Kernelized Supervised Dictionary Learning

Mehrdad J. Gangeh*, *Member, IEEE,* Ali Ghodsi, and Mohamed S. Kamel, *Fellow, IEEE*

***Abstract*—In this paper, we propose supervised dictionary learning (SDL) by incorporating information on class labels into the learning of the dictionary. To this end, we propose to learn the dictionary in a space where the dependency between the signals and their corresponding labels is maximized. To maximize this dependency, the recently introduced Hilbert Schmidt independence criterion (HSIC) is used. One of the main advantages of this novel approach for SDL is that it can be easily kernelized by incorporating a kernel, particularly a data-dependent kernel such as normalized compression distance, into the formulation. The learned dictionary is compact and the proposed approach is fast. We show that it outperforms other unsupervised and supervised dictionary learning approaches in the literature, using real-world data.**

***Index Terms*—Pattern recognition and classification, classification methods, non-parametric methods, dictionary learning, HSIC, supervised learning.**

I. Introduction

DICTIONARY learning and sparse representation (DLSR) are two closely-related topics that have roots in the decomposition of signals to some predefined bases, such as the Fourier transform. However, what makes DLSR distinct from the representation using predefined bases is that first, the bases are learned here from the data, and second, only a few components in the dictionary are needed to represent the data (sparse representation). This latter attribute can also be seen in the decomposition of signals using some predefined bases such as wavelets [1].

The concept of dictionary learning and sparse representation originated in different communities attempting to solve different problems, which are given different names. Some of them are: sparse coding (SC), which was originated by neurologists as a model for simple cells in mammalian primary visual cortex [2]; independent component analysis (ICA), which was originated by researchers in signal processing to estimate the underlying hidden components of multivariate statistical data (refer to [3] for a review of ICA); least absolute shrinkage and selection operator (*lasso*), which was originated by statisticians to find linear regression models when there are many more predictors than samples, where some constraints have to be considered to fit the model. In the *lasso*, one of the constraints introduced by Tibshirani was the ℓ_1 norm that led to sparse coefficients in the linear regression model [4]. Another technique which also leads to DLSR is nonnegative matrix factorization (NNMF), which aimed to decompose a matrix to two nonnegative matrices, one of which can be considered to be the dictionary, and the other the coefficients [5]. In NNMF, usually both the dictionary and coefficients are sparse [5], [6]. This list is not complete, and there are variants for each of the above techniques, such as blind source separation (BSS) [7], compressed sensing [8], basis pursuit (BP) [9], and orthogonal matching pursuit (OMP) [10], [11]. It is beyond the scope of this paper to include the description of all these techniques (interested readers can refer to [12]–[14] for a review on dictionary learning and sparse representation).

The main results of all these research efforts is that a class of signals with sparse nature, such as images of natural scenes, can be represented using some *primitive elements* that form a dictionary, and that each signal in this class can be represented by using only a few elements in the dictionary, i.e., by a sparse representation. In fact, there are, at least, two ways in the literature to exploit sparsity [15]: first, using a linear/nonlinear combination of some predefined bases, e.g., wavelets [1]; second, using primitive elements in a learned dictionary, such as the techniques employed in SC or ICA. This latter approach is our focus in this paper and has led to state-of-the-art results in various applications such as texture classification [16], [17], face recognition [18]–[20], image denoising [21], [22], etc.

We may categorize the various dictionary learning with sparse representation approaches proposed in the literature in different ways: one where the dictionary consists of predefined or learned bases as stated above, and the other based on the model used to learn the dictionary and coefficients. These models can be *generative* as used in the original formulation of SC [2], ICA [3], and NNMF [5]; *reconstructive* as in the *lasso* [4]; or discriminative such as SDL-D (supervised dictionary learning-discriminative) in [15]. The two former approaches do not consider the class labels in building the dictionary, while the last one (i.e., the discriminative one) does. In other words, we state that dictionary learning can be performed unsupervised or supervised, with the difference that in the latter, the class labels in the training set are used to build a more discriminative dictionary for the particular classification task in hand.

In this paper, we propose a novel supervised dictionary learning (SDL) by incorporating information on class labels into the learning of the dictionary. The dictionary is learned in a space where the dependency between the data and their corresponding labels is maximized. We propose to maximize this dependency by using the recently introduced Hilbert Schmidt

This research was funded, in part, by the Natural Sciences and Engineering Research Council (NSERC) of Canada under Canada Graduate Scholarship (CGS D3-378361-2009) and also by Ontario Graduate Scholarship.

M. J. Gangeh is with the Department of Medical Biophysics, University of Toronto, Toronto, ON, Canada and also with the Department of Radiation Oncology, Sunnybrook Health Sciences Center, Toronto, ON, Canada. At the time of doing this research, he was with the Center for Pattern Analysis and Machine Intelligence, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada e-mail: mehrdad.gangeh@utoronto.ca.

A. Ghodsi is with the Department of Statistics and Actuarial Science, University of Waterloo, Waterloo, ON, Canada e-mail: aghodsib@uwaterloo.ca.

