DISCRIMINATIVE DICTIONARY LEARNING VIA MUTUAL EXCLUSION

Raghu G. Raj

U.S. Naval Research Laboratory, Radar Division
Washington D.C. 20375, U.S.A.

ABSTRACT

We apply our recently developed concept of mutual exclusivity [1] in the context of discriminative coding, to the problem of learning dictionary for representing signals drawn from N classes in a way that optimizes their discriminability. We first briefly review our mutual-exclusivity concept and then deploy it a simple discriminative dictionary learning algorithm that directly generalizes the well-known KSVD algorithm which is addressed for the traditional problem of signal coding. We demonstrate performance improvements over traditional KSVD based feature extraction schemes and conclude by describing avenues for future research.

Index Terms— ATR, discriminative dictionary, mutual exclusivity, sparsity, feature selection

1. INTRODUCTION

A fundamental signal processing and machine learning problem is the discrimination of signals into *N* different classes. Traditionally the emphasis has been laid on optimizing the classification engine (given the features of the object being analyzed) ranging from the classical LDA (linear discriminative analysis) of Fischer [2] to Support Vector Machines (SVMs) of Vapnik [3]. More recently *probabilistic graphical modeling* based approaches such as [4-6] have shown considerable promise in delivering state-of-the-art results for ATR (automatic target recognition).

However an equally important aspect of the problem is the determination of features (corresponding to the signals)—given which the classification engine can be applied. The choice of features is crucial in determining the success of the overall classification system *regardless* of the classification engine being applied. In this paper we focus on this latter problem in the context of discriminative dictionary learning (DDL).

The generation of features for signal classification generally occurs in two stages. The first stage consists of the determination of *low-level features* based oftentimes on physical intuition of the processes generating the data or the phenomenology associated with the application domain. Examples of these include, respectively, micro-Doppler features in radar systems [7], SIFT descriptors and their variants for images [8]. For the second stage we observe that in principle such low-level features can again be treated as data vectors from which higher-level features can be extracted and so forth (as is done for example in so-called deep neural networks [9]). At the end of this process we again have a set of feature vectors corresponding to each class (which could comprise of any combination of features from various levels).

We demonstrate in this paper that the final set of features thus obtained from the above two-stage process, organized in the form of a data matrix *X* (where each column of the matrix is a feature of the signal), can be discriminatively coded in an optimum fashion using the concept of mutual exclusivity that we introduced in [1] via our dM-KSVD (Discriminative Mutually-Exclusive KSVD) algorithms that we proposed in this paper. The proposed dM-KSVD algorithms are designed to maximize the separability of the resulting features in feature space.

In the next section we briefly review the concept of mutual exclusivity and its important properties. Thereafter we describe our dM-KSVD algorithm in some detail in Section 3. In Section 4 we demonstrate Simulation and conclude in Section 5 with a discussion of the future work stemming from this work.

2. MUTUAL EXCLUSIVITY

Let $\Phi \in R^{dxD}$ be a dictionary of unit norm vectors in the columns such that any vector in $x \in R^d$ can be described by linear combination of columns of Φ. We typically operate in the under-determined case wherein $D > d$. Given this suppose we have 2 classes of vectors generated from Φ thus:

$$\mathbf{X_1} = \mathbf{\Phi C_1} \quad (1)$$

$$\mathbf{X_2} = \mathbf{\Phi C_2} \quad (2)$$

such that $C_i = [c_1^i, \cdots, c_N^i]$ is a matrix of *N* codes ($c_1^i \in R^D$) corresponding to *class i*; and where $X_i = [x_1^i, \cdots, x_N^i] \in R^{dxN}$ contains the corresponding set of realizations ($x_1^i \in R^d$) of *class i*. Given this we define mutual exclusivity as follows [1]:

Definition 1. Two codes c_1 and c_2 are mutually exclusive if for all *k* we have: $(c_1(k) \neq 0) \Rightarrow (c_2(k) = 0)$ and $(c_2(k) \neq 0) \Rightarrow (c_1(k) = 0)$

Definition 2. Given matrix codes C_1 and C_2 for class#1 and class#2 respectively as defined above, we define the 3 different *mutual-exclusion operators* $\mathfrak{M}_k(C_1, C_2)$ between the two classes as follows:

1. $\mathfrak{M}_1(C_1, C_2) = \|C_1 \odot C_2\|_1$ (3)

