Discriminative Dictionary Learning for Mixture Component Detection with Application to RF Signal Recognition

Hao Chen and Seung-Jun Kim
Dept. of Computer Science & Electrical Engineering
University of Maryland, Baltimore County
Baltimore, MD 21250
{chenhao1,sjkim}@umbc.edu

Thomas Chatt
Lockheed Martin Corporation
thomas.j.chatt@lmco.com

***Abstract*—Pattern classification algorithms based on sparse dictionary learning are derived. After training a discriminative dictionary and a linear classifier using the samples of the individual classes, the aim is to apply the dictionary and classifier for recognizing the component signals in a mixture of different class signals. A key issue is to prevent the "leakage" of strong signal components to weaker components in the classifier. We tackled this issue by encouraging orthogonality among the discriminants during the training, applying the algorithms to RF signal recognition verified the efficacy of the approach.**

I. Introduction

In the past decade, cognitive radio research has contributed in developing radio systems that are aware of the operating RF spectrum environment and adapt their transmission strategies accordingly. The cognitive radio paradigm thus mitigates the inefficiencies of hard spectrum allocation through opportunistic and dynamic spectrum usage. However, the developed spectrum sensing and access techniques were largely based on the domain knowledge of the RF signals and the propagation characteristics [1], [2]. Therefore, much effort was placed on developing elaborate signal models and corresponding signal processing mechanisms.

Recently, there has been a surge of interest in data-driven machine learning approaches in various areas, notably computer vision and artificial intelligence. Exploiting the prevalence of data, one of the key ideas is to learn appropriate signal representations suitable for the given application from the data themselves [3]. Sparse representation models were shown effective for face recognition tasks even with pose variation and occlusion [4]. Task-specific signal dictionaries were learned from the data [5]. Breakthrough performances were demonstrated in image classification and speech recognition using deep neural network architectures [6], [7]. In RF signal classification, nonparametric signal clustering was performed in [8]. Convolutional neural networks were employed to classify the modulations of communication signals [9].

In this work, classification algorithms based on dictionary learning are derived. Our goal is to train a discriminative dictionary and a linear projection based on the examples of the signals in the individual classes, but to apply the trained dictionary and classifier for classifying the component signals when the input has a mixture of signals belonging to multiple classes. The motivation is the following. Suppose that there are C signal classes and the mixture input contains signals from L classes. Then, the number of possible combinations is $\binom{C}{L}$. Thus, performing the training directly using the mixture samples and the labels indicating the constituent components can quickly become prohibitive as C and L grow.

In the context of RF signal classification, one can collect samples from Wi-Fi, Zigbee, Bluetooth, and frequency hopping spread spectrum (FHSS) transmissions individually. Then, our goal is to train the dictionary and classifier that can detect the component signals in any mixtures such as Wi-Fi + Zigbee or Wi-Fi + Bluetooth + FHSS, where the component signals share a common band.

Related works in discriminative dictionary learning include [5], [10]–[12]. However, these works do not treat component detection in mixtures. Note that our derivation has similarities with [10] in that Fisher discrimination criterion is adopted. Differently from [10], however, our methods do not train separate dictionaries for individual classes. Furthermore, a linear projection matrix is learned jointly.

The rest of the paper is organized as follows. In Sec. II, an algorithm for learning the discriminative dictionary and linear projection is derived. In Sec. III, the algorithm is extended to better cope with mixture component detection. The numerical test results of the application to RF signal recognition are presented in Sec. IV. Conclusions are provided in Sec. V.

II. Discriminative Dictionary Learning

A. Problem Formulation

Consider a dataset consisting of M-dimensional feature vectors from C classes. Let $\mathbf{X}_c \in \mathbb{R}^{M\times N_c}$ be the collection of feature vectors $\mathbf{x} \in \mathcal{X}_c$ belonging to class c, where $N_c = |\mathcal{X}_c|$ is the number of class-c samples. Define $\mathcal{X} := \bigcup_{c=1}^{C} \mathcal{X}_c$ and $\mathbf{X} := [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_C] \in \mathbb{R}^{M\times N}$, where $N = \sum_c N_c$. It is

This work was supported in part by NSF grant 1547347 and Lockheed Martin Corporation.