M. S. Kamel is with the Center for Pattern Analysis and Machine Intelligence, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada e-mail: mkamel@pami.uwaterloo.ca.

independence criterion (HSIC) [23], [24]. The dictionary is then learned in this new space. Although supervised dictionary learning has been proposed by others, as will be reviewed in the next section, this work is different from the others in the following aspects:

1) The formulation is simple and straightforward;
2) The proposed approach introduces a closed form formulation for the computation of the dictionary. This is different from other approaches, in which the computation of dictionary and sparse coefficients has to be iteratively and often alternately performed, which causes high computational load;
3) The proposed approach also leads to separable problem on the minimization of the coefficients, which can be solved in closed form using soft thresholding. This further improves the performance of the proposed algorithm in terms of speed;
4) The approach is very efficient in terms of dictionary size (compact dictionary). Our results show that the proposed dictionary can produce significantly better results than other supervised dictionary methods at small dictionary sizes. An important special case is when the dictionary size is smaller than the dimensionality of data. This turns the learning of a dictionary whose size is usually larger than the dimensionality of the data, i.e., an *overcomplete* dictionary, into the learning of a *subspace*;
5) The proposed approach can be easily kernelized by incorporating a kernel into the formulation. Data-dependent kernels based on, e.g., normalized compression distance (NCD) [25], [26], can be used in this kernelized SDL to further improve the discrimination power of the designed system. To the best of our knowledge, no other kernelized SDL approach has been proposed in the literature yet, and none of the proposed SDLs in the literature can be kernelized in a straightforward way.

The organization of the rest of the paper is as follows: in Section II, we review the current SDL approaches in the literature and their shortcomings. Then we review the mathematical background and the formulation for proposed approach in Section III. The experimental setup and results are presented in Sections IV, followed by discussion and conclusion in Section V.

II. Background and Related Work

In this section, we provide an overview on the dictionary learning and sparse representation, and a brief review of recent attempts on making the approach more suitable for classification tasks.

A. Dictionary Learning and Sparse Representation

Considering a finite training set of signals $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{p\times n}$, where p is the dimensionality and n is the number of data samples, according to classical dictionary learning and sparse representation (DLSR) techniques (refer to [12] and [13] for a recent review on this topic), these signals can be represented by a linear decomposition over a few dictionary atoms by minimizing a loss function as given below

$$L(\mathbf{X}, \mathbf{D}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} l(\mathbf{x}_i, \mathbf{D}, \boldsymbol{\alpha}), \tag{1}$$

where $\mathbf{D} \in \mathbb{R}^{p\times k}$ is the dictionary of k atoms, and $\boldsymbol{\alpha} \in \mathbb{R}^{k\times n}$ are the coefficients.

This loss function can be defined in various ways based on the application in hand. However, what is common in DLSR literature is to define the loss function L as the reconstruction error in a mean-squared sense, with a sparsity-inducing function ψ as a regularization penalty to ensure the sparsity of coefficients. Hence, (1) can be written as

$$L(\mathbf{X}, \mathbf{D}, \boldsymbol{\alpha}) = \min_{\mathbf{D},\boldsymbol{\alpha}} \frac{1}{2}\|\mathbf{X} - \mathbf{D}\boldsymbol{\alpha}\|_{\mathrm{F}}^2 + \lambda\psi(\boldsymbol{\alpha}), \tag{2}$$

where subscript F indicates the Frobenius norm and λ is the regularization parameter that affects the number of nonzero coefficients.

An intuitive measure of sparsity is ℓ_0 norm, which indicates the number of nonzero elements in a vector[1]. However, the optimization problem obtained from replacing sparsity-inducing function ψ in (2) with ℓ_0 is nonconvex, and the problem is NP-hard (refer to [13] for a recent comprehensive discussion on this issue). There are two main proposed approximate solutions to overcome this problem: the first is based on greedy algorithms, such as the well-known orthogonal matching pursuit (OMP) [10], [11], [13]; the second works by approximating a highly discontinuous ℓ_0 norm by a continuous function such as the ℓ_1 norm. This leads to an approach, which is widely known in the literature as *lasso* [4] or *basis pursuit* (BP) [9], and (2) converts to

$$L(\mathbf{X}, \mathbf{D}, \boldsymbol{\alpha}) = \min_{\mathbf{D},\boldsymbol{\alpha}} \sum_{i=1}^{n} \left(\tfrac{1}{2}\|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right). \tag{3}$$

In (3), the main optimization goal for computation of the dictionary and sparse coefficients is minimizing the reconstruction error in the mean-squared sense. While this works well in applications where the primary goal is to reconstruct signals as accurately as possible, such as in denoising, image inpainting, and coding, it is not the ultimate goal in classification tasks [27], as discriminating signals is more important here. Hence, recently, there have been several attempts to include category information in computing either dictionary, coefficients, or both. In the following subsection, we will provide a brief overview of proposed supervised dictionary learning approaches in the literature. To this end, we will try to categorize the proposed approaches into five different categories, while we admit that this taxonomy of approaches is not unique and could be done differently.

B. Supervised Dictionary Learning in Literature

As mentioned in the previous subsection, (3) provides a reconstructive formulation for computing the dictionary and sparse coefficients, given a set of data samples. Although

[1] ℓ_0 norm of vector $\mathbf{x}$ is defined as $\|\mathbf{x}\|_0 = \#\{i : x_i \neq 0\}$.

the problem is not convex on both dictionary $\mathbf{D}$ and coefficients $\boldsymbol{\alpha}$, this optimization problem is convex if it is solved iteratively and alternately on these two unknowns. Several fast algorithms have recently been proposed for this purpose, such as K-SVD [28], online learning [29], and cyclic coordinate descent [30]. However, none of these approaches takes into account the category information for learning either the dictionary or the coefficients.

