Sparse Modeling of Posterior Exemplars for Keyword Detection

Dhananjay Ram[1,2], *Afsaneh Asaei*[1], *Pranay Dighe*[1,2], *Hervé Bourlard*[1,2]

[1]Idiap Research Institute, Martigny, Switzerland
[2]Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland
{dhananjay.ram,afsaneh.asaei,pranay.dighe,herve.bourlard}@idiap.ch

Abstract

Sparse representation has been shown to be a powerful modeling framework for classification and detection tasks. In this paper, we propose a new keyword detection algorithm based on sparse representation of the posterior exemplars. The posterior exemplars are phone conditional probabilities obtained from a deep neural network. This method relies on the concept that a keyword exemplar lies in a low-dimensional subspace which can be represented as a sparse linear combination of the training exemplars. The training exemplars are used to learn a dictionary for sparse representation of the keywords and background classes. Given this dictionary, the sparse representation of a test exemplar is used to detect the keywords. The experimental results demonstrate the potential of the proposed sparse modeling approach and it compares favorably with the state-of-the-art HMM-based framework on Numbers'95 database.

Index Terms: Keyword Detection, Deep neural network posterior features, Compressive sensing, Sparse word posterior probabilities, Dictionary learning, Sparse modeling

1. Introduction

Keyword detection deals with identification of selected words in speech utterances. The field received added attention due to the ever increasing volume of speech data being stored or shared through the internet. These are primarily raw data, without any transcription. Keyword detection provides a way for content based indexing of the data. In this context, exemplar-based keyword detection plays a key role to enable the detection procedure without any requirement for transcription.

1.1. Prior Works

The keyword detection methods can be broadly classified into two approaches namely, supervised and unsupervised [1]. Among different supervised approaches, acoustic keyword spotting and large vocabulary continuous speech recognition (LVCSR) based system are most popular.

In acoustic keyword detection system [2], parallel network of keywords and background filler model are generated. The filler model, also called garbage model, is constructed using phone loops which are modeled using a HMM/GMM system. Log-likelihood scores are computed using Viterbi decoding to take the decision. In LVCSR based system [3], N-best answers along with their Viterbi alignments are generated, which is used to calculate normalized likelihood score for the keywords. Although, automatic speech recognition (ASR) has been a popular method for keyword detection (spotting), it finds limitation in this task due to its dependency on corresponding language model, training data (which can be very limited in under-resourced languages) and relatively lower accuracy. On the other hand, keyword detection does not necessarily need transcribed speech and its performance can be much better than a ASR system. Hence, unsupervised keyword detection methods are studied particularly in the context of exemplar-based spoken term detection.

The most famous technique in unsupervised keyword detection is template matching using dynamic time warping (DTW). In this method, the sequence of keyword exemplars (templates) are aligned with test utterances to find the degree of similarity. Both spectral and posterior based features has been used for this purpose [4]. Although, exemplar-based matching is very effective, the computational cost of this method is prohibitive; so extensive research has been conducted to propose variants of DTW that can be applied at lower cost with improved accuracy [5].

1.2. Our Contributions

This paper proposes a novel exemplar-based approach for keyword detection. Motivated by the success of exemplar-based sparse representation in classification and detection tasks [6, 7], our goal is to use the sparse representation of the test exemplars for keyword detection. Exemplar-based sparse representation has been recently applied in speech recognition and demonstrated great potential for speech classification [8, 9, 10]. To the extent of our knowledge, this idea has not been studied in the context of spoken term detection.

The intuition behind this work is that a speech utterance can be decomposed as a combination of the keyword and background words. Considering a dictionary of the training exemplars for the keyword and background speech, a test exemplar has a sparse representation using this dictionary where the values of the sparse representation coefficients determine the weights of the training exemplars in our compositional model. Sparse representation leads to an inherent capability of discriminating different classes [7, 11] corresponding to the keyword or background speech.

We cast the keyword detection problem as the problem of subspace classification via sparse representation. To that end, a dictionary for characterizing the space of keyword and background exemplars is learned from the training data. The dictionary learning for sparse representation models the space of keyword and background exemplars as union of subspaces where any realization of the test exemplar lies on a low-dimensional subspace. The sparse recovery process implicitly leads to a competition between the two subspaces of keyword and background exemplars. Therefore, the recovered sparse representation is naturally discriminative and in this high-dimensional space, the keyword and background subspaces become separable. One of the advantages of our proposed approach is that there is no explicit assumption on the statistical distribution of the observed data as in the previous keyword detection (spotting) algorithms [3, 2, 12].

