K-LDA: AN ALGORITHM FOR LEARNING JOINTLY OVERCOMPLETE AND DISCRIMINATIVE DICTIONARIES

Mohsen Joneidi[a], *Jamal Golmohammady*[b], *Mostafa Sadeghi*[a], *Massoud Babaie-Zadeh*[a], *Christian Jutten* [c*]

[a]Electrical Engineering Department, Sharif University of Technology, Tehran, IRAN.
[b]Computer Engineering Department, Sharif University of Technology, Tehran, IRAN.
[c]GIPSA-Lab, Grenoble, and Institut Universitaire de France, France.

ABSTRACT

A new algorithm for learning jointly reconstructive and discriminative dictionaries for sparse representation (SR) is presented. While in a usual dictionary learning algorithm like K-SVD only the reconstructive aspect of the sparse representations is considered to learn a dictionary, in our proposed algorithm, which we call K-LDA, the discriminative aspect of the sparse representations is also addressed. In fact, K-LDA is an extension of K-SVD in the case that the class informations (labels) of the training data are also available. K-LDA takes into account these information in order to make the sparse representations more discriminate. It makes a trade-off between the amount of reconstruction error, sparsity, and discrimination of sparse representations. Simulation results on synthetic and hand-written data demonstrate the promising performance of our proposed algorithm.

Index Terms— Dictionary Learning, Singular Value Decomposition, Linear Discriminant Analysis, Discriminative Learning

1. INTRODUCTION

Sparse representation modelling has recently drawn much interest in signal processing community. This is mainly due to the fact that an important variety of signals such as natural images admit sparse representations in terms of some basis functions. This sparsity property has been successfully exploited in many signal processing applications, e.g, image processing [2, 3], video processing [4], and classification tasks [5]. Sparse and overcomplete models were first introduced in [1] for modelling the spatial receptive fields in the human visual system.

Consider a signal in the vector form $\mathbf{y} \in \mathbb{R}^m$ which is going to be sparsely represented as a linear combination of the columns of $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_k]$: $\mathbf{y} = \sum_{i=1}^{k} x_i \mathbf{d}_i = \mathbf{D}\mathbf{x}$. In this context $\mathbf{D}$ is called *dictionary*, and each of its columns is called *atom*. The dictionary is usually over-complete, meaning that $m < k$. By sparse we mean that $\mathbf{x}$, the representation of $\mathbf{y}$ in $\mathbf{D}$, has as few as possible non-zero coefficients. The problem of finding the sparsest representation of a signal in a given dictionary has been extensively studied during the last decade, and numerous algorithms have been proposed [10].

As the role of the dictionary is indisputable to obtain sparse enough representations, its determination and design has been widely investigated during the last few years [11, 12]. Although there are some pre-defined dictionaries which are known to be well-matched to specific classes of signals, e.g., Discrete Cosine Transform (DCT), significant research efforts have shown that *learning* the dictionary using sample signals from the special signal class at hand provide much better performance in various applications including image enhancement, compression, and classification. In dictionary learning (DL) the goal is to adapt the atoms of the dictionary to a number of training signals $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^{N}$ in such a way that each training signal can be sparsely represented based on the learned atoms.

Dictionary learning problem is generally defined as follows

$$(\mathbf{D}, \mathbf{X}) = \underset{\mathbf{D}\in\mathcal{D}, \mathbf{X}\in\mathcal{X}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2, \qquad (1)$$

where $\|.\|_F$ is the Frobenius norm, $\mathcal{D}$ is the admissible dictionary set which is usually defined as the set of all matrices with unit column-norm and $\mathcal{X}$ is the set of matrices X with sparse columns.

To solve (1) an iterative procedure is usually employed in which the objective function is alternatively minimized over one variable ($\mathbf{D}$ or $\mathbf{X}$) while the other is fixed. Minimization over $\mathbf{X}$ with a fixed $\mathbf{D}$ is called the sparse coding stage, while minimization over $\mathbf{D}$ with $\mathbf{X}$ being fixed and equal to the previously found coefficient matrix is called the dictionary update stage. Most dictionary learning algorithms differ mainly in the way they perform the dictionary update stage.