The first and simplest approach to include category information in DLSR is computing one dictionary per class, i.e., using the training samples in each class to compute part of the dictionary, and then composing all these partial dictionaries into one. Perhaps the earliest work in this direction is the so-called texton-based approach [17], [31], [32]. In this approach, *k*-means is applied to the training samples in each class, and the *k* cluster centers computed are considered as the dictionary for this class. These partial dictionaries are eventually composed into one dictionary. In [19], the training samples are used as the dictionary in face recognition and hence, effectively falls in the same category as training one dictionary per class. However, no actual training is performed here, and the whole training samples are used directly in the dictionary. Using the training samples as dictionary yields a very large and possibly inefficient dictionary due to noisy training instances. To obtain a smaller dictionary, Yang *et al.* proposed learning a smaller dictionary for each class called a *metaface* (the proposed approach was in a face recognition application, but it is general and can be used in any application) and then compose them into one dictionary [33]. One major drawback of this approach is that the training samples in one class are used for computing the atoms in the dictionary, irrespective of the training samples form other classes. This means that if training samples across classes have some common properties, these shared properties cannot be learned in common in the dictionary. Ramirez *et al.* proposed overcoming this problem by including an incoherence term in (3) to encourage independency of dictionaries from different classes, while still allowing for different classes to share features [34]. The main drawback of all approaches in this first category of SDL is that they may lead to a very large dictionary, as the size of the composed dictionary grows linearly with the number of classes.

The second category of SDL approaches learn a very large dictionary unsupervised in the beginning, then merge the atoms in the dictionary by optimizing an objective function that takes into account the category information. One major work in literature in this direction is based on the information bottleneck that iteratively merges two dictionary atoms that cause the smallest decrease in the mutual information between dictionary atoms and class labels [35]. Another major work is based on merging two dictionary atoms so as to minimize the loss of mutual information between histogram of dictionary atoms, over signal constituents, e.g., image patches, and class labels [36]. One main drawback of this category of SDL is that the reduced dictionary obtained usually performs at most the same as the original one hence, since the initial dictionary is learned unsupervised (although due to its large size it includes almost all possible atoms that helps to improve the performance of classification task) the consecutive pruning stage is inefficient in terms of computational load. This can be significantly improved by finding a discriminative dictionary from the beginning.

The third category of SDL, which is based on several research works published in [15], [37]–[41] can be considered a major leap in SDL. In this category, the classifier parameters and dictionary are learned in a joint optimization problem. Although this idea is more sophisticated than the previous two, its major disadvantage is that the optimization problem is nonconvex and complex. If it is done alternately between dictionary learning and classifier parameters learning, it is quite likely that they will become stuck in local minima. On the other hand, due to the complexity of the problem, except for the bilinear classifier in [15], other papers only consider linear classifiers, which is usually too simple to solve difficult problems, and can only be successful in simple classification tasks as shown in [15]. In [38], Zhang and Li propose a technique called discriminative K-SVD (DK-SVD). DK-SVD truly jointly learns the classifier parameters and dictionary, without alternating between these two steps. This prevents the possibility of getting stuck in local minima. However, only linear classifiers are considered in DK-SVD, which may lead to poor performance in difficult classification tasks. Another major problem with the approaches in this category of SDL is that there exist many parameters involved in the formulation, which are hard and time-consuming to tune (see for example [15], [41]).

The fourth category of SDL approaches include the category information in the learning of the dictionary. This is done, for example, by minimizing the information loss due to predicting labels from a supervised dictionary learned instead of original training data samples (this approach is known as *info-loss* in the SDL literature) [42], or by deploying extremely randomized decision forests [43]. This latter approach can also fall in the second category of SDLs, as it seems that it starts from a very large dictionary using random forests, and tries to prune it later to conclude with a smaller dictionary. The same just as in the previous category of SDL, the info-loss approach has the major drawback that it may stuck in local minima. This is mainly because the optimization has to be done iteratively and alternately on two updates, as there is no closed form solution for the approach.

The fifth category of SDLs include class category in learning the coefficients [27] or in learning both dictionary and coefficients [20], [44]. Supervised coefficient learning in all these papers [20], [27], [44] has been performed more or less in the same way using Fisher discrimination criterion [45], i.e., by minimizing the within-class covariance of coefficients and at the same time maximizing their between-class covariance. As for the dictionary, while [27] uses predefined bases, [20] proposes a discriminative fidelity term that encourages learning dictionary atoms of one class from the training samples of the same class, and at the same time penalizes their learning by the training samples from other classes. The joint optimization problem due to Fisher discrimination criterion on the coefficients and the discriminative fidelity term on the dictionary proposed in [20] is not convex, and has to be solved alternately and iteratively between these two terms until it

converges. However, there is no guarantee in this approach to find the global minimum. Also, it is not clear whether the improvement obtained in classification by including Fisher discriminant criterion on coefficients justifies the additional computation load imposed on the learning, as there is no comparison provided in [20] on the classification with and without including supervision on coefficients.

In next section, we explain the mathematical formulation for our proposed approach, which we believe belongs to the fourth category of SDLs explained above, i.e., including category information to learn a supervised dictionary.

III. Methods

To incorporate the category information into the dictionary learning, we propose to decompose the signals using some learned bases that represent them in a space where the dependency between the signals and their corresponding class labels is maximized. To this end, we need a(n) (in)dependency test measure between two random variables. Here, we propose to use Hilbert-Schmidt independence criterion (HSIC) as the (in)dependency measure. In this section, we first describe HSIC, and then provide the formulation for our proposed supervised dictionary learning (SDL) approach. Subsequently, kernelized SDL is formulated that enables embedding kernels, including data-dependent ones, into the proposed SDL. This can significantly improve the discrimination power of the designed dictionary, which is essential in difficult classification tasks, as will be shown in our experiments in Subsection IV-E later.

A. Hilbert Schmidt Independence Criterion

There are several techniques in the literature to measure the (in)dependence of random variables, such as mutual information [46] and Kullback-Leibler (KL) divergence [47]. In addition to these measures, there has recently been great interest in measuring (in)dependency using criteria based on functions in reproducing kernel Hilbert spaces (RKHSs). Bach and Jordan were those who first accomplished this, by introducing kernel dependence functionals that significantly outperformed alternative approaches [48]. Later, Gretton *et al.* proposed another kernel-based approach called the Hilbert-Schmidt independence criterion (HSIC) to measure the (in)dependence of two random variables $\mathcal{X}$ and $\mathcal{Y}$ [23]. Since its introduction, the HSIC has been used in many applications, including feature selection [49], independent component analysis [50], and sorting/matching [51].