2. $\mathfrak{M}_2(\mathrm{C}_1, \mathrm{C}_2) = \| |C_1| + |C_2| \|_F - \sqrt{\|C_1\|_F^2 + \|C_2\|_F^2}$ (4)

3. $\mathfrak{M}_3(\mathrm{C}_1, \mathrm{C}_2) = \sqrt{\|C_1\|_F^2 + \|C_2\|_F^2} - \frac{\|C_1\|_F^2 + \|C_2\|_F^2}{\| |C_1| + |C_2| \|_F + k}$ (5)

where, $|D|$ yields a matrix whose every element is an absolute value of the corresponding element in D; where $\|.\|_F$ is the matrix Frobenius norm operator; and where $k \geq 0$ is a pre-defined constant. When referring to mutual exclusivity operators in general we use the subscript $\mathfrak{M}$. When $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^d$ are vectors, for shorthand we denote $\mathfrak{M}(x, y)$ by $\mathfrak{M}(z)$ where $z = [x^T y^T]^T$. Following are some relevant theoretical properties of the $\mathfrak{M}$ operator.

Lemma 1 [1] (Positivity): $\mathfrak{M}(\mathrm{C}_1, \mathrm{C}_2) \geq 0$ □

Lemma 2 [1] (*Mutual Exclusivity*): $\mathfrak{M}(\mathrm{C}_1, \mathrm{C}_2) = 0$ if and only if the non-trivial set of matrix codes C_1 and C_2 are mutually exclusive □

Lemma 3 (Separable and Approximate Convexity):
Let $\mathrm{x}, \mathrm{y} \in \mathbb{R}^d$

a) $\mathfrak{M}_1(x, y)$ is separably convex in each variable
b) $\mathfrak{M}_2$ is approximately convex in the following sense:
$\mathfrak{M}_2(\lambda \mathrm{x} + (1-\lambda)\mathrm{y}) \leq \lambda \mathfrak{M}_2(x) + (1-\lambda)\mathfrak{M}_2(y) + 0.5.\lambda(1-\lambda)\|x-y\|^2$
c) $\mathfrak{M}_3$ is approximately convex in the following sense:
$\mathfrak{M}_3(\lambda \mathrm{x} + (1-\lambda)\mathrm{y}) \leq \lambda \mathfrak{M}_3(x) + (1-\lambda)\mathfrak{M}_3(y) + \mathrm{C}.\lambda(1-\lambda)\|x-y\|^2$

where $C \approx 0.92$. Furthermore all the $\mathfrak{M}$ operators above are asymptotically convex (i.e. as $d \to \infty$) for Gaussian signals□

In the next section we describe how the mutual exclusivity operator is deployed for constructing discriminative dictionary learning algorithms.

3. DDL ALGORITHMS

The central computational task that we are concerned with is as follows: Find the best dictionary to discriminate the samples $\{x_i^1\}_{i=1}^N$ drawn from class#1 from the samples $\{x_i^2\}_{i=1}^N$ drawn from class#2, by solving:

$$\begin{aligned} &\text{minimize}_{\Phi,\mathrm{C}} \left\|X_1 - \Phi C_1\right\|_F^2 + \left\|X_2 - \Phi C_2\right\|_F^2 \\ &\text{subject to } \mathfrak{M}(C_1, C_2) = 0 \\ &\qquad \|C_1\|_1 \leq \tau \\ &\qquad \|C_2\|_1 \leq \tau \end{aligned} \quad (6)$$

(where $X_1 \in \mathbb{R}^{nxN}$, $X_2 \in \mathbb{R}^{nxN}$ are matrices whose columns are drawn respectively from Class#1 and Class#2)

We solve the above computational task via two algorithmic procedures—dM-KSVD1 and dM-KSVD2—that incorporate the mutual exclusivity operator in different ways. These algorithms are described, respectively, in *Figure 3* and *Figure 4*. Our dM-KSVD algorithms furnish a dictionary Φ that enables the generation of discriminative feature vectors for Class#1 and Class#2 that are both sparse and mutually exclusive with respect to Φ.