K-Singular Value Decomposition (K-SVD) is a well-

*This work has been partially funded by the Iran National Science Foundation (INSF) under Contract 91004600 and by the European project ERC-2012-AdG320684-CHESS.

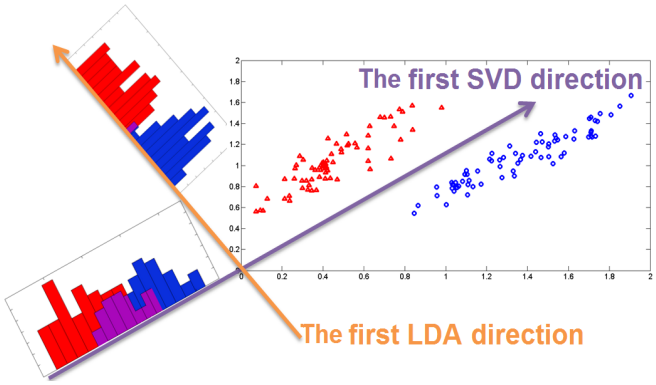

Fig. 1. The first directions found by SVD and LDA for some 2D labelled data. Projection of the data on the found directions are also shown (violet histogram shows an overlap between the two classes).

known dictionary learning algorithm [6]. At the dictionary update stage, K-SVD updates the atoms of the dictionary sequentially, in which to update each atom, the others are kept fixed. Moreover, along with each atom, the non-zero entries of its associated row vector in $\mathbf{X}$ are also updated. This leads to a matrix rank-1 approximation problem, which can be solved by SVD.

In the case of labelled data, i.e., when the training signals belong to different classes, a usual DL problem as can be seen from equation (1), does not incorporate the label information of the data. In fact, a DL algorithm aims to improve just the reconstructive aspect of the representations; not their discrimination capability. A well-known learning-based tool to enhance the discrimination power of the representations is the Linear Discriminative Analysis (LDA). LDA finds directions which by projecting the training data onto them, the discrimination between classes becomes as high as possible. Figure 1 shows the difference of SVD and LDA in the case of some 2D data. Unlike SVD, projection of the data on the rank-1 subspace spanned by the first LDA direction (basis) can discriminate the labelled data. Histograms of the projected data (the representations) are also shown in this figure.

Note that although the goal of K-SVD is not to improve the discrimination of the data, it somewhat enhances the amount of discrimination by converting the data into a high-dimensional space. Figure 2 illustrates this for some 2D data.

In the case of labelled training data for classification, the objective function of (1) should be appropriately modified to take into account these information. To this aim, a discrimination term $\mathcal{Q}(\mathbf{Y}, \mathbf{D}, \mathbf{h})$ should be added to (1) which results in the following problem,

$$(\mathbf{D}, \mathbf{X}) = \underset{\mathbf{D}\in\mathcal{D}, \mathbf{X}\in\mathcal{X}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \mathcal{Q}(\mathbf{Y}, \mathbf{D}, \mathbf{h}) \qquad (2)$$

where, $\mathbf{h} \in \mathbb{R}^N$ contains the labels of the training signals.

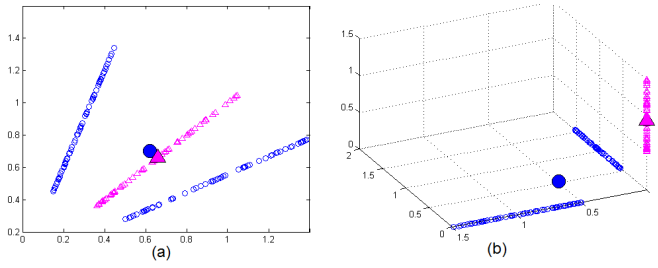

Fig. 2. (a) Some 2D data belonging to two classes. (b) Sparse representations of data in (a) on 3 atoms obtained by K-SVD. The big markers indicate the centroids of the classes.