One can derive HSIC as a measure of (in)dependence between two random variables $\mathcal{X}$ and $\mathcal{Y}$ using two different approaches: first by computing the Hilbert-Schmidt norm of the cross-covariance operators in RKHSs as shown in [23], [24]; or second, by computing maximum mean discrepancy (MMD) of two distributions mapped to a high dimensional space, i.e., computed in RKHSs [52], [53]. We believe that this latter approach is more straightforward and hence, use it to describe HSIC.

Let $\mathcal{Z} := \{(\mathbf{x}_1, \mathbf{y}_1,), ..., (\mathbf{x}_n, \mathbf{y}_n)\} \subseteq \mathcal{X} \times \mathcal{Y}$ be n independent observations drawn from $p := P_{\mathcal{X}\times\mathcal{Y}}$. To investigate whether $\mathcal{X}$ and $\mathcal{Y}$ are independent, we need to determine whether distribution p factorizes, i.e., whether p is the same as $q := P_{\mathcal{X}} \times P_{\mathcal{Y}}$.

The mean of distributions are defined as follows

$$\mu[P_{\mathcal{X}\times\mathcal{Y}}] := \mathbf{E}_{xy}[v((x,y),.)], \quad (4)$$

$$\mu[P_{\mathcal{X}} \times P_{\mathcal{Y}}] := \mathbf{E}_x\mathbf{E}_y[v((x,y),.)], \quad (5)$$

where $\mathbf{E}_{xy}$ is the expectation over $(x,y) \sim P_{\mathcal{X}\times\mathcal{Y}}$ and kernel $v((x,y),(x',y'))$ is defined in RKHS over $\mathcal{X} \times \mathcal{Y}$. By computing the mean of distributions p and q in RKHS, we effectively take into account higher order statistics than the first order, by mapping these distributions to a high-dimensional feature space. Hence, we can use $\text{MMD}(p,q) := \|\mu[P_{\mathcal{X}\times\mathcal{Y}}] - \mu[P_{\mathcal{X}} \times P_{\mathcal{Y}}]\|_2$ as a measure of (in)dependence of the random variables $\mathcal{X}$ and $\mathcal{Y}$. The higher the value of MMD, the closer the two distributions p and q and hence, the more dependent are random variables $\mathcal{X}$ and $\mathcal{Y}$.

Now suppose that $\mathcal{H}$ and $\mathcal{G}$ are two RKHSs in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Hence, by the Riesz representation theorem, there are feature mappings $\phi(x) : \mathcal{X} \to \mathbb{R}$ and $\psi(y) : \mathcal{Y} \to \mathbb{R}$ such that $k(x,x') = \langle\phi(x), \phi(x')\rangle_{\mathcal{H}}$ and $l(y,y') = \langle\psi(y), \psi(y')\rangle_{\mathcal{G}}$. Moreover, suppose that $v((x,y),(x',y')) = k(x,x')l(y,y')$, i.e., the RKHS is a direct product of $\mathcal{H} \otimes \mathcal{G}$ of the RKHSs on $\mathcal{X}$ and $\mathcal{Y}$. Then $\text{MMD}(p,q)$ can be written as

$$\begin{aligned}\text{MMD}^2(p,q) &= \|\mathbf{E}_{xy}[k(x,.)l(y,.)] \\ &\quad - \mathbf{E}_x[k(x,.)]\mathbf{E}_y[l(y,.)]\|_2^2 \\ &= \mathbf{E}_{xy}\mathbf{E}_{x'y'}[k(x,x')l(y,y')] \\ &\quad - 2\mathbf{E}_x\mathbf{E}_y\mathbf{E}_{x'y'}[k(x,x')l(y,y')] \\ &\quad + \mathbf{E}_x\mathbf{E}_y\mathbf{E}_{x'}\mathbf{E}_{y'}[k(x,x')l(y,y')]. \quad (6)\end{aligned}$$

This is exactly the HSIC, and equivalent to the Hilbert-Schmidt norm of the cross-covariance operator in RKHSs [23].

For practical purposes, HSIC has to be estimated using a finite number of data samples. Considering $\mathcal{Z} := \{(\mathbf{x}_1, \mathbf{y}_1,), ..., (\mathbf{x}_n, \mathbf{y}_n)\} \subseteq \mathcal{X} \times \mathcal{Y}$ as n independent observations drawn from $p := P_{\mathcal{X}\times\mathcal{Y}}$, an empirical estimate of HSIC is defined as follows [23]

$$\text{HSIC}(\mathcal{Z}) = \frac{1}{(n-1)^2}\text{tr}(\mathbf{KHLH}), \quad (7)$$

where tr is the trace operator, $\mathbf{H}, \mathbf{K}, \mathbf{L} \in \mathbb{R}^{n\times n}, K_{i,j} = k(x_i, x_j), L_{i,j} = l(y_i, y_j)$, and $\mathbf{H} = \mathbf{I} - n^{-1}\mathbf{e}\mathbf{e}^\top$ ($\mathbf{I}$ is the identity matrix, and $\mathbf{e}$ is a vector of n ones, and hence, $\mathbf{H}$ is the centering matrix). It is important to note that according to (7), to maximize the dependency between two random variables $\mathcal{X}$ and $\mathcal{Y}$, the empirical estimate of HSIC, i.e., $\text{tr}(\mathbf{KHLH})$ should be maximized.