postulated that the $\mathbf{X}$ can be characterized well by a union-of-subspaces model. That is, given a dictionary $\mathbf{D} \in \mathbb{R}^{M\times K}$ with K atoms, $\mathbf{X}$ can be approximated well as $\mathbf{X} \approx \mathbf{DZ}$ for a coefficient matrix $\mathbf{Z} \in \mathbb{R}^{K\times N}$. A typical method to learn the dictionary is to impose sparsity constraint on $\mathbf{Z}$ as in

$$\min_{\mathbf{D}\in\mathcal{D},\mathbf{Z}} \|\mathbf{X}-\mathbf{DZ}\|_F^2 + \lambda\|\mathbf{Z}\|_1 \tag{1}$$

where $\mathcal{D}$ is a constraint set for the dictionary. For example, $\mathcal{D} := \{[\mathbf{d}_1,\ldots,\mathbf{d}_K] \in \mathbb{R}^{M\times K} : \|\mathbf{d}_k\|_2^2 \le 1, k = 1,\ldots,K\}$ can be used for unit-norm atoms. $\|\cdot\|_F$ is the Frobenious norm, $\|\mathbf{Z}\|_1$ is the sum of the absolute values of all entries in $\mathbf{Z}$, and $\lambda > 0$ is a hyperparameter tuning the sparsity of $\mathbf{Z}$.

Our goal is, however, not just to represent the data well, but also to separate the data in different classes. To obtain a discriminative dictionary, the idea of Fisher linear discriminant is adopted as in [10]. However, differently from [10], where the coefficients $\mathbf{Z}$ were directly used in the Fisher criterion, a linear transformation is employed here to reduce the dimensionality of the discriminant function and align the discriminant better to the class centroids.

Let $\mathbf{z} \in \mathcal{Z}_c$ is the coefficient vector corresponding to a sample $\mathbf{x} \in \mathcal{X}_c$. Then $\mathbf{W} \in \mathbb{R}^{K\times Q}$ defines the linear discriminant variables through $\mathbf{y} = \mathbf{W}^T\mathbf{z} \in \mathbb{R}^Q$ for $\mathbf{z} \in \mathcal{Z} := \bigcup_{c=1}^C \mathcal{Z}_c$, where Q is the dimension of the discriminant variables satisfying $C \le Q \le K$. Define the within-class scatter $\mathbf{S}_W$ and the between-class scatter $\mathbf{S}_B$ as ($\cdot^T$ denotes transposition)

$$\mathbf{S}_W := \sum_{c=1}^{C}\sum_{\mathbf{z}\in\mathcal{Z}_c} (\mathbf{z}-\mathbf{m}_c)(\mathbf{z}-\mathbf{m}_c)^T \tag{2}$$

$$\mathbf{S}_B := \sum_{c=1}^{C} N_c(\mathbf{m}_c-\mathbf{m})(\mathbf{m}_c-\mathbf{m})^T \tag{3}$$

respectively, where $\mathbf{m}_c := N_c^{-1}\sum_{\mathbf{z}\in\mathcal{Z}_c}\mathbf{z}$ is the class sample mean, and $\mathbf{m} := N^{-1}\sum_{\mathbf{z}\in\mathcal{Z}}\mathbf{z}$ is the sample mean over all data. The idea of Fisher discriminant analysis is to minimize the within-class scatter of the discriminants $\mathbf{W}^T\mathbf{S}_W\mathbf{W}$ at the same time maximizing the between-class scatter $\mathbf{W}^T\mathbf{S}_B\mathbf{W}$. A possible penalty term to minimize for this purpose is

$$f(\mathbf{W},\mathbf{Z}) := \mathrm{tr}\left(\mathbf{W}^T\mathbf{S}_W\mathbf{W}\right) - \mathrm{tr}\left(\mathbf{W}^T\mathbf{S}_B\mathbf{W}\right) + \|\mathbf{W}^T\mathbf{Z}\|_F^2 \tag{4}$$

where the last term is added to ensure convexity in $\mathbf{Z}$ [13]. let $\mathbb{1}_N$ be an $N\times N$ matrix, whose entries all equal to 1. Define also $N\times N$ matrices