Some methods define $\mathcal{Q}(\mathbf{Y}, \mathbf{D}, \mathbf{h})$ as the loss function of a classifier [7, 8]. For instance, Discriminative K-SVD (DK-SVD) [7] is an extension of K-SVD which takes into account the label information of the training signals. DK-SVD simultaneously learns a dictionary and a linear classifier which can be used to predict the label of a new test data. This algorithm indeed exploits the fact that the sparse representations of the data have more discrimination power. So, it learns the linear classifier based on these high-dimensional representations of the data.

In this paper we propose a new DL algorithm, called K-LDA, to learn jointly reconstructive and discriminative atoms. Our algorithm is based on adding a discrimination term to the objective function of (1). The proposed algorithm regularizes the level of reconstruction and discrimination of the learned atoms. Note that this differs from DK-SVD in the sense that K-LDA aims to improve the discrimination power of the sparse representations not learning a linear classifier, as DK-SVD does. It will be shown in the experimental results that our proposed algorithm provides higher discrimination than K-SVD and DK-SVD.

The rest of the paper is organized as follows. Section 2 details our proposed method. In this section we first review the Fisher criterion, which is a discriminative measure, and then inspiring by it, we present our proposed DL problem. Section 3 presents the experimental results. Concluding remarks are given in Section 4.

2. PROPOSED METHOD

2.1. Fisher criterion

Fisher criterion is one of the well-known discriminative measures which is also used in LDA. Suppose that the input data belong to C classes with N_i samples being in the class c_i, $i = 1 \ldots C$. Fisher criterion is defined as $\text{tr}\{\mathbf{S}_W^{-1}(\mathbf{Y})\mathbf{S}_B(\mathbf{Y})\}$, in which $\mathbf{S}_W$ and $\mathbf{S}_B$ are the within-class and between-class scatter matrices, respectively, defined as

$$\mathbf{S}_W(\mathbf{Y}) = \sum_{i=1}^{C} \sum_{\mathbf{y}_k \in c_i} (\mathbf{y}_k - \mathbf{m}_i)(\mathbf{y}_k - \mathbf{m}_i)^T$$

$$\mathbf{S}_B(\mathbf{Y}) = \sum_{i=1}^{C} N_c(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T$$

where $\mathbf{m}_i$ and $\mathbf{m}$ are the mean vectors of the ith class and $\mathbf{Y}$, respectively.

LDA learns a projection matrix $\mathbf{W}$ such that $\mathbf{W}^T\mathbf{Y}$ has the largest Fisher criterion (or equivalently, the largest discrimination). To this aim, LDA solves the following problem,

$$\mathbf{W} = \underset{\mathbf{W}}{\operatorname{argmax}} \ \operatorname{tr}\left\{\left(\mathbf{W}^T\mathbf{S}_W(\mathbf{Y})\mathbf{W}\right)^{-1}\left(\mathbf{W}^T\mathbf{S}_B(\mathbf{Y})\mathbf{W}\right)\right\}$$
$$\text{subject to } \forall\, k, j\ (k \neq j): \quad \|\mathbf{w}_k\|_2 = 1, \quad \mathbf{w}_k^T\mathbf{w}_j = 0 \quad (3)$$

where $\mathbf{w}_i$ is the ith column of $\mathbf{W}$.

2.2. The new DL problem

SVD is a tool to obtain an adaptive set of basis functions for training data. The maximum number of basis functions that can be extracted by SVD is equal to dimension of the data. As previously mentioned, K-SVD iteratively uses SVD for obtaining an over-complete dictionary. In the case of labelled data, LDA learns at most $C-1$ basis functions where C is the total number of classes. That is why this method can not be used for learning over-complete dictionaries.