B. Proposed Supervised Dictionary Learning

To formulate our proposed SDL, we start from the reconstruction error given in (3). Let there be a finite training set of n data points, each of which consists of p features, i.e., $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{p\times n}$. We further assume that features in data samples are centered, i.e., their mean is removed and hence, each row of $\mathbf{X}$ sums to zero. We address the problem of finding a linear decomposition of data $\mathbf{X} \in \mathbb{R}^{p\times n}$ using

some bases $\mathbf{U} \in \mathbb{R}^{p\times k}$ such that the reconstruction error is minimum in the mean-squared sense, i.e.,

$$\min_{\mathbf{U},\mathbf{v}_i} \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{U}\mathbf{v}_i\|_2^2, \tag{8}$$

where $\mathbf{v}_i$ is the vector of k reconstruction coefficients in the subspace defined by $\mathbf{U}^\top\mathbf{X}$. We can rewrite (8) in matrix form as follows

$$\min_{\mathbf{U},\mathbf{V}} \|\mathbf{X} - \mathbf{U}\mathbf{V}\|_{\mathrm{F}}^2, \tag{9}$$

where $\mathbf{V} \in \mathbb{R}^{k\times n}$ is the matrix of coefficients. Since both $\mathbf{U}$ and $\mathbf{V}$ are unknown, this problem is ill-posed and does not have a unique solution unless we impose some constraints on the matrix $\mathbf{U}$. If we, for example, assume that the columns of $\mathbf{U}$ are orthonormal, i.e., $\mathbf{U}^\top\mathbf{U} = \mathbf{I}$, (9) can be written as a constrained optimization problem as follows

$$\begin{aligned} \min_{\mathbf{U},\mathbf{V}} \quad & \|\mathbf{X} - \mathbf{U}\mathbf{V}\|_{\mathrm{F}}^2. \\ \text{s.t.} \quad & \mathbf{U}^\top\mathbf{U} = \mathbf{I} \end{aligned} \tag{10}$$

To further investigate the optimization problem in (10), we assume that the matrix $\mathbf{U}$ is fixed, and find the optimum matrix of coefficients $\mathbf{V}$ in terms of $\mathbf{X}$ and $\mathbf{U}$ by taking the derivative of the objective function given in (10) in respect to $\mathbf{V}$

$$\begin{aligned} \frac{\partial}{\partial \mathbf{V}} \|\mathbf{X} - \mathbf{U}\mathbf{V}\|_{\mathrm{F}}^2 &= \frac{\partial}{\partial \mathbf{V}} \mathrm{tr}[(\mathbf{X} - \mathbf{U}\mathbf{V})^\top(\mathbf{X} - \mathbf{U}\mathbf{V})] \\ &= \frac{\partial}{\partial \mathbf{V}} [\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) - 2\mathrm{tr}(\mathbf{X}^\top\mathbf{U}\mathbf{V}) \\ &\quad + \mathrm{tr}(\mathbf{V}^\top\mathbf{U}^\top\mathbf{U}\mathbf{V})] \\ &= -2\mathbf{U}^\top\mathbf{X} + 2\mathbf{U}^\top\mathbf{U}\mathbf{V}. \end{aligned}$$

Equating the above derivative to zero and knowing that $\mathbf{U}^\top\mathbf{U} = \mathbf{I}$, we obtain

$$\mathbf{V} = \mathbf{U}^\top\mathbf{X}. \tag{11}$$

By plugging the $\mathbf{V}$ found in (11) into the objective function of (10) we obtain

$$\begin{aligned} \min_{\mathbf{U}} \left\|\mathbf{X} - \mathbf{U}\mathbf{U}^\top\mathbf{X}\right\|_{\mathrm{F}}^2 &= \min_{\mathbf{U}} \mathrm{tr}[(\mathbf{X} - \mathbf{U}\mathbf{U}^\top\mathbf{X})^\top \\ &\quad (\mathbf{X} - \mathbf{U}\mathbf{U}^\top\mathbf{X})] \\ &= \min_{\mathbf{U}} [\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) - 2\mathrm{tr}(\mathbf{X}^\top\mathbf{U}\mathbf{U}^\top\mathbf{X}) \\ &\quad + \mathrm{tr}(\mathbf{X}^\top\mathbf{U}\mathbf{U}^\top\mathbf{U}\mathbf{U}^\top\mathbf{X})] \\ &= \max_{\mathbf{U}} \mathrm{tr}(\mathbf{X}^\top\mathbf{U}\mathbf{U}^\top\mathbf{X}) \\ &= \max_{\mathbf{U}} \mathrm{tr}((\mathbf{U}^\top\mathbf{X})^\top\mathbf{U}^\top\mathbf{X}). \end{aligned}$$

Let $\mathbf{K} = (\mathbf{U}^\top\mathbf{X})^\top\mathbf{U}^\top\mathbf{X}$, which is a linear kernel on the transformed data in the subspace $\mathbf{U}^\top\mathbf{X}$; recalling that the features are centered in the original space, multiplying the data $\mathbf{X}$ by the centering matrix $\mathbf{H}$ does not make any change. Hence, we can write

$$\begin{aligned} \max_{\mathbf{U}} \mathrm{tr}((\mathbf{U}^\top\mathbf{X})^\top\mathbf{U}^\top\mathbf{X}) &= \max_{\mathbf{U}} \mathrm{tr}((\mathbf{U}^\top\mathbf{X}\mathbf{H})^\top\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{I}) \\ &= \max_{\mathbf{U}} \mathrm{tr}(\mathbf{H}(\mathbf{U}^\top\mathbf{X})^\top\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{I}) \\ &= \max_{\mathbf{U}} \mathrm{tr}([(\mathbf{U}^\top\mathbf{X})^\top\mathbf{U}^\top\mathbf{X}]\mathbf{H}\mathbf{I}\mathbf{H}) \\ &= \max_{\mathbf{U}} \mathrm{tr}(\mathbf{K}\mathbf{H}\mathbf{I}\mathbf{H}), \end{aligned} \tag{12}$$

where $\mathbf{I}$ is the identity matrix. To derive (12), we have used the identities $\mathbf{H}^\top = \mathbf{H}$ and $\mathbf{X}\mathbf{H} = \mathbf{X}\mathbf{H}\mathbf{I}$ and also noted that the trace operator is invariant to the rotation of its arguments.