2. Keyword Detection using Sparse Representation

We cast the keyword detection problem as classification of the subspace of test exemplars. The subspaces correspond either to the *keyword* or to the *background*.

2.1. Sparse Modeling of Posterior Exemplars

To formalize the problem, let x_t denote the acoustic features extracted from the speech utterance at frame t. The Mel-frequency cepstral coefficients (MFCC) or perceptual linear prediction (PLP) features are typical acoustic features used for classification of speech data. In our posterior-based framework, these acoustic features are used as the inputs to the deep neural network (DNN) to extract the phonetic conditional posterior probabilities $p(q_k|x_t)$. The set of all posterior probabilities are denoted by $y_t = [p(q_1|x_t)\, p(q_2|x_t) \dots p(q_K|x_t)]^\top$ and used as the posterior exemplar. Our goal is to detect a keyword using a sequence of posterior exemplars $\mathbf{Y} = [y_1 \dots y_T]$ directly without performing speech recognition.

We assume that the posterior exemplars lie on a low-dimensional (non-linear) manifold which can be characterized using a union of subspaces (UoS) model. This relation is formalized as

$$\underbrace{\begin{bmatrix} p(q_1|x_t) \\ p(q_2|x_t) \\ \vdots \\ p(q_K|x_t) \end{bmatrix}}_{y_t} = \underbrace{\begin{bmatrix} p(q_1|w_1) & \cdots & p(q_1|w_L) \\ p(q_2|w_1) & \cdots & p(q_2|w_L) \\ \vdots & \vdots & \vdots \\ p(q_K|w_1) & \cdots & p(q_K|w_L) \end{bmatrix}}_{\text{Dictionary matrix:}\mathbf{D}} \times \underbrace{\begin{bmatrix} p(w_1|x_t) \\ \vdots \\ p(w_l|x_t) \\ \vdots \\ p(w_L|x_t) \end{bmatrix}}_{\alpha_t} \tag{1}$$

where w_l is a linguistically driven unit associated to each subspace. For simplicity and without loss of generality, we assume that w_l indicates a word. Of course, any sup/sub-word subspace can be considered and (1) holds due to the marginalization rule of probabilities. It may be noted that the phonetic posterior probabilities and the acoustic features x_t become independent given the underlying word class, i.e. $p(q_k|w_l, x_t) = p(q_k|w_l)$.

Following the UoS model, the space of phonetic representation for a word w_l denoted by $d_l = [p(q_1|w_l)\, p(q_2|w_l)\, \dots p(q_K|w_l)]^\top$ can be expressed as

$$\underbrace{\begin{bmatrix} p(q_1|w_l) \\ p(q_2|w_l) \\ \vdots \\ p(q_K|w_l) \end{bmatrix}}_{d_l} = \underbrace{\begin{bmatrix} p(q_1|sw_1^{w_l}) & \cdots & p(q_1|sw_{S_{w_l}}^{w_l}) \\ p(q_2|sw_1^{w_l}) & \cdots & p(q_2|sw_{S_{w_l}}^{w_l}) \\ \vdots & \vdots & \vdots \\ p(q_K|sw_1^{w_l}) & \cdots & p(q_K|sw_{S_{w_l}}^{w_l}) \end{bmatrix}}_{\text{Word manifold modeling dictionary:}\mathbf{D}_{w_l}} \times \underbrace{\begin{bmatrix} p(sw_1^{w_l}|w_l) \\ \vdots \\ p(sw_s^{w_l}|w_l) \\ \vdots \\ p(sw_{S_{w_l}}^{w_l}|w_l) \end{bmatrix}}_{a_{w_l}} \tag{2}$$

where $sw_s^{w_l}$ stands for the sth sub-word unit of the word w_l, and S_{w_l} represents the total number of (over-complete) bases to model the sub-space of word w_l. Based on (1) and (2), $\alpha_t = [a_{w_1}^\top p(w_1|x_t) \dots a_{w_L}^\top p(w_L|x_t)]^\top$.

The number of words is typically far more than the number of phonetic classes, $L \gg K$; hence, (1) is a an under-determined linear system. On the other hand, only one word is associated to any posterior exemplar and the activation of the representative coefficients are grouped as a_{w_l}. Hence, α_t has a group sparsity structure. This prior knowledge on the structure of α_t enables us to recover the subspace dependent probabilities for an acoustic observation $p(w_l|x_t)$ given the UoS dictionary for sparse representation. In the following sections, we explain the dictionary learning procedure for modeling the space of posterior exemplars in Section 2.2. Given this dictionary learned from the training exemplars, we explain the keyword detection using sparse representation of the test exemplars in Section 2.3.

