units is obtained using the proposed feature but in future we would like to extend this work for automatic speech recognition.

References

Abrol, V., Sharma, P., Sao, A.K., 2013. Speech enhancement using compressed sensing. In: Proceedings of the INTERSPEECH, pp. 3274–3278.

Abrol, V., Sharma, P., Sao, A.K., 2016a. Greedy dictionary learning for kernel sparse representation based classifier. Pattern Recognit. Lett. 78, 64–69. doi: 10.1016/j.patrec.2016.04.014.

Abrol, V., Sharma, P., Sao, A.K., 2016b. Greedy double sparse dictionary learning for sparse representation of speech signals. Speech Commun. 85, 71–82. doi: 10.1016/j.specom.2016.09.004.

Aharon, M., Elad, M., Bruckstein, A., 2006. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. Signal Process. 54 (11), 4311–4322. doi: 10.1109/TSP.2006.881199.

Anstreicher, K.M., 1999. Linear programming in $O([n^3/\ln n]L)$ operations. SIAM J. Optim. 9 (4), 803–812. doi: 10.1137/S1052623497323194.

Baby, D., Virtanen, T., Gemmeke, J.F., V. Hamme, H., 2015. Coupled dictionaries for exemplar-based speech enhancement and automatic speech recognition. IEEE/ACM Trans. Audio Speech Lang. Process. 23 (11), 1788–1799. doi: 10.1109/TASLP.2015.2450491.

Barnard, M., Koniusz, P., Wang, W., Kittler, J., Naqvi, S.M., Chambers, J., 2014. Robust multi-speaker tracking via dictionary learning and identity modeling. IEEE Trans. Multimedia 16 (3), 864–880. doi: 10.1109/TMM.2014.2301977.

Cai, T.T., Wang, L., 2011. Orthogonal matching pursuit for sparse signal recovery with noise. IEEE Trans. Inf. Theory 57 (7), 4680–4688. doi: 10.1109/TIT.2011.2146090.

Dai, W., Xu, T., Wang, W., 2012. Simultaneous codeword optimization (SimCO) for dictionary update and learning. IEEE Trans. Signal Process. 60 (12), 6340–6353. doi: 10.1109/TSP.2012.2215026.

Davis, S., Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust. Speech Signal Process. 28 (4), 357–366. doi: 10.1109/TASSP.1980.1163420.

Dileep, A.D., Sekhar, C.C., 2013. HMM based intermediate matching Kernel for classification of sequential patterns of speech using support vector machines. IEEE Trans. Audio Speech Lang. Process. 21 (12), 2570–2582. doi: 10.1109/TASL.2013.2279338.

Dong, W., Zhang, D., Shi, G., Wu, X., 2011. Image Deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. IEEE Trans. Image Process. 20 (7), 1838–1857. doi: 10.1109/TIP.2011.2108306.

Donoho, D., Stark, P., 1989. Uncertainty principles and signal recovery. SIAM J. Appl. Math. 49 (3), 906–931. doi: 10.1137/0149053.

Donoho, D.L., Tsaig, Y., 2008. Fast solution of l1-norm minimization problems when the solution may be sparse. IEEE Trans. Inf. Theory 54 (11), 4789–4812. doi: 10.1109/TIT.2008.929958.

Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Classification. Wiley, New York, USA.

Elad, M., 2010. Sparse and Redundant Representations - From Theory to Applications in Signal and Image Processing. Springer, New York, USA.

Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., Pallett, D. S., Dahlgren, N. L., 1993. DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus (Tech. Rep. NISTIR 4930, NIST).

Gemmeke, J.F., Virtanen, T., Hurmalainen, A., 2011. Exemplar-based sparse representations for noise robust automatic speech recognition. IEEE Trans. Audio Speech Lang. Process. 19 (7), 2067–2080. doi: 10.1109/TASL.2011.2112350.

Giacobello, D., Christensen, M.G., Murthi, M.N., Jensen, S.H., Moonen, M., 2010. Retrieving sparse patterns using a compressed sensing framework: applications to speech coding based on sparse linear prediction. IEEE Signal Process. Lett. 17 (1), 103–106. doi: 10.1109/LSP.2009.2034560.

Haris, B.C., Sinha, R., 2012. Sparse representation over learned and discriminatively learned dictionaries for speaker verification. In: Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing, pp. 4785–4788. doi: 10.1109/ICASSP.2012.6288989.

Hermansky, H., 1990. Perceptual linear prediction (PLP) analysis for speech. J. Acoust. Soc. Am. 87 (4), 1738–1752 http://dx.doi.org/10.1121/1.399423 .

Hermansky, H., Ellis, D.P.W., Sharma, S., 2000. Tandem connectionist feature extraction for conventional HMM systems. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 3, pp. 1635–1638. doi: 10.1109/ICASSP.2000.862024.

ISOLET Corpus, Release 1.1, Oregon Graduate Institute, Center for Spoken Language Understanding. 2000.

Jafari, M.G., Plumbley, M.D., 2011. Fast dictionary learning for sparse representations of speech signals. IEEE J. Sel. Topics Signal Process. 5 (5), 1025–1031. doi: 10.1109/JSTSP.2011.2157892.

Karsmakers, P., Pelckmans, K., Suykens, J., Hamme, H.V., 2007. Fixed-size Kernel logistic regression for phoneme classification. In: Proceedings of the INTERSPEECH, pp. 78–81.

Lee, K.F., Hon, H.W., 1989. Speaker-independent phone recognition using hidden Markov models. IEEE Trans. Acoust. Speech Signal Process. 37 (11), 1641–1648. doi: 10.1109/29.46546.