To enable providing an interpretation for (12), we recall that identity matrix $\mathbf{I}$ represents a kernel on a random variable, where each data sample has maximum similarity to itself and no similarity, whatsoever, to others. Hence, based on empirical HSIC, the objective function given in (12) indicates that the transformation $\mathbf{U}$ transforms the centered data[2] $\mathbf{X}\mathbf{H}$ to a space where the dependency of random variables x and another random variable whose kernel is identity matrix $\mathbf{I}$ is maximized. This means that using transformation $\mathbf{U}$, the random variable x is transformed such that each data sample has maximum similarity/correlation to itself and no similarity to other data samples. It is well known in the literature that these bases are the principal components of the signal $\mathbf{X}$ that represent the data in an uncorrelated space. With a few manipulations, the objective function given in (12) can be rewritten as follows:

$$\begin{aligned} \max_{\mathbf{U}} \mathrm{tr}((\mathbf{U}^\top\mathbf{X})^\top\mathbf{U}^\top\mathbf{X}) &= \max_{\mathbf{U}} \mathrm{tr}((\mathbf{U}^\top\mathbf{X}\mathbf{H})^\top\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{I}) \\ &= \max_{\mathbf{U}} \mathrm{tr}(\mathbf{H}\mathbf{X}^\top\mathbf{U}\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{I}) \\ &= \max_{\mathbf{U}} \mathrm{tr}(\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{I}\mathbf{H}\mathbf{X}^\top\mathbf{U}). \end{aligned}$$

In other words, we have shown that the optimization problem in (10) is equivalent to

$$\begin{aligned} \max_{\mathbf{U}} \quad & \mathrm{tr}(\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{I}\mathbf{H}\mathbf{X}^\top\mathbf{U}), \\ \text{s.t.} \quad & \mathbf{U}^\top\mathbf{U} = \mathbf{I} \end{aligned} \tag{13}$$

According to the Rayleigh-Ritz Theorem [54], the solution of the optimization problem in (13) is the top eigenvectors of $\mathbf{\Phi} = \mathbf{X}\mathbf{H}\mathbf{I}\mathbf{H}\mathbf{X}^\top$ corresponding to the largest eigenvalues of $\mathbf{\Phi}$. Here, $\mathbf{X}\mathbf{H}\mathbf{I}\mathbf{H}\mathbf{X}^\top$ is the covariance matrix of $\mathbf{X}$.

To summarize, we showed above that the linear decomposition of signals that minimizes the reconstruction error in the mean-squared sense represents the data in an uncorrelated space. This is, in fact, the same as in the principal component analysis (PCA), where the top eigenvectors of the covariance matrix are computed. However, as mentioned before, although minimization of reconstruction error is the ultimate goal in applications such as denoising and coding, in classification tasks, the main goal is maximum discrimination of classes. Hence, we are looking for a decomposition that represents the data in a space where the decomposed data have maximum dependency with their labels. To this end, we propose the new optimization problem as follows

$$\begin{aligned} \max_{\mathbf{U}} \quad & \mathrm{tr}(\mathbf{U}^\top\mathbf{X}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{X}^\top\mathbf{U}), \\ \text{s.t.} \quad & \mathbf{U}^\top\mathbf{U} = \mathbf{I} \end{aligned} \tag{14}$$

where $\mathbf{L}$ is a kernel, e.g., a linear kernel, on the labels $\mathbf{Y} \in \{0,1\}^{c\times n}$, i.e., $\mathbf{L} = \mathbf{Y}^\top\mathbf{Y}$ and c is the number of classes. Here, each column of $\mathbf{Y}$ is $\mathbf{y}_i = \{0,...,1,...,0\}^\top$. In other words, there is exactly one nonzero element in each column $\mathbf{Y}$, where the position of the nonzero element indicates

[2]Here, centered data means that the features are centered, not individual data samples.

the class of the corresponding data sample. The optimization problem given in (14), compromises the reconstruction error to achieve a better discrimination power. Similar to the previous case, the solution for the optimization problem given in (14) is the top eigenvectors of $\mathbf{\Phi} = \mathbf{XHLHX}^\top$. These eigenvectors compose the supervised dictionary to be learned. This dictionary spans the space where the dependency between data $\mathbf{X}$ and corresponding labels $\mathbf{Y}$ is maximized. The coefficients can be computed in this space using the *lasso* as given in (3). However, by recalling (11), a closed-form solution for the coefficients can be invoked as being explained next. Thus far, we have already computed $\mathbf{U}$ in (11) as the top eigenvectors of $\mathbf{\Phi} = \mathbf{XHLHX}^\top$. This makes the dictionary, i.e., $\mathbf{D} = \mathbf{U}$. By replacing this learned dictionary $\mathbf{D}$ into (11), and understanding that $\mathbf{V}$ includes the coefficients we can compute them in sparse way by solving the following minimization problem:

$$\min_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \left(\tfrac{1}{2} \|\mathbf{D}^\top \mathbf{x}_i - \boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1 \right), \tag{15}$$

where $\mathbf{x}_i \in \mathbb{R}^p$ is the i^{th} data sample and $\boldsymbol{\alpha}_i \in \mathbb{R}^k$ (k is the number of dictionary atoms) is the corresponding coefficient to be computed. We can write this minimization problem for each data sample separately as follows

$$\min_{\boldsymbol{\alpha}_i} \left(\tfrac{1}{2} \|\mathbf{D}^\top \mathbf{x}_i - \boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1 \right). \tag{16}$$