We use $\operatorname{tr}\{\mathbf{D}^T S_W^{-1}(\mathbf{Y})S_B(\mathbf{Y})\mathbf{D}\}$ as an alternative of LDA term. The following expression finds the same solution for $\mathbf{D}$ as LDA method does for $\mathbf{W}$, which is determined by eigenvectors of $S_W^{-1}(\mathbf{Y})S_B(\mathbf{Y})$,

$$\underset{\mathbf{D}}{\operatorname{argmax}} \ \operatorname{tr}\{\mathbf{D}^T\mathbf{S}_W^{-1}(\mathbf{Y})\mathbf{S}_B(\mathbf{Y})\mathbf{D}\} \quad \text{subject to } \forall i: \ \|\mathbf{d}_i\|_2 = 1. \quad (4)$$

From linear algebra lemmas, we know that

$$\operatorname{tr}\{\mathbf{D}^T\mathbf{S}_W^{-1}(\mathbf{Y})\mathbf{S}_B(\mathbf{Y})\mathbf{D}\} = \sum_{i=1}^{k} \mathbf{d}_i^T\mathbf{S}_W^{-1}(\mathbf{Y})\mathbf{S}_B(\mathbf{Y})\mathbf{d}_i \quad (5)$$

We will see that this property help us to derive our proposed algorithm.

Finally, our proposed DL formulation is as follows,

$$\min_{\mathbf{D},\mathbf{X}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 - \lambda_1 \operatorname{tr}\left\{\mathbf{D}^T S_W^{-1}(\mathbf{Y})S_B(\mathbf{Y})\mathbf{D}\right\} \quad \text{subject to } \forall i: \ \|\mathbf{x}_i\|_0 \leq T \text{ and } \forall j \ \|\mathbf{d}_j\|_2 = 1 \quad (6)$$

Note that problem (6) is jointly non-convex in $\mathbf{D}$ and $\mathbf{X}$. To solve it, like a usual DL problem, we use alternating minimization over the two involved variables.

When $\mathbf{D}$ is fixed the discrimination term is constant with respect to $\mathbf{X}$. So, the problem becomes equivalent to performing an ordinary sparse representation. We use Orthogonal Matching Pursuit (OMP) [9] to this aim.

When $\mathbf{X}$ is fixed, we optimize the objective function atom by atom. In other words, for updating $\mathbf{d}_i$ the others, i.e., $\mathbf{d}_j$ for $j \neq i$ are fixed. We define $\mathbf{x}(\Omega)$ as a vector containing those entries of $\mathbf{x}$ that are indexed by Ω, and $\mathbf{E}(:,\Omega)$ as a matrix containing those columns of $\mathbf{E}$ that are indexed by Ω. Assume that we want to update the ith atom, $\mathbf{d}_i$, along with the non-zero entries of $\mathbf{x}_T^i$, the ith row of $\mathbf{X}$. We define $\Omega_i = \left\{j : \ \mathbf{x}_T^i(j) \neq 0\right\}$ as the support of $\mathbf{x}_T^i$.

To update $\mathbf{d}_i$ and $\mathbf{x}_T^i$ the following problem has to be solved

$$\min_{\mathbf{d}_i,\mathbf{x}_T^i} \|\underbrace{\mathbf{Y} - \sum_{j\neq i} \mathbf{d}_j\mathbf{x}_T^j}_{\mathbf{E}_i} - \mathbf{d}_i\mathbf{x}_T^i\|_F^2 - \lambda_1\mathbf{d}_i^T\mathbf{S}_W^{-1}(\mathbf{Y})\mathbf{S}_B(\mathbf{Y})\mathbf{d}_i + \lambda_2\|\mathbf{d}_i\|_2^2 \quad (7)$$

where $\mathbf{E}_i$ is the error matrix associated with $\mathbf{d}_i$. The last term in the objective function was added to take into account the constraint $\|\mathbf{d}_i\|_2 = 1$.

The interpretation of problem (7) is that we want to learn $\mathbf{d}_i$'s in such a way that the reconstruction and the discrimination aspects of all training signals be sufficiently addressed. Like K-SVD, we should update the atom $\mathbf{d}_i$ using only the data that have used it in their representations. In other words, the atom $\mathbf{d}_i$ should provide discrimination for its own training signals (which belongs generally to different classes).