$$\mathbf{H}_1 := \mathrm{bdiag}\left\{\frac{1}{N_1}\mathbb{1}_{N_1}, \frac{1}{N_2}\mathbb{1}_{N_2}\ldots, \frac{1}{N_C}\mathbb{1}_{N_C}\right\} \tag{5}$$

$$\mathbf{H}_2 := \frac{1}{N}\mathbb{1}_N \tag{6}$$

where $\mathrm{bdiag}\{\cdot\}$ constructs a block-diagonal matrix by arranging the matrices in $\{\cdot\}$ on the diagonal. Then, it can be verified that

$$\begin{aligned} f(\mathbf{W},\mathbf{Z}) &= \|\mathbf{W}^T\mathbf{Z}(\mathbf{I}-\mathbf{H}_1)\|_F^2 \\ &\quad - \|\mathbf{W}^T\mathbf{Z}(\mathbf{H}_1-\mathbf{H}_2)\|_F^2 + \|\mathbf{W}^T\mathbf{Z}\|_F^2. \end{aligned} \tag{7}$$

1: Initialize $\mathbf{D}$ and $\mathbf{W}$
2: Repeat
3: Update $\mathbf{Z}$ by solving $\min_{\mathbf{Z}} \|\mathbf{X}-\mathbf{DZ}\|_F^2 + \lambda\|\mathbf{Z}\|_1 + \mu f(\mathbf{W},\mathbf{Z})$
4: Set $\mathbf{A} = \mathbf{ZZ}^T$, $\mathbf{B} := \mathbf{XZ}^T$, and $\mathbf{D}^{(0)} = \mathbf{D}$
5: For $i = 1,2,\ldots,I_{\max}$
6: For $k = 1,2,\ldots,K$
7: $\bar{\mathbf{D}} = [\mathbf{d}_1^{(i)},\ldots,\mathbf{d}_{k-1}^{(i)},\mathbf{d}_k^{(i-1)},\ldots,\mathbf{d}_K^{(i-1)}]$
8: $\tilde{\mathbf{d}}_k^{(i)} = \frac{1}{A_{kk}}\left(\mathbf{b}_k - \bar{\mathbf{D}}\mathbf{a}_k\right) + \mathbf{d}_k^{(i-1)}$
9: $\mathbf{d}_k^{(i)} = \frac{\tilde{\mathbf{d}}_k^{(i)}}{\max\left\{\|\tilde{\mathbf{d}}_k^{(i)}\|_2, 1\right\}}$
10: Next k
11: Next i
12: Set $\mathbf{D} = \mathbf{D}^{(I_{\max})}$
13: Update $\mathbf{W}$ by setting the q-th column to the q-th smallest eigenvector of $\mathbf{S}_W - \mathbf{S}_B + \mathbf{ZZ}^T$ for all q
14: Until convergence

TABLE I
ALGORITHM 1 FOR SOLVING (8)–(9).

The overall optimization problem is

$$\min_{\mathbf{D}\in\mathcal{D},\mathbf{Z},\mathbf{W}} \|\mathbf{X}-\mathbf{DZ}\|_F^2 + \lambda\|\mathbf{Z}\|_1 + \mu f(\mathbf{W},\mathbf{Z}) \tag{8}$$

$$\text{subject to } \mathbf{W}^T\mathbf{W} = \mathbf{I} \tag{9}$$

where $\mu > 0$ is a hyperparameter. Constraint (9) is added to avoid the trivial solution $\mathbf{W} = \mathbf{0}$, and to make sure that all columns of $\mathbf{W}$ do not become identical.

B. Learning Algorithm Derivation

The optimization problem (8)–(9) is not convex in $(\mathbf{Z},\mathbf{D},\mathbf{W})$ jointly. However, if any two variables are fixed, the optimization with respect to the remaining variable can be solved easily. Thus, an alternating minimization method is proposed.

First, with $\mathbf{D}$ and $\mathbf{W}$ fixed, the problem for $\mathbf{Z}$ is a convex optimization problem, which can be solved fast with various algorithms that can deal with a ℓ_1-norm regularizer. For example, the fast iterative shrinkage-thresholding algorithm (FISTA) can be employed [14].