Both the dM-KSVD algorithms replace the traditional sparse coding stage with a discriminative coding stage while leaving the KSVD based dictionary update intact. An important quantity that is used by these algorithms is the *steering vector* which we define as follows:

Definition 3. Let $\{c_i^1\}_{i=1}^N$ and $\{c_i^2\}_{i=1}^N$ be training samples of Class#1 and Class#2 respectively, then the *steering vector* of Class#*i* (with respect to Class#*j*) is given by:

$$g_i(x) = \frac{1}{N}\sum_{t=1}^N \mathfrak{M}_1(x, c_t^j) \quad \square$$

In dM-KSVD1 the training vectors of a given class are iteratively coded with respect to the associated steering vector for that class, followed by update of the steering vector and so forth. In dM-KSVD2 however all the training vectors are coded separately w.r.t. the steering vectors of both classes and thereafter the sparsest of the two solutions is chosen to be the final code vector associated with the given sample. The resulting discriminative codes (features) of the training samples are used to training a classifier (in our case SVMs) which can then be used to classify the test samples.

Once the discriminative dictionary is obtained as above, the test samples are discriminatively coded in the manner in which it was performed for the dM-KSVD1 or dM-KSVD2 algorithm (depending on which method was used to calculate the dictionary Φ). Thereafter the resulting test sample is classified using the classifier trained above.

4. SIMULATION RESULTS

We created two classes of vectors with controlled levels of similarity and distortion as follows. Firstly we generated both classes from a overcomplete dictionary: $\Phi = [\Phi_1\ \Phi_2] \in \mathbb{R}^{dxD}$, where $\Phi_1 \neq \Phi_2$ and $\Phi_\mathrm{i} \in \mathbb{R}^{dx(\frac{D}{2})}$ is a orthonormal dictionary (i.e. $d = D$)—which is chosen to be either Fourier, Dirac or Haar dictionaries. Thus three different overcomplete dictionaries are considered: *Dirac-Haar* (DH), *Fourier-Dirac (*FD), and *Fourier-Haar* (FH) dictionaries. We chose $d = 32$ (i.e. D =64) in our simulations.

The K atoms from class#*i* are drawn according to a two-sided Gaussian distribution indexed over the atoms (columns) of the sub-dictionaries which we define as follows:

$$p_i(n;\ u_i,\ \sigma_l^i, \sigma_h^i) = \begin{cases} \frac{2\sigma_l^i}{\sigma_l^i + \sigma_h^i}\phi_l(n; u_i,\ \sigma_l^i) & if\ n \leq u_i \\ \frac{2\sigma_h^i}{\sigma_l^i + \sigma_h^i}\phi_h(n; u_i,\ \sigma_h^i) & if\ n > u_i \end{cases} \quad (7)$$

Where:

$$\phi_\mathrm{l}(\mathrm{n};\ \mathrm{u_i},\ \sigma_\mathrm{l}^\mathrm{i}) = \frac{1}{\sqrt{2\pi}\sigma_\mathrm{l}} \exp\left(-\frac{(\mathrm{n}-\mathrm{u_i})^2}{(\sigma_\mathrm{l}^\mathrm{i})^2}\right) \quad (8)$$

$$\phi_h(n; u_i,\ \sigma_h^i) = \frac{1}{\sqrt{2\pi}\sigma_h} exp\left(-\frac{(n-u_i)^2}{(\sigma_h^i)^2}\right) \quad (9)$$

In our simulations we chose K=8, $(\mathrm{u_l} = d/2, \sigma_l^1 = d/4, \sigma_h^1 = 2d/5)$, $(\mathrm{u_l} = 3d/4, \sigma_l^1 = 2d/5, \sigma_h^1 = d/4)$.

We note that the Fourier-Dirac and Fourier-Haar dictionaries are used to generate the training and test samples. The optimum KSVD and dM-KSVD dictionaries calculated from the training samples are used to extract features from the samples.

Table 1.1 and Table 1.2 shows, respectively the performance dM-KSVD1 algorithm for the Fourier-Dirac and Fourier-Haar dictionaries. Table 2.1 and Table 2.2 show the corresponding performance of dM-KSVD2; and Table 3.1 and Table 3.2 show the corresponding performance of KSVD feature extraction procedure.

Figure 1 and Figure 2 show a snapshot of mutual exclusivity values of test samples due to K-DVD and dM-KSVD1 discriminative coding schemes for the case of Fourier-Dirac and Fourier-Haar dictionaries respectively.

From these results we can clearly observe the performance gains due to employing dictionaries due the dM-KSVD algorithms for extracting discriminative features as compared to the transitional pure sparsity based approaches. We also see how the features calculated due to the dM-KSVD1 approach does indeed render the features to be more mutually exclusive as compared to KSVD approach.