Note that Ω_i is indeed the set of inputs indices that use $\mathbf{d}_i$ in their representations. Also define $\mathbf{Y}_{[i]} = \mathbf{Y}(:,\Omega_i)$, $\mathbf{E}_{[i]} = \mathbf{E}(:,\Omega_i)$, and $\mathbf{x}_{[i]}$ a row vector of length $|\Omega_i|$. Then, the problem of updating $\mathbf{d}_i$ along with the non-zero entries of $\mathbf{x}_T^i$ becomes as follows,

$$\min_{\mathbf{d}_i,\mathbf{x}_{[i]}} \|\mathbf{E}_{[i]} - \mathbf{d}_i\mathbf{x}_{[i]}\|_F^2 - \lambda_1\mathbf{d}_i^T S_W^{-1}(\mathbf{Y}_{[i]})S_B(\mathbf{Y}_{[i]})\mathbf{d}_i + \lambda_2\|\mathbf{d}_i\|_2^2 \quad (8)$$

Setting the derivative with respect to $\mathbf{x}_{[i]}$ equal to zero results in $\mathbf{x}_{[i]} = \mathbf{d}_i^T\mathbf{E}_{[i]}$.

Setting the derivative with respect to $\mathbf{d}_i$ equal to zero results in the following expression

$$-\mathbf{E}_{[i]}\mathbf{x}_{[i]}^T + \mathbf{d}_i\mathbf{x}_{[i]}\mathbf{x}_{[i]}^T - \lambda_1 S_W^{-1}(\mathbf{Y}_{[i]})S_B(\mathbf{Y}_{[i]})\mathbf{d}_i + \lambda_2\mathbf{d}_i = 0 \quad (9)$$

By re-arranging the above equation we reach to the following equation

$$\mathbf{A}\mathbf{d}_i = \beta\mathbf{d}_i \quad (10)$$

where,

$$\begin{cases} \beta = \mathbf{x}_{[i]}\mathbf{x}_{[i]}^T + \lambda_2 \\ \mathbf{A} = \mathbf{E}_{[i]}\mathbf{E}_{[i]}^T + \lambda_1 S_W^{-1}(\mathbf{Y}_{[i]})S_B(\mathbf{Y}_{[i]}) \end{cases} \quad (11)$$

This is an eigen-decomposition problem. The solution for $\mathbf{d}_i$ is the first left-singular vector of the matrix $\mathbf{A}$. In other words, if the SVD of $\mathbf{A}$ be $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$, then $\mathbf{d}_i$ is the first column of $\mathbf{U}$.

Algorithm 1 shows a description of the proposed algorithm. If we set $\lambda_1 = 0$ the LDA term will be removed and then K-LDA will be simplified to the K-SVD algorithm, because eigenvectors of $\mathbf{E}$ are equal to eigenvectors of $\mathbf{E}\,\mathbf{E}^{\mathbf{T}}$.

Algorithm 1 K-LDA dictionary learning

Task: Learning an overcomplete, reconstructive and discriminative dictionary
inputs $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^N$, $\mathbf{h}$, λ
Initialization: set $t = 0$, $\mathbf{D}^{(t)} = \mathbf{D}^0$
The main loop: Repeat until convergence:
Sparse Approximation: $\mathbf{X}^{(t)}$ =OMP($\mathbf{Y}$, $\mathbf{D}^{(t)}$)
Dictionary Update: Set $\mathbf{D} = \mathbf{D}^{(t)}$ and $\mathbf{X} = \mathbf{X}^{(t)}$, $\mathbf{E} = \mathbf{Y} - \mathbf{D}\mathbf{X}$
for $i = 1, \ldots, K$ **do**
    $\mathbf{E}_i = \mathbf{E} + \mathbf{d}_i \mathbf{x}_T^i$
    $\mathbf{E}_{[i]} = \mathbf{E}_i(:, \Omega_i)$ where $\Omega_i = \{j : \mathbf{x}_T^i(j) \neq 0\}$
    $\mathbf{x}_{[i]} = \mathbf{x}_T^i(\Omega_i)$
    $\mathbf{Y}_{[i]} = \mathbf{Y}(:, \Omega_i)$
    $\mathbf{A} = \mathbf{E}_{[i]}\mathbf{E}_{[i]}^T + \lambda_1 S_W^{-1}(\mathbf{Y}_{[i]}) S_B(\mathbf{Y}_{[i]})$
    $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$
    $\mathbf{d}_i = \mathbf{U}(:, 1)$
    $\mathbf{x}_{[i]} = \mathbf{d}_i^T \mathbf{E}_{[i]}$
end for
Set $t = t + 1$, $\mathbf{D}^{(t)} = \mathbf{D}$, and go back to the sparse representation stage

3. EXPERIMENTAL RESULTS

In this section we evaluate the efficiency of the proposed algorithm in the case of synthetic labelled data and images of handwritten digits. The results where compared with K-SVD and DK-SVD algorithms.