Low, S.Y., Pham, D.S., Venkatesh, S., 2013. Compressive speech enhancement. Speech Commun. 55 (6), 757–768. doi: 10.1016/j.specom.2013.03.003.

van der Maaten, L.J.P., Hinton, G., 2008. Visualizing high-dimensional data using t-SNE. J. Mach. Learn. Res. 9, 2579–2605.

Mailhe, B., Gribonval, R., Bimbot, F., Vandergheynst, P., 2009. A low complexity orthogonal matching pursuit for sparse signal approximation with shift-invariant dictionaries. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3445–3448. doi: 10.1109/ICASSP.2009.4960366.

Mitra, V., Nam, H., Espy-Wilson, C.Y., Saltzman, E., Goldstein, L., 2011. Articulatory information for noise robust speech recognition. IEEE Trans. Audio Speech Lang. Process. 19 (7), 1913–1924. doi: 10.1109/TASL.2010.2103058.

O'Toole, A.J., Deffenbacher, K.A., Valentine, D., Mckee, K., Huff, D., Abdi, H., 1997. The perception of face gender: the role of stimulus structure in recognition and classification. Mem. Cognit. 26 (1), 146–160. doi: 10.3758/BF03211378.

O'Toole, A.J., Dominique, K.D., Valentine, D., 1994. Structural aspects of face recognition and the other race effect. Mem. Cognition 22 (2), 208–224. doi: 10.3758/BF03208892.

Povey, D., Kingsbury, B., Mangu, L., Saon, G., Soltau, H., Zweig, G., 2005. fMPE: discriminatively trained features for speech recognition. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing., Vol. 1, pp. 961–964. doi: 10.1109/ICASSP.2005.1415275.

Rabiner, L.R., Schafer, R.W., 2010. Theory and Applications of Digital Speech Processing. Pearson Education Limited, NJ, USA.

Rath, T.M., Manmatha, R., 2003. Word image matching using dynamic time warping. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. II−521−II−527. doi: 10.1109/CVPR.2003.1211511.

Rubinstein, R., Zibulevsky, M., Elad, M., 2008. Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit, CS Technion.

Saenko, K., Livescu, K., Glass, J., Darrell, T., 2009. Multistream articulatory feature-based models for visual speech recognition. IEEE Trans. Pattern Anal. Mach. Intell. 31 (9), 1700–1707. doi: 10.1109/TPAMI.2008.303.

Sainath, T.N., Carmi, A., Kanevsky, D., Ramabhadran, B., 2010a. Bayesian compressive sensing for phonetic classification. In: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing, pp. 4370–4373. doi: 10.1109/ICASSP.2010.5495638.

Sainath, T.N., Ramabhadran, B., Nahamoo, D., Kanevsky, D., Sethy, A., 2010b. Sparse representation features for speech recognition. In: Proceedings of the INTERSPEECH, pp. 2254–2257.

Sainath, T.N., Ramabhadran, B., Picheny, M., Nahamoo, D., Kanevsky, D., 2011. Exemplar-based sparse representation features: from TIMIT to LVCSR. IEEE Trans. Audio Speech Lang Process. 19 (8), 2598–2613. doi: 10.1109/TASL.2011.2155060.

Sezer, O.G., Guleryuz, O.G., Altunbasak, Y., 2015. Approximation and compression with sparse orthonormal transforms. IEEE Trans. Image Process. 24 (8), 2328–2343. doi: 10.1109/TIP.2015.2414879.

Sharma, P., Abrol, V., Dileep, A.D., Sao, A.K., 2015. Sparse coding based features for speech units classification. In: Proceedings of the INTERSPEECH, pp. 712–715.

Shashanka, M.V.S., Raj, B., Smaragdis, P., 2007. Sparse overcomplete decomposition for single channel speaker separation. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 2, . doi: 10.1109/ICASSP.2007.366317.

Shejin, T., Sao, A.K., 2012. Significance of dictionary for sparse coding based face recognition. In: Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG), pp. 1–6.

Sivaram, G.S.V.S., Nemala, S.K., Elhilali, M., Tran, T.D., Hermansky, H., 2010. Sparse coding for speech recognition. In: Proceedings of the IEEE Conference on Acoustics Speech and Signal Processing, pp. 4346–4349. doi: 10.1109/ICASSP.2010.5495649.

Tosic, I., Frossard, P., 2011. Dictionary learning. IEEE Signal Process. Mag. 28 (2), 27–38. doi: 10.1109/MSP.2010.939537.

Tropp, J.A., Gilbert, A.C., 2007. Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Inf. Theory 53 (12), 4655–4666. doi: 10.1109/TIT.2007.909108.

Varga, A., Steeneken, H.J.M., 1993. Assessment for automatic speech recognition: II. NOISEX-92: a database and an experiment to study the effect of additive noise on speech recognition systems. Speech Commun. 12 (3), 247–251 http://dx.doi.org/10.1016/0167-6393(93)90095-3.

Xu, T., Wang, W., Dai, W., 2013. Sparse coding with adaptive dictionary learning for underdetermined blind speech separation. Speech Commun. 55 (3), 432–450. doi: 10.1016/j.specom.2012.12.003.

Yilmaz, E., Gemmeke, J.F., V. Hamme, H., 2014. Noise robust exemplar matching using sparse representations of speech. IEEE Trans. Audio Speech Lang. Process. 22 (8), 1306–1319. doi: 10.1109/TASLP.2014.2329188.

Zubair, S., Yan, F., Wang, W., 2013. Dictionary learning based sparse coefficients for audio classification with max and average pooling. Digit. Signal Process. 23 (3), 960–970 http://dx.doi.org/10.1016/j.dsp.2013.01.004.