With $\mathbf{Z}$ and $\mathbf{W}$ fixed, the objective of (8) is minimized with respect to $\mathbf{D} \in \mathcal{D}$. This is a convex optimization problem, and it can be solved, for example, using the block coordinate descent (BCD) method, where each column of $\mathbf{D}$ constitutes a block. Define $\mathbf{A} := \mathbf{ZZ}^T$ and $\mathbf{B} := \mathbf{XZ}^T$. Denote the (k,k)-entry of $\mathbf{A}$ as A_{kk}, and the k-column of $\mathbf{A}$ (or $\mathbf{B}$) as $\mathbf{a}_k$ ($\mathbf{b}_k$). Then, by denoting the k-th column of $\mathbf{D}$ at iteration i as $\mathbf{d}_k^{(i)}$, the update is done for $k = 1,2,\ldots,K$ at each iteration $i = 1,2,\ldots$, as

$$\tilde{\mathbf{d}}_k^{(i)} = \frac{1}{A_{kk}}\left(\mathbf{b}_k - \mathbf{D}^{(i-1)}\mathbf{a}_k\right) + \mathbf{d}_k^{(i-1)} \tag{10}$$

$$\mathbf{d}_k^{(i)} = \frac{\tilde{\mathbf{d}}_k^{(i)}}{\max\left\{\|\tilde{\mathbf{d}}_k^{(i)}\|_2, 1\right\}} \tag{11}$$

until convergence.

Finally, with $\mathbf{D}$ and $\mathbf{Z}$ fixed, it can be shown that the resulting problem for $\mathbf{W}$ boils down to an eigenvector problem. Specifically, the columns of $\mathbf{W}$ are chosen to be the unit-norm eigenvectors of $(\mathbf{S}_W - \mathbf{S}_B + \mathbf{ZZ}^T)$ corresponding to the Q smallest eigenvalues. The algorithm is summarized in Table I.

1: Initialize $\mathbf{D}$, $\mathbf{W}$, and $\mathbf{U}$
2: Repeat
3: Update $\mathbf{Z}$ by solving
$\min_{\mathbf{Z}} \left[\|\mathbf{X}-\mathbf{D}\mathbf{Z}\|_F^2+\lambda\|\mathbf{Z}\|_1+\mu f(\mathbf{W},\mathbf{Z}) + \nu g(\mathbf{W},\mathbf{Z},\mathbf{U})\right]$
4: Update $\mathbf{D}$ via lines 4–12 in Table I
5: Update $\mathbf{W}$ via (14)
6: Update $\mathbf{U}$ via (15)
7: Until convergence

TABLE II
ALGORITHM 2 FOR SOLVING (13)

III. Discriminative Dictionary Learning for Mixture Detection

A. Problem Formulation

Now suppose that a test sample is due to a *mixture* of L different classes signals. The goal is to identify the individual components present in the mixture. Note that using the mixture samples and the corresponding labels of the constituent components for training would require generating $\binom{C}{L}$ training sets. Instead, our pragmatic approach is to still train the dictionary using the single-component samples in $\mathcal{X}$, but ensure that the discriminant variables have a favorable structure for detecting the component features in mixtures.

An important issue that materializes is how to prevent the "leakage" of a strong component from confusing the detectors for the other class components. In other words, one needs to ensure certain orthogonality among the discriminants for different classes. A simple idea is to introduce an additional penalty term to the training objective that promotes orthogonality between the class centroids $\{\mathbf{W}^T\mathbf{m}_c\}$. Let $\mathbf{M} := [\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_C]$, a penalty that encodes this notion is

$$g(\mathbf{W},\mathbf{Z},\mathbf{U}) := \|\mathbf{W}^T\mathbf{M}-\mathbf{U}\|_F^2 \quad \text{s.t. } \mathbf{U}^T\mathbf{U}=\mathbf{I} \tag{12}$$

The overall optimization problem thus becomes

$$\begin{aligned}\min_{\mathbf{D}\in\mathcal{D},\mathbf{Z},\mathbf{W},\mathbf{U}} & \|\mathbf{X}-\mathbf{D}\mathbf{Z}\|_F^2+\lambda\|\mathbf{Z}\|_1 \\ & +\mu f(\mathbf{W},\mathbf{Z})+\nu g(\mathbf{W},\mathbf{Z},\mathbf{U}) \\ \text{subject to } & \mathbf{U}^T\mathbf{U}=\mathbf{I}\end{aligned} \tag{13}$$

where $\nu > 0$ is a hyperparameter.