We further remark that unlike the dM-KSVD1 algorithm, in dM-KSVD2 discriminative coding scheme there is *no explicit guarantee* (i.e. by virtue of the algorithmic construction of the D-KSVD#2 algorithm) that the mutual exclusivity will decrease as compared to KSVD based sparsity coding scheme—nevertheless, classification performance w.r.t. KSVD based coding schemes can be observed as shown below.

5. DISCUSSION

We have demonstrated how our newly developed concept of *mutual exclusivity* can be deployed in constructing a dictionary Φ that can be used to generate *discriminative codes* that are optimized for classification purposes. We presented two different flavors of DDL algorithms that build upon the well known KSVD algorithm by replacing the sparse coding step by a *discriminative coding step* that exploits the mutual exclusivity operator. We used only the $\mathfrak{M}_1$ operator in our DDL algorithms. Future work includes reducing the computational complexity via advanced convex optimization techniques, and incorporating other mutual exclusivity operators such as $\mathfrak{M}_2$ and $\mathfrak{M}_3$ into DDL algorithms.

12. REFERENCES

[1] R.G. Raj, "An Asymptotically Convex Approach to Discriminative Coding", *IEEE Statistical Signal Processing Workshop*, Ann Arbor, Michigan, August 2012.

[2] R.A. Fischer, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics* 7 (2): 179–188, 1936.

[3] V.N. Vapnik, *Statistical Learning Theory*, 1st edition, Wiley-Interscience, 1998.

[4] V.Y.F. Tan, S. Sanghavi, J.W. Fisher III, and A.S. Willsky, "Learning graphical models for hypothesis testing and classification," *IEEE Trans. Sig. Proc.*, vol. 58, no. 11, Nov 2010.

[5] U. Srinivas, V. Monga, and R.G. Raj, "Exploiting feature dependencies for automatic target recognition via discriminative graphical models," *IEEE Trans. on AES*, *Under Peer Review.*

[6] U. Srinivas, V. Monga, and R.G. Raj, "Automatic target recognition using discriminative graphical models," *Proc. of ICIP*, September 2011.

[7] R.G. Raj, V.C. Chen and R. Lipps, "Analysis of Radar Human Gait Signatures," *IET Signal Processing*, vol.4, no.3, pp.234-244, June 2010.

[8] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, 60, 2, pp. 91-110, 2004.

[9] M. Ranzato, Y. Boureau, and Y. LeCunn, "Sparse feature learning for deep belief networks," *Proc. of NIPS* 2007

	Class1	Class2
Class1	0.9688	0.0312
Class2	0.0781	0.9219

Table 1.1: dM-KSVD1 Performance: For the case where Fourier-Dirac dictionary is used to generate the Classes

	Class1	Class2
Class1	0.9531	0.0469
Class2	0.0703	0.9297

Table 2.1: dM-KSVD2 Performance: For the case where Fourier-Dirac dictionary is used to generate the Classes

	Class1	Class2
Class1	0.8750	0.1250
Class2	0.1094	0.8906

Table 3.1: KSVD Performance: For the case where Fourier-Dirac dictionary is used to generate the Classes

	Class1	Class2
Class1	0.9063	0.0937
Class2	0.1719	0.8281

Table 1.2: dM-KSVD1 Performance: For the case where Fourier-Haar dictionary is used to generate the Classes

	Class1	Class2
Class1	0.9297	0.0703
Class2	0.1719	0.8281

Table 2.2: dM-KSVD2 Performance: For the case where Fourier-Haar dictionary is used to generate the Classes

	Class1	Class2
Class1	0.8291	0.1709
Class2	0.1562	0.8438

Table 3.2: KSVD Performance: For the case where Fourier-Haar dictionary is used to generate the Classes

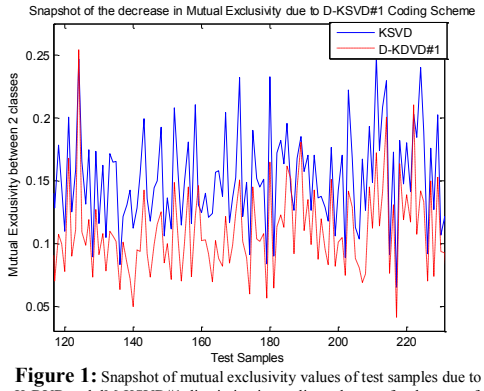

Figure 1: Snapshot of mutual exclusivity values of test samples due to K-DVD and dM-KSVD#1 discriminative coding schemes for the case of *Fourier-Dirac* generative dictionary