The minimization problem in (16) is separable with respect to each element of $\boldsymbol{\alpha}_i$. Hence, we can rewrite (16) as

$$\min_{\boldsymbol{\alpha}_i} \sum_{j=1}^{k} \left\{ \tfrac{1}{2} \left([\mathbf{D}^\top \mathbf{x}_i]_j - \alpha_{ij} \right)^2 + \lambda \left| \alpha_{ij} \right| \right\}, \tag{17}$$

where $[\mathbf{D}^\top \mathbf{x}_i]_j$ and α_{ij} are the j^{th} elements of $\mathbf{D}^\top \mathbf{x}_i$ and $\boldsymbol{\alpha}_i$, respectively, and $|.|$ is the absolute value of its argument. The problem given in (17) has closed-form solution and can be solved using soft-thresholding [55], [56] with the soft-thresholding operator $S_\lambda(.)$, i.e.,

$$\alpha_{ij} = S_\lambda \left([\mathbf{D}^\top \mathbf{x}_i]_j \right), \tag{18}$$

where $S_\lambda(t)$ is defined as follows

$$S_\lambda(t) = \begin{cases} t - 0.5\lambda & \text{if } t > 0.5\lambda \\ t + 0.5\lambda & \text{if } t < -0.5\lambda \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

In conclusion, we propose our supervised dictionary learning as given in Algorithm 1.

One important advantage of the proposed approach in Algorithm 1 is that both the dictionary and coefficients can be computed in closed form. Besides, learning the dictionary and the coefficients are performed separately, and we do not need to learn these two iteratively and alternately, as is common in most supervised dictionary learning approaches in the literature (refer to Subsection II-B).

Algorithm 1 Supervised Dictionary Learning

Input: Training data, $\mathbf{X}_{\text{tr}}$, test data, $\mathbf{X}_{\text{ts}}$, kernel matrix of labels $\mathbf{L}$, training data size, n, size of dictionary, k.
Output: Dictionary, $\mathbf{D}$, coefficients for training and test data, $\boldsymbol{\alpha}_{\text{tr}}$ and $\boldsymbol{\alpha}_{\text{ts}}$.

1: $\mathbf{H} \leftarrow \mathbf{I} - n^{-1}\mathbf{e}\mathbf{e}^\top$
2: $\mathbf{\Phi} \leftarrow \mathbf{X}_{\text{tr}}\mathbf{HLHX}_{\text{tr}}^\top$
3: **Compute Dictionary:** $\mathbf{D} \leftarrow$ eigenvectors of $\mathbf{\Phi}$ corresponding to top k eigenvalues
4: **Compute Training Coefficients:** For each data sample $\mathbf{x}_{\text{tr}_i}$ in the training set, use $\alpha_{ij} = S_\lambda \left([\mathbf{D}^\top \mathbf{x}_{\text{tr}_i}]_j \right), j = 1, ..., k$ to compute the corresponding coefficient
5: **Compute Test Coefficients:** For each data sample $\mathbf{x}_{\text{ts}_i}$ in the test set, use $\alpha_{ij} = S_\lambda \left([\mathbf{D}^\top \mathbf{x}_{\text{ts}_i}]_j \right), j = 1, ..., k$ to compute the corresponding coefficient

C. Kernelized Supervised Dictionary Learning

One of the main advantages of the proposed formulation for SDL, compared to other techniques in the literature, is that we can easily embed a kernel into the formulation. This enables nonlinear transformation of data into a high-dimensional feature space where the discrimination of classes can be more efficiently performed. This is especially beneficial by incorporating data-dependent kernels[3], such as those based on normalized compression distance [25].

Kernelizing the proposed approach is straightforward. Suppose that $\mathbf{\Psi}$ is a feature map representing the data in feature spaces $\mathcal{H}$ as follows:

$$\begin{aligned} \mathbf{\Psi} : X &\rightarrow \mathcal{H} \\ \mathbf{X} &\mapsto \mathbf{\Psi}(\mathbf{X}). \end{aligned} \tag{20}$$

To kernelize the proposed SDL, we express the matrix of bases $\mathbf{U}'$ as a linear combination of the projected data points into the feature space using representation theory [57], i.e., $\mathbf{U}' = \mathbf{\Psi}(\mathbf{X})\mathbf{W}$. In other words, $\mathbf{W} \in \mathbb{R}^{n \times k}$ represents $\mathbf{U}' \in \mathbb{R}^{p' \times k}$ in feature space $\mathbf{\Psi}(\mathbf{X}) \in \mathbb{R}^{p' \times n}$. By replacing $\mathbf{X}$ by $\mathbf{\Psi}(\mathbf{X})$ and $\mathbf{U}'$ by $\mathbf{\Psi}(\mathbf{X})\mathbf{W}$ in the objective function of (14), we obtain

$$\begin{aligned} \text{tr}(\mathbf{U}'^\top \mathbf{\Psi}(\mathbf{X})\mathbf{HLH}\mathbf{\Psi}(\mathbf{X})^\top \mathbf{U}') &= \text{tr}(\mathbf{W}^\top \mathbf{\Psi}(\mathbf{X})^\top \mathbf{\Psi}(\mathbf{X}) \\ &\quad \mathbf{HLH}\mathbf{\Psi}(\mathbf{X})^\top \mathbf{\Psi}(\mathbf{X})\mathbf{W}) \\ &= \text{tr}(\mathbf{W}^\top \mathbf{KHLHKW}), \end{aligned}$$

with the constraint

$$\begin{aligned} \mathbf{U}'^\top \mathbf{U}' &= \mathbf{W}^\top \mathbf{\Psi}(\mathbf{X})^\top \mathbf{\Psi}(\mathbf{X})\mathbf{W} \\ &= \mathbf{W}^\top \mathbf{KW}, \end{aligned}$$

where $\mathbf{K} = \mathbf{\Psi}(\mathbf{X})^\top \mathbf{\Psi}(\mathbf{X})$ is a kernel function on data. Combining this objective function and the constraint, the optimization problem for the kernelized SDL is

$$\begin{aligned} \max_{\mathbf{W}} \quad & \text{tr}(\mathbf{W}^\top \mathbf{KHLHKW}), \\ \text{s.t.} \quad & \mathbf{W}^\top \mathbf{KW} = \mathbf{I} \end{aligned} \tag{21}$$

[3]Although it is true that all kernels are computed on the data and hence, are data-dependent, the term is used in the literature to refer to those types of kernels that do not have any closed form.