B. Algorithm Derivation

To solve (13), the alternating minimization strategy is adopted here again. With $\mathbf{Z}, \mathbf{U}$ and $\mathbf{W}$ fixed, the problem for $\mathbf{D}$ is the same as in Sec. II, so lines 4–12 in Table I can again be employed. To update $\mathbf{Z}$, it is noted that the objective of (13) consists of a convex quadratic term in $\mathbf{Z}$ and an ℓ_1-norm penalty term. Thus, the FISTA can be employed to minimize the objective over $\mathbf{Z}$ with $\mathbf{D}$, $\mathbf{W}$, and $\mathbf{U}$ fixed. Next, the problem for $\mathbf{W}$ is convex quadratic, whose solution is obtained in a closed form as

$$\begin{aligned}\mathbf{W} = \{&\mu\mathbf{Z}\left[(\mathbf{I}-\mathbf{H}_1)^2-(\mathbf{H}_1-\mathbf{H}_2)^2+\mathbf{I}\right]\mathbf{Z}^T \\ &+\nu\mathbf{M}\mathbf{M}^T\}^{-1}\nu\mathbf{M}\mathbf{U}^T.\end{aligned} \tag{14}$$

Finally, the update for $\mathbf{U}$ is done by solving

$$\mathbf{U} = \arg\min_{\mathbf{U}:\mathbf{U}^T\mathbf{U}=\mathbf{I}} \|\mathbf{U}-\mathbf{W}^T\mathbf{M}\|_F^2. \tag{15}$$

BLE	**0**	**0**	**0**	0	0	0
Bluetooth	**1**	0	0	**1**	**0.93**	0
FHSS	0.05	**0**	0.07	**0**	0.93	**0**
Zigbee	0.95	0	**0.07**	1	**0**	**0**

(a) Algorithm 1.

BLE	**0**	**0**	**0**	0	0	0
Bluetooth	**0.04**	0	0	**0**	**0**	0.11
FHSS	0.03	**0**	0.17	**0**	0	**0**
Zigbee	0.01	0	**0.17**	0	**0**	**0.11**

(b) Algorithm 2.

TABLE III
CLASSIFICATION ERROR RATES WHEN $L = 2$.

For $K \geq P$ and $\mathbf{Y} := \mathbf{W}^T\mathbf{M}$ full-rank, (15) has a solution given by $\mathbf{U} = \mathbf{Y}\mathbf{V}\mathbf{S}^{-1/2}\mathbf{V}^T$, where $\mathbf{V}$ is orthonormal and $\mathbf{S}$ diagonal with $\mathbf{Y}^T\mathbf{Y} = \mathbf{V}\mathbf{S}\mathbf{V}^T$ [15]. The overall algorithm is listed in Table II.

For testing a new sample $\mathbf{x}$, one first computes the corresponding sparse code $\mathbf{z}$ via

$$\min_{\mathbf{z}} \|\mathbf{x}-\mathbf{D}\mathbf{z}\|_2^2+\lambda\|\mathbf{z}\|_1 \tag{16}$$

and use various classifiers on $\mathbf{y} = \mathbf{W}^T\mathbf{z}$, such as the nearest neighbor (NN) or logistic regression classifiers.

IV. Application to RF Signal Recognition

The proposed method was applied to the recognition of various RF signals in the 2.4 GHz band. First, the complex baseband samples of RF transmissions were collected inside a RF shield box, at a sampling rate of 100 MHz. We also collected the complex samples at the sampling rate of 40 MHz. A 2-layer scattering network was employed to extract the feature vectors $\mathbf{x}$ from the complex signals of duration 200 ms [16]. Deep scattering spectrum is a contractive representation, which can capture higher-order statistics and scale interactions, while remaining stable (Lipschitz continuous) to deformations [17]. The window size for temporal averaging was 100 ms. Two wavelets per octave were used in the first layer, and 1 wavelet per octave in the second. A Morlet wavelet was used for the bandpass filter, and Gabor for the lowpass.