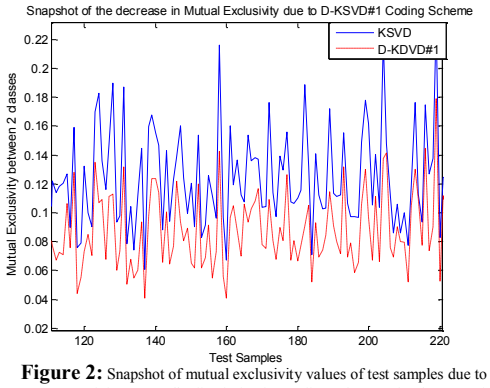

Figure 2: Snapshot of mutual exclusivity values of test samples due to K-DVD and dM-KSVD#1 discriminative coding schemes for the case of *Fourier-Haar* generative dictionary

dM-KSVD1 Algorithm (Type#1 of Discriminative Mutually-Exclusive (dM) KSVD Algorithm):

0) Initialization: Initialize the dictionary matrix $\Phi^{(0)} \in \mathbb{R}^{nxK}$ with l^2 normalized columns. Set J=1.

Repeat until convergence (stopping rule):

1) Discriminative Coding Stage:

– Initialize the codes for each of the elements of the two classes by sparsely coding the data vectors with respect to dictionary $\Phi^{(0)}$ to obtain codes $\{c_i^1\}_{i=1}^N$ and $\{c_i^2\}_{i=1}^N$ for Class#1 and Class#2 respectively

– Discriminatively code samples of Class#1:

For m = 1 to M

- Form the steering vector g_1 for Class#1:
$$g_1(x) = \frac{1}{N}\sum_{t=1}^{N} \mathfrak{M}_1(x, c_i^2)$$
- Discriminative code Update: If mod(m, 2)==1, then solve for all *i*:
$$c_i^1 = \min_c \|x_i^1 - \Phi c\|_2^2 + \lambda\|c\|_1 + \mu g_1(c) \qquad (?)$$
- Steering vector update: If mod(m, 2)==0, then re-calculate the codes for Class#2 based on $\{c_i^1\}_{i=1}^N$ above. Based on this re-calculate the steering vector g_1.

– Discriminatively code samples of Class#2 in a manner analogous to described above

2) Perform KSVD-type Codebook Update:

Given $X = [X_1, X_2]$ *and* $C = [C_1, C_2]$; then, perform the standard KSVD codebook update to obtain $\Phi^{(J)}$

3) Set J = J + 1

Figure#3: dM-KSVD1 Algorithm

dM-KSVD2 Algorithm (Type#2 of Discriminative Mutually-Exclusive (dM) KSVD Algorithm):

0) Initialization: Initialize the dictionary matrix $\Phi^{(0)} \in \mathbb{R}^{nxK}$ with l^2 normalized columns. Set J=1.

Repeat until convergence (stopping rule):

1) Discriminative Coding Stage:

– Initialize the codes for each of the elements of the two classes by sparsely coding the data vectors with respect to dictionary $\Phi^{(0)}$ to obtain codes $\{c_i^1\}_{i=1}^N$ and $\{c_i^2\}_{i=1}^N$ for Class#1 and Class#2 respectively

– Discriminatively code samples of Class#1:

Form the steering vector g_1 for Class#1: $g_1(x) = \frac{1}{N}\sum_{t=1}^{N} \mathfrak{M}_1(x, c_i^2)$

For m = 1 to N

- Discriminatively code all training samples w.r.t. Class#1 steering vector:
$$c_i^{1,1} = \min_c \|x_i^1 - \Phi c\|_2^2 + \lambda\|c\|_1 + \mu g_1(c)$$
$$c_i^{2,1} = \min_c \|x_i^2 - \Phi c\|_2^2 + \lambda\|c\|_1 + \mu g_1(c)$$

– Discriminatively code samples of Class#2 in a manner *analogous to described above w.r.t. steering vector* g_2 for Class#2: $g_2(x) = \frac{1}{N}\sum_{t=1}^{N} \mathfrak{M}_1(x, c_i^1)$

– ***For each i=1:N,*** let c_i^1 be the assigned the sparsest solution between $c_i^{1,1}$ and $c_i^{2,1}$; and similarly for c_i^2.

2) Perform KSVD-type Codebook Update:

Given $X = [X_1, X_2]$ *and* $C = [C_1, C_2]$; then, perform the standard KSVD codebook update to obtain $\Phi^{(J)}$

3) Set J = J + 1

Figure#4: dM-KSVD2 Algorithm