Synthetic labelled data

In this experiment, two sets of data were randomly generated as $\mathbf{Y}_i = \mathbf{D}_i \mathbf{X}_i$, where $\mathbf{D}_i \in \mathbb{R}^{64\times 10}$ are normalized-columns matrices with *i.i.d.* zero-mean, unit-variance Gaussian entries, and $\mathbf{X}_i \in \mathbb{R}^{10\times 2000}$, $i = 1, 2$ are random sparse-column matrices having at most 5 non-zeros in each column. The training data were then set as $\mathbf{Y} = [\mathbf{Y}_1 \mathbf{Y}_2]$ in which the data in $\mathbf{Y}_1$ constitutes the first class and those in $\mathbf{Y}_2$ constitutes the second class. We then applied $\mathbf{Y}$ to K-SVD, DK-SVD and our proposed algorithm to learn an over-complete dictionary of size 64×128 and sparsity of 5.

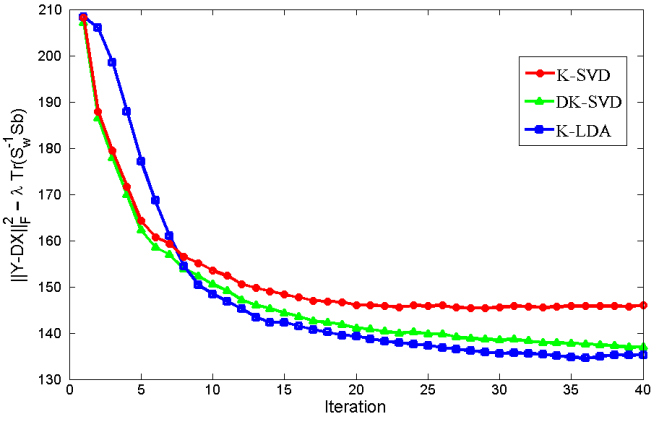

Fig. 3. Values of $\|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 - \lambda \text{tr}(\mathbf{S}_w^{-1}\mathbf{S}_b)$ versus iteration for 4000 synthetic data with $\lambda = 0.2$ in K-LDA.

Figure 3 shows the values of $||\mathbf{Y} - \mathbf{D}\mathbf{X}||_F^2 - \lambda \text{tr}(\mathbf{S}_w^{-1}\mathbf{S}_b)$ along iterations (averaged over 100 trials) for these three algorithms. As can be seen, the proposed algorithm achieved a lower value.

Hand-written digits

Dictionary learning for hand-written digits of size 28×28 was simulated in this experiment. We first performed feature extraction on these raw data by applying the DCT transform on each data and selecting the 15×15 top coefficients. We then converted them into equivalent vectors of length 225, and reduced their dimensionality to 100 by applying Principal Component Analysis (PCA). For each digit about 4000 data were available, from which 500 data were used for learning the dictionary. We finally applied the obtained training data to the algorithms to learn an over-complete dictionary of size 100×160 with sparsity of 5.

Figure 4 shows the amounts of discrimination of the representations provided by the algorithms during the iterations in which the parameters of DK-SVD and K-LDA have been set such that their reconstruction errors be equal to that of K-SVD. This figure again says that K-LDA has made more improvement in the discrimination of the representations compared to DK-SVD and especially the K-SVD. As the dominant computational burden in all algorithms is due to performing eigen-decomposition, their running times are approximately the same and depend on the needed iterations to converge.

Figure 5 shows the effect of λ on discrimination in K-LDA. As can be seen, the discrimination is improved with increasing the value of λ.