A. Experiments with 100 MHz Samples

In order to see the performance of the mixture case, Algorithm 1 and 2 were tested using a 4-class dataset, which consists of the Bluetooth Low Energy (BLE), Bluetooth, FHSS, and Zigbee signals. The training set contained $1,500$ samples per class and the test set 450 samples per class. We trained a dictionary of size $K = 100$ with $\lambda = 1$, which resulted around 30% non-zero entries in $\mathbf{Z}$. The values for μ and ν were set to 0.1 and 1, respectively. We used the NN classifier in this experiment.

The average classification error rates from using Algorithm 1 and 2 are listed in Table III, when the number of the component signals L is equal to 2. Each column represents a particular mixture combination. The boldface numbers are the miss-detection rates of the corresponding class signals, which were *present* in the mixture. The non-bold numbers represent the false detection rates of the corresponding class, which was *absent* in the mixture. It can be seen from Table III(a)

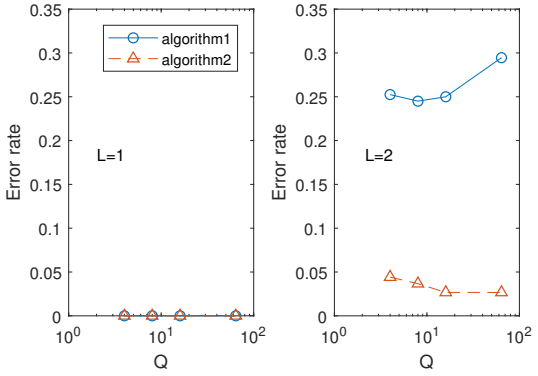

Fig. 1. Classification error rates for pure and mixture signals.

Training SNR \ Testing SNR	20 dB	0 dB	−20 dB	−40 dB
20 dB	1.00	0.93	0.30	0.17
0 dB	0.95	1.00	0.28	0.17
−20 dB	0.50	0.59	1.00	0.17
−40 dB	0.17	0.17	0.19	0.17

TABLE IV
CLASSIFICATION ACCURACY OF NON-MIXTURE SIGNALS AT DIFFERENT SNR LEVELS.

that Algorithm 1 often makes mistakes with Bluetooth signals. This is because that the Bluetooth samples collected for the experiments were around 15 dB weaker than the other class signals, giving rise to a *near-far* scenario. No changes in the individual signal powers were made when generating the mixture signals. It can be seen from Table III(b) that the near-far problem is much alleviated by using Algorithm 2.

This advantage is further illustrated in Fig. 1, where the average classification error rates for different values of Q are plotted. The left panel corresponds to the pure signal case, and the right panel to the case where $L = 2$ signals were mixed. It can be seen that Algorithm 2, which encourages the orthogonality in the class centroids, performs much better than Algorithm 1 for the mixtures.

B. Experiments with 40 MHz Samples

For the 40 MHz dataset, 45 sec worth of RF data for each of the 6 classes, namely BLE, Bluetooth, FHSS1, FHSS2, Wi-Fi1 and Wi-Fi2, were collected. From those raw RF footages, 449 samples of duration 200 ms were extracted per class by allowing overlaps of up to 100 ms. Out of the 449 samples, 300 samples were used for training, 74 samples for cross-validation, and 75 samples for testing in the non-mixture case. A dictionary with $K = 100$ atoms was trained using only the non-mixture samples by means of Algorithm 2. The value of λ was set to 1.5, which resulted around 26% non-zero entries in $\mathbf{Z}$. The values of μ and ν were set to 0.1 and 1000, respectively. A logistic regression classifier was trained using 200 mixture samples in total, in addition to the non-mixture samples. The logistic regression classifier can cope with the nonlinearity effect manifested in the mixture case, and also provide the estimates of posterior probabilities of the individual classes.