Algorithm 1 Class-specific online dictionary learning

```
Require: : $\mathbf{Y}_{w_l}$, $\lambda$ (regularization parameter), $\mathbf{D}^0_{w_l}$ (initialization)
for $t = 1$ to $T$ do
    Sparse coding of $y_t$ to determine $\alpha_t$:
        $\alpha_t = \arg\min_{\alpha} \left\{ \frac{1}{2} \|y_t - \mathbf{D}^{(t-1)}_{w_l} \alpha\|_2^2 + \lambda \|\alpha\|_1 \right\}$
    Updating $\mathbf{D}^{(t)}_{w_l}$ with $\mathbf{D}^{(t-1)}_{w_l}$ as warm restart:
        $\mathbf{D}^{(t)}_{w_l} = \arg\min_{\mathbf{D}_{w_l}} \left\{ \frac{1}{t} \sum_{i=1}^{t} (\frac{1}{2} \|y_i - \mathbf{D}_{w_l} \alpha_i\|_2^2 + \lambda \|\alpha_i\|_1) \right\}$
end for
return $\mathbf{D}^{(T)}_{w_l}$
```

2.2. Class-specific Dictionary Learning

The phonetic components of each word are a small subset of the whole phonetic space. Hence, the manifold of word posterior representation is intuitively low-dimensional. We rely on dictionary learning to characterize the word manifolds D_{w_l} individually. Sparse representation using the dictionaries for the keyword and background speech (all the other words) enables us to identify the low-dimensional subspace of the test exemplars and its correspondence to the keyword or background classes.

Given a training set of features $\mathbf{Y} = [y_1, ..., y_T] \in \mathbb{R}^{K \times T}$, a dictionary $\mathbf{D} \in \mathbb{R}^{K \times M}$ and sparse representation $\mathbf{A} = [\alpha_1, ..., \alpha_T]$ for $\mathbf{Y}$; the objective function for classical dictionary learning techniques is defined as

$$\arg\min_{\mathbf{D}, \mathbf{A}} \frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{2} \|y_t - \mathbf{D}\, \alpha_t\|_2^2 + \lambda \|\alpha_t\|_1 \right) \tag{3}$$

where λ is the regularization parameter. The first term in this expression, quantifies the energy-based *reconstruction error*. The second term denotes the ℓ_1-norm of α defined as $\|\alpha\|_1 = \sum_i |\alpha_i|$ which controls the sparsity of α_t. The joint optimization of this objective function with respect to both $\mathbf{D}_{w_l}$ and α_t simultaneously is non-convex, it can be solved as a convex objective by optimizing for one while keeping the other fixed.

In this paper, we consider the fast online optimization proposed by Mairal et at. [13] for learning the dictionaries word-wise based on stochastic approximations. The algorithm basically alternates between a step of sparse decoding for the current training feature y_t and then optimizes the previous estimate of dictionary $\mathbf{D}^{(t-1)}_{w_l}$ to determine the new estimate $\mathbf{D}^{(t)}_{w_l}$ using stochastic gradient descent. The algorithm has been shortly summarized in Algorithm 1.

Given the dictionaries for sparse representation of the keyword and background words, in Section 2.3 we explain our keyword detection algorithm using sparse representation.

2.3. Keyword-Background Sparse Representation

The union of subspaces approach as exploited for sparse modeling of posterior exemplars through (1)–(2) is applicable for general classification of the linguistic events. As the dictionaries are learned class-wise to model the generative process underlying the training exemplars of (1), the reconstruction error obtained from sparse representation has been shown very effective for classification and detection tasks [6, 7].

In this paper, our goal is binary classification of the posterior exemplars as *keyword* or *background.* We assume that keyword is w_1 and the background consists of $\{w_l\}_{l=2}^{L}$. Accordingly, we define

$$\mathbf{D}^k = \mathbf{D}_{w_1}, \quad \mathbf{D}^b = [\mathbf{D}_{w_2} \dots \mathbf{D}_{w_L}] \tag{4}$$

The class-specific dictionaries constructed in this manner, in principle, should be able to span the whole space of the keyword and background posteriors. More specifically, the test exemplars corresponding to a keyword can be approximated as a linear combination of training exemplars from the keyword exemplars. Similarly, if the background dictionary consists of all representative background words[1], the features corresponding to background can be sparsely represented as a linear combination of training background exemplars.