Table 1 shows successful classification rates for a set of test data (3500 data for each digit). Support Vector Machine (SVM) was used as a linear classifier in the sparse domain. As can be seen, DK-SVD and K-LDA have performed similarly,

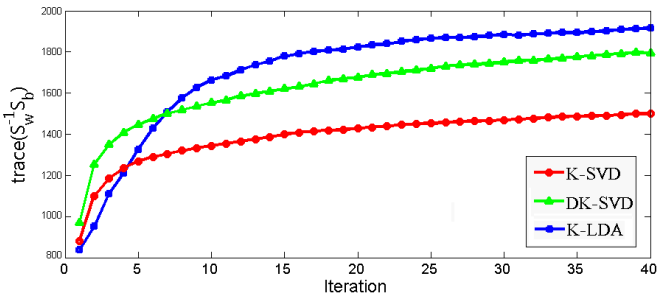

Fig. 4. Discrimination ($\text{tr}(\mathbf{S}_W^{-1}\mathbf{S}_B)$) versus iteration performed by the 3 algorithms.

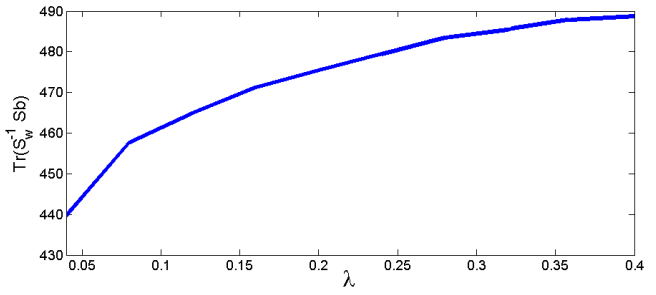

Fig. 5. Effect of λ on discrimination.

with K-LDA being slightly better.

4. CONCLUSION

We introduced a new algorithm for learning jointly discriminative and reconstructive dictionaries. The proposed algorithm, which we called K-LDA, exploits the information about the labels of training data directly in updating the dictionary atoms. K-LDA is able to make a trade-off among the amount of sparsity, reconstruction error and discrimination of the sparse representation of the data. A main difference between K-LDA and other discriminative dictionary learning algorithms like DK-SVD is that K-LDA aims to make the data in the transformed sparse domain as discriminative as possible, while the other algorithms just learn a classifier on the sparse representations of the data. As was shown in the simulation results, K-LDA outperforms K-SVD and DK-SVD in providing both good reconstruction error and discrimination.

Table 1. Successful classification rates versus number of training data for K-SVD, DK-SVD and K-LDA.

Number of training data	100	200	300	400	500
K-SVD	90.25	91.62	92.70	93.51	94.04
DK-SVD	91.88	93.14	93.94	94.65	94.90
K-LDA	92.21	93.39	94.22	94.71	95.07

5. REFERENCES

[1] B. A. Olshausen, and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?," *Vision research*, vol. 37, pp. 3311–3325, 1997.

[2] M. Elad, and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing,* vol. 15, no. 12, pp. 3736–3745, 2006.

[3] J. Mairal, G. Sapiro, and M. Elad, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing,* vol. 17, no. 1, pp. 53–69, 2007.

[4] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *Multiscale Modeling and Simulation*, vol. 7, no. 1, pp. 214–241, 2008.

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

[6] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representations," *IEEE Trans. on Signal Processing,* vol. 54, no. 11, pp. 4311–4322, 2006.

[7] Q. Zhang, and B. Baoxin "Discriminative K-SVD for dictionary learning in face recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2691–2698, 2010.

[8] Z. Jiang, Z. Lin, and LS. Davis "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1697–1704, 2011.

[9] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. on Inf. Theory*, vol. 50, pp. 2231–2242, 2004.

[10] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[11] M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, "Dictionary learning for sparse representation: A novel approach," *IEEE Signal Processing Letters*, vol. 20, no. 12, pp. 1195–1198, December 2013.

[12] I. Tosic and P. Frossard, "Dictionary learning: What is the right representation for my signal?," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 27–38, 2011.