First, the classification performance at different SNR levels was analyzed. The entire classification algorithm was trained

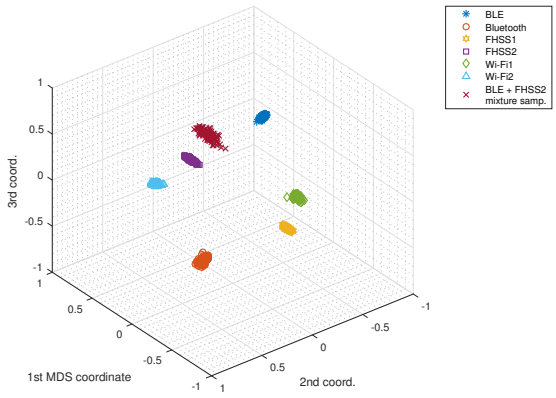

Fig. 2. MDS of $\mathbf{W}^T\mathbf{Z}$ for training and mixture samples with BLE + FHSS2.

at various SNR levels $\texttt{SNR}_{train} \in A' := \{20, 0, -20, -40\}$ in dB, and tested at $\texttt{SNR}_{test} \in A'$. It can be seen from Table IV that if the training is done at 20 dB SNR, the SNR mismatch of up to 20 dB can be tolerated. Also, if the training SNR matches with the testing SNR, the classifier performs well at as low as −20 dB SNR.

Before analyzing the classification accuracy for the mixture case, in order to gain understanding of how the training with pure non-mixture samples can work with the mixture test samples as well, the discriminant feature vectors $\mathbf{W}^T\mathbf{Z}$ are visualized. The value of Q is set to 16. Thanks to our orthogonalization penalty, the 6 cluster centroids corresponding to the 6 classes are approximately orthogonal in the 16-dimensional space. Since it is difficult to visualize 16-dimensional vectors, multi-dimensional scaling (MDS) is employed to reduce the dimension down to 3. Needless to say, the orthogonality of the 6 centroids cannot be preserved in the 3-dimensional space. Still the method can portray reasonably well the situation in the original 16-dimensional space.

Fig. 2 shows the six point clouds, each corresponding to the training samples of a class. The point clouds were obtained from the dictionary trained from the collected samples without adding extra simulated noise. The six point clouds are seen well separated from each other, which is important for the accuracy of classification. Fig. 2 also illustrates the case of mixture signals. In particular, the red crosses correspond to the feature vectors for the mixture of BLE and FHSS2 signal classes. It can be seen that the red crosses are situated somewhat midway between the BLE cloud and the FHSS2 cloud, and at the same time, far away from other class clouds, allowing the classifier to reliably declare that the samples are mixtures of the BLE and FHSS2 signals. It must be emphasized that such an intuitively pleasing property of the obtained feature arises precisely from the *orthogonality* of the individual class features, engineered by our dictionary learning formulation, as well as the approximate *linearity* of deep scattering spectrum under mixture inputs, which is a benefit of well-designed deep layered architecture.

In order to assess the classification performance in the mixture signal case, random combinations of the test samples from L different classes were generated. Subsequently, the

L	1	2	3	4	5
Accuracy	1.00	1.00	0.98	0.96	0.97

TABLE V
CLASSIFICATION ACCURACY FOR EQUAL POWER CASES.

SNR (dB)	∞	20	0	-20	-40
Accuracy ($L = 2$)	0.99	1.00	1.00	1.00	0.31

TABLE VI
CLASSIFICATION ACCURACY FOR EQUAL-POWER MIXTURES OF $L = 2$ COMPONENTS.

logistic regression classifier estimated the posterior probabilities $\Pr(c|\mathbf{W}^T\mathbf{z})$ for $c = 1, 2, \ldots, C$. Then, L classes with highest posterior probabilities were picked and the accuracy was assessed. Table V lists the resulting accuracies when L was varied from $L = 1$ (non-mixture) to $L = 5$. The individual signal powers were not adjusted before adding the components. Also, additive noise was not considered. It can be seen that a very good classification accuracy is achieved for all the mixture scenarios.

To see the robustness of the classifier for mixture signals under additive Gaussian noise, various SNR levels were simulated and the classification performance with $L = 2$ components was assessed. It can be seen from Table VI that the classification is quite reliable up to -20 dB SNR.

To analyze the robustness of the proposed technique under near-far scenarios, we repeated the experiment with one out of the $L = 2$ signals having the power 3 dB, 6 dB, 10 dB, or 20 dB weaker than the other. All possible combinations using the 6 classes were considered and the accuracy was averaged. Note that the logistic regression classifier was trained using only the equal power mixture samples. Table VII shows the classification accuracy of the *weaker* signals. The accuracies for the stronger signals were all close to 100%. It can be seen from Table VII that up to 10 dB power difference can be quite well tolerated, but the performance degrades significantly at a 20 dB power ratio.