Our proposed keyword detection algorithm is based on sparse representation using the keyword and background dictionaries. The sparse representation enables projection of the features from low-dimensional space of phonetic posteriors to a much higher dimensional space where the classification becomes easier since the new representation space is more discriminative [11]. The support of the sparse components are associated to the independent subspaces of the dictionary required for characterization of the low-dimensional subspace of the observations [11].

The sparse vectors corresponding to the dictionaries $\mathbf{D}^k$ and $\mathbf{D}^b$ are denoted by α_t^k and α_t^b. Therefore, by combining the two dictionaries, the test exemplar can be written as a sparse linear combination of training exemplars from both keyword and background as

$$\begin{aligned} y_t &= \mathbf{D}^k\alpha_t^k + \mathbf{D}^b\alpha_t^b \\ &= \underbrace{[\mathbf{D}^k \quad \mathbf{D}^b]}_{\mathbf{D}} \times \underbrace{\begin{bmatrix}\alpha_t^k \\ \alpha_t^b\end{bmatrix}}_{\alpha_t} = \mathbf{D}\,\alpha_t \end{aligned} \tag{5}$$

The sparse vector, α_t is obtained by projecting the posterior feature vector y_t onto a higher dimensional space using the dictionary, $\mathbf{D}$. The sparse recovery process inherently introduces a competition between the two subspaces. Thus, the recovered sparse representation is naturally discriminative. In Section 2.4, we explain how keyword detection can be achieved from the sparse representations.

2.4. Keyword Detection

Given the sparse representation of a test exemplar at time frame t using the keyword and background dictionaries, the class of y_t is obtained by comparing the reconstruction error for the respective classes. The errors are calculated as follows

$$e_k(y_t) = \|y_t - \mathbf{D}^k\alpha_t^k\|_2 \tag{6}$$

$$e_b(y_t) = \|y_t - \mathbf{D}^b\alpha_t^b\|_2 \tag{7}$$

where, $e_k(y_t)$ and $e_b(y_t)$ are the error terms corresponding to the keyword and background classes. These error terms are fed to the detector to take a frame level decision. The output of the detector is calculated by,

$$\Delta(y_t) = e_b(y_t) - e_k(y_t) \gtrless \delta \tag{8}$$

where, δ is a predefined threshold. If $\Delta(y_t) > \delta$, then y_t is labeled as a keyword-frame, otherwise y_t is marked as a background-frame.

Once the frame level decision is made, the next step is to take an utterance level decision, i.e. whether the keyword exists in the utterance under consideration or not. In order to achieve this, the frame level decisions are accumulated by counting the number of continuous frames corresponding to a particular keyword. This provides the length of a keyword in a test utterance in terms of the number of frames. Now, to take the final decision, this length is compared with a predefined threshold [12]. The lengths of different keywords can be extracted from the development data and different statistical measures (e.g. mean, variance) can be used as threshold. We have used the minimum length of a keyword as the corresponding threshold.

3. Experimental Analysis

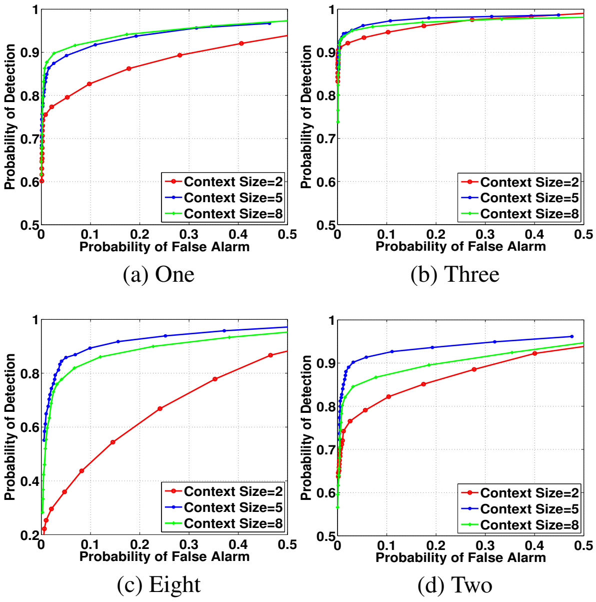
The keyword detection experiments are conducted to evaluate the performance of the proposed sparse modelling framework. We have used Numbers'95 database for our experiments. Overall there are 31 words in form of continuous speech out of which 11 words are used for our keyword detection experiments. These keywords are 'zero', 'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine' and 'fifteen'. In total, there are around 17k sentences, among which 60% are used for training, 20% for development and the rest for testing.

To perform the keyword detection, first the word-specific dictionaries are learned as discussed in Section 2.2 using the online dictionary learning algorithm presented in Algorithm 1 [13]. There are two sets of words in the database, one having large number of exemplars and other having very small number of exemplars. While learning the word-specific dictionaries, for the first set we have used 50 exemplars to initialize the dictionary whereas, for the other set, we have used 60% of the exemplars for initialization. Rest of the data is used to train the dictionaries. The number of exemplars to initialize dictionaries is restricted to a predefined size in order to keep the size of the dictionary lower. We have also added a dictionary for silence which is learned in a similar way by extracting the silences from the training data. This task is straightforward since a silence state is already trained at the output of the DNN extracting the posterior exemplars. The DNN setup to extract posterior exemplars is similar to [14].

In order to integrate the temporal information inherent in the speech signal, context appending of the posterior exemplar for a frame with neighbouring frames is implemented. For a context size c, the new feature vector, y_t' corresponding to a frame y_t is constructed by stacking c left and right frames onto y_t such that $y_t' = [y_{(t-c)}^\top \ \cdots \ y_t^\top \ \cdots \ y_{(t+c)}^\top]^\top$.

Once the word-specific dictionaries are learned, the *keyword* and *background* dictionary for a *keyword* is constructed as discussed in Section 2.3 through concatenation of the keyword and background dictionaries as shown in Equation (5). This dictionary is then used to obtain the sparse representation of a test utterance on a frame-by-frame basis. Afterwards, the

[1] Recall that for simplicity we refer to w_l as a word. In general, w_l can represent any sub-word unit.



(a) One

(b) Three

(c) Eight

(d) Two

Figure 1: ROC curves for different keywords

Table 1: Probability of Detection and False Alarm for different Keywords for a Predefined Threshold

Keyword	**Prob. of Detection** **and False Alarm** (our approach)	**Prob. of Detection** **and False Alarm** (HMM based [12])
zero	0.9813 / 0.0149	0.9400 / 0.0150
one	0.9160 / 0.0686	0.9800 / 0.0950
two	0.8208 / 0.0150	-
three	0.9373 / 0.0132	-
four	0.8284 / 0.1028	0.9270 / 0.1370
five	0.8414 / 0.0028	0.8270 / 0.0016
six	0.8165 / 0.0173	-
seven	0.8123 / 0.0424	-
eight	0.8194 / 0.0673	-
nine	0.8280 / 0.0501	-
fifteen	0.7692 / 0.3592	0.6730 / 0.3310

reconstruction errors are calculated and a frame level decision is made using a predefined threshold δ as explained in Section 2.4. Final decision is made by comparing the length of detected keyword with its minimum length learned from the training data.

The ROC curves are plotted by varying the frame level threshold δ in a given range. These curves are generated for three different context sizes. The resulting curves are shown in Figure 1 for four different keywords. The curves clearly show the dependency of keyword detection performance on the context size. Although in most cases a context of 5 gives better results, some of them are better for a context of 8. This behaviour can be explained as dependency of the optimal context size on the length of a keyword. For smaller-duration words (e.g. two) lower context size and for larger-duration words (e.g. fifteen) higher context size works better. But, very small context size (e.g. 2) may affect the performance due to its incapability to capture sufficient temporal information. On the other hand, larger context size may also affect the performance by capturing temporal information from neighbouring words. Thus to achieve an optimal operating point, we need to tune δ as well as the context size.

Additionally, we compare our results with an HMM-based approach proposed in [12]. Hence, we extract the results for a specific operating point on the ROC curves of the keywords that corresponds to [12] as listed in Table 1. A context size of 8 was chosen for these experiments. We have varied our frame level threshold δ in a range to obtain a set of probabilities of detection and false alarm. The false alarm corresponding to the words presented in [12] is chosen to be as close as possible and shown with the corresponding probability of detection. For other words, we have chosen similar false alarm values and displayed them with corresponding detection probability. A comparison between the two results indicate that our system has comparable or better performance.