Power ratio (dB)	0	3	6	10	20
Accuracy ($L = 2$)	0.997	0.976	0.952	0.808	0.335

TABLE VII
CLASSIFICATION ACCURACY FOR NON-EQUAL POWER CASES.

V. CONCLUSION

Machine learning-based classification algorithms for raw RF signals were developed using deep layered architectures. In order to mitigate the excessive burden of training from scratch, a deep layered architecture called the scattering network was adopted, which is similar in nature to convolutional neural networks (CNNs) but does not require training. It was then identified that a critical task in RF signal classification is to handle mixture component detection. Two important challenges emerged. First, rather than using the signal samples from all $2^C - 1$ combinations of individual classes for training, it was desired for the efficiency of training, only the C non-mixture datasets should be used. Second, the trained features should have certain robustness against the near-far situations, where stronger signals can overwhelm the detection of the weaker signals. A discriminative dictionary learning approach was devised, which approximately orthogonalized the trained features of different classes, addressing both of the challenges effectively. Experiments using real datasets collected at a 100 MHz sampling rate, the clear advantage of the orthogonalization strategy was seen from mixture detection accuracies. reliable detection was possible at an SNR level as low as -30 dB. In the more challenging 40 MHz sampling rate case, the mixture component detection was seen to be reliably done when the power ratio of the component signals was up to 10 dB. In the equal power case, almost perfect classification of 2-component mixtures was seen at an SNR of -20 dB.

REFERENCES

[1] E. Axell, G. Leus, E. G. Larsson, and H. V. Poor, "Spectrum sensing for cognitive radio: State-of-the-art and recent advances," *IEEE Sig. Process. Magazine*, vol. 29, no. 3, pp. 101–116, Apr. 2012.

[2] S.-J. Kim, E. Dall'Anese, J. A. Bazerque, K. Rajawat, and G. B. Giannakis, "Advances in spectrum sensing and cross-layer design for cognitive radio networks," in *Academic Press Library in Signal Processing*, R. Chellappa and S. Theodoridis, Eds., chapter 9, pp. 471–502. Academic Press, Waltham, MA, 2014.

[3] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[4] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[5] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 791–804, Apr. 2012.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., pp. 1097–1105. 2012.

[7] G. Hinton, L. Deng, D. Yu, et al., "Deep neural networks for acoustic modeling in speech recognition," *IEEE Sig. Process. Magazine*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[8] M. Bkassiny, S. K. Jayaweera, and Y. Li, "Multidimensional dirichlet process-based non-parametric signal classification for autonomous self-learning cognitive radios," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5413–5423, Nov. 2013.

[9] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. of the 17th Int'l. Conf. Engineering Applications Neural Netw. (EANN)*, Aberdeen, UK, Sept. 2016, pp. 213–226.

[10] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *Proc. of IEEE Int'l. Conf. Comput. Vision (ICCV)*, Nov. 2011, pp. 543–550.

[11] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proc. of IEEE Conf. Comput. Vision Pattern Recogn. (CVPR)*, San Francisco, CA, June 2010, pp. 2691–2698.

[12] S.-J. Kim, "Online kernel dictionary learning," in *Proc. of IEEE Global Conf. Sig. Info. Process.*, Orlando, FL, Dec. 2015, pp. 14–16.

[13] S. Li and Y. Fu, "Learning robust and discriminative subspace with low-rank constraints," *IEEE Trans. Neural Networks Learning Syst.*, vol. 27, no. 11, pp. 2160–2173, Nov. 2016.

[14] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[15] R. Lai and S. Osher, "A splitting method for orthogonality constrained problems," *J. Sci. Comput.*, vol. 58, pp. 431–449, 2014.

[16] J. Andén and S. Mallat, "Deep scattering spectrum," *IEEE Trans. Sig. Process.*, vol. 62, no. 16, pp. 4114–4128, Aug. 2014.

[17] J. Bruna and S. Mallat, "Invariant scattering convolution networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1872–1886, Aug. 2013.