4. Extension to Subword Models

The background dictionary defined in (4) can be constructed using any subword dictionaries such as phones. The concatenation of these subword dictionaries yield the background dictionary. Then, the dictionary corresponding to the keyword is learned from the available examples. Although the phones of the keyword already exist in the background dictionary, their temporal information is only modeled in the target dictionary through context appending that captures the trajectory or transition between subword units. Hence, sparse representation of the keyword discriminates between the target and background dictionaries in favor of the more representative “bases“. On the other hand, the non-keyword speech can be better represented using background dictionary due to the mismatch in background and target temporal trajectories. This approach eliminates the need of transcriptions for the keyword (spoken term) and the background dictionary can be learned from the (well resourced) speech database independent of the keyword. Our experiments on Numbers’95 using phone dictionaries yield comparable results to word modeling presented in Figure 1.

5. Conclusions

In this paper, we proposed a novel keyword detection algorithm based on sparse representation of the posterior exemplars. In contrast to the conventional exemplar-based methods, we used dictionary learning to model the manifolds of keyword and background exemplars as union of subspaces where the posterior exemplars admit sparse representations. This approach enables us to classify the subspace of the test exemplars using the reconstruction error of sparse representation.

The sparse recovery process implicitly leads to a competition between the two subspaces of keyword and background exemplars. Hence, the recovered sparse representation leads to better discrimination and enables detection of a keyword at the frame level. The frame level decisions are accumulated to make an utterance level decision through consecutive counting. This approach makes the decoding very simple and the decision threshold has an intuitive relationship with the keyword length. The proposed idea is successfully implemented and compared with the HMM-based approach.

6. Acknowledgments

The research leading to these results has received funding from by SNSF projects on “Parsimonious Hierarchical Automatic Speech Recognition (PHASER)” grant agreement number 200021-153507 and “Adaptive Multilingual Speech Processing (A-MUSE)” grant agreement number 200020-144281.

7. References

[1] A. Mandal, K. P. Kumar, and P. Mitra, "Recent developments in spoken term detection: a survey," *International Journal of Speech Technology*, vol. 17, no. 2, pp. 183–198, 2014.

[2] R. C. Rose and D. B. Paul, "A hidden markov model based keyword recognition system," in *Acoustics, Speech, and Signal Processing, 1990. ICASSP-90., 1990 International Conference on*. IEEE, 1990, pp. 129–132.

[3] M. Weintraub, "LVCSR log-likelihood ratio scoring for keyword spotting," in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, vol. 1. IEEE, 1995, pp. 297–300.

[4] M.-C. Silaghi and H. Bourlard, "Iterative posterior-based keyword spotting without filler models," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*. Citeseer, 1999, pp. 213–216.

[5] Y. Zhang and J. R. Glass, "Unsupervised spoken keyword spotting via segmental dtw on gaussian posteriorgrams," in *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*. IEEE, 2009, pp. 398–403.

[6] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.

[7] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Sparse representation for target detection in hyperspectral imagery," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 3, pp. 629–640, 2011.

[8] T. N. Sainath, B. Ramabhadran, D. Nahamoo, D. Kanevsky, D. Van Compernolle, K. Demuynck, J. F. Gemmeke, J. R. Bellegarda, and S. Sundaram, "Exemplar-based processing for speech recognition: An overview," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 98–113, 2012.

[9] T. N. Sainath, B. Ramabhadran, M. Picheny, D. Nahamoo, and D. Kanevsky, "Exemplar-based sparse representation features: From TIMIT to LVCSR," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 8, pp. 2598–2613, 2011.

[10] J. F. Gemmeke, T. Virtanen, and A. Hurmalainen, "Exemplar-based sparse representations for noise robust automatic speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 7, pp. 2067–2080, 2011.

[11] A. Fawzi and P. Frossard, "Classification of unions of subspaces with sparse representations," in *Signals, Systems and Computers, 2013 Asilomar Conference on*. IEEE, 2013, pp. 1368–1372.

[12] H. Ketabdar, J. Vepa, S. Bengio, and H. Bourlard, "Posterior based keyword spotting with a priori thresholds," in *International Conference on Spoken Language Processing (ICSLP)*, no. LIDIAP-CONF-2006-017, 2006.

[13] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research (JMLR)*, vol. 11, pp. 19–60, 2010.

[14] P. Dighe, A. Asaei, and H. Bourlard, "Sparse modeling of neural network posterior probabilities for exemplar-based speech recognition," *Speech Communication (to appear)*, 2015.