Algorithm 2 Kernelized Supervised Dictionary Learning

Input: Kernel on training data, $\mathbf{K}_{\text{tr}}$, kernel on test data, $\mathbf{K}_{\text{ts}}$, kernel on labels $\mathbf{L}$, training data size, n, size of dictionary, k.
Output: Dictionary, $\mathbf{D}$, coefficients for training and test data, $\boldsymbol{\alpha}_{\text{tr}}$ and $\boldsymbol{\alpha}_{\text{ts}}$.

1: $\mathbf{H} \leftarrow \mathbf{I} - n^{-1}\mathbf{e}\mathbf{e}^\top$
2: $\boldsymbol{\Phi} \leftarrow \mathbf{K}_{\text{tr}}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{K}_{\text{tr}}$
3: **Compute Dictionary:** $\mathbf{D} \leftarrow$ top k generalized eigenvectors of the generalized eigenvalue problem $\boldsymbol{\Phi}\mathbf{u} = \lambda_0\mathbf{K}\mathbf{u}$.
4: **Compute Training Coefficients:** For each column $\mathbf{k}_{\text{tr}_i}$ of the $\mathbf{K}_{\text{tr}}$, use $\alpha_{ij} = S_\lambda\left([\mathbf{D}^\top\mathbf{k}_{\text{tr}_i}]_j\right), j = 1, ..., k$ to compute the corresponding coefficient
5: **Compute Test Coefficients:** For each column $\mathbf{k}_{\text{ts}_i}$ of the $\mathbf{K}_{\text{ts}}$, use $\alpha_{ij} = S_\lambda\left([\mathbf{D}^\top\mathbf{k}_{\text{ts}_i}]_j\right), j = 1, ..., k$ to compute the corresponding coefficient

whose solution is the top generalized eigenvectors of the generalized eigenvalue problem $\mathbf{KHLHKu} = \lambda_0\mathbf{Ku}$[4] ($\lambda_0$ is a scalar and $\mathbf{u}$ is a vector) according to the Rayleigh-Ritz Theorem [54]. To realize how the coefficients can be computed for the training and test sets, we replace $\mathbf{U}^{'} = \boldsymbol{\Psi}(\mathbf{X})\mathbf{W}$ in (11), knowing that $\mathbf{X}$ has to be also replaced by $\boldsymbol{\Psi}(\mathbf{X})$, to obtain

$$\begin{aligned}\mathbf{V}^{'} &= \mathbf{W}^\top\boldsymbol{\Psi}(\mathbf{X})^\top\boldsymbol{\Psi}(\mathbf{X}) \\ &= \mathbf{W}^\top\mathbf{K}. \end{aligned} \tag{22}$$

The form given in (22) is very similar to what is given in (11) and from now on we can use the same steps as provided for the proposed SDL in previous subsection to compute the coefficients. In other words, considering $\mathbf{D} = \mathbf{W}$ and knowing that $\mathbf{V}^{'}$ includes the coefficients, we can find the sparse coefficients using similar formulation as in (15), i.e.,

$$\min_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \left(\tfrac{1}{2}\|\mathbf{D}^\top\mathbf{k}_i - \boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right), \tag{23}$$

where $\mathbf{k}_i \in \mathbb{R}^n$ is one column of kernel matrix $\mathbf{K}$. This problem is again separable and each element of coefficient $\boldsymbol{\alpha}_i$ can be computed using the soft-thresholding operator

$$\alpha_{ij} = S_\lambda\left([\mathbf{D}^\top\mathbf{k}_i]_j\right). \tag{24}$$

The algorithm for kernelized SDL is given in Algorithm (2).

IV. Experiments

In this section, we evaluate the performance of the proposed SDL on various datasets and in different applications such as analyzing face data, digit recognition, and in classification of real-world data such as satellite images and textures. We will show through various experiments the main advantages of the proposed SDL, such as a compact dictionary – i.e., a discriminative dictionary even at small dictionary size – and fast performance. Also, we will show how its kernelized version enables embedding data-dependent kernels into the proposed SDL to significantly improve the performance on difficult classification tasks. Table I provides the details of the datasets used in our experiments, their dimensionality, number of classes, and the number of instances per class, as well as in the training and test sets as used in our experiments.

A. Implementation Details

In our approach, the first step is to compute the dictionary by computing the (generalized) eigenvectors of $\boldsymbol{\Phi}$ as provided in Algorithms 1 or 2. To avoid rank deficiency in the computation of kernel on labels, we add the identity matrix of the same size to the kernel, i.e., $\mathbf{L} = \mathbf{Y}^\top\mathbf{Y} + \mathbf{I}$. Then we need to calculate the coefficients using soft-thresholding approach as given in (18) or (24) for the proposed SDL or KSDL, respectively. The optimal value of the regularization parameter in soft thresholding (λ^*), which controls the level of sparsity, has been computed by 10-fold cross-validation on the training set to minimize the mean-squared error. This λ^* is then used to compute the coefficients for both training and test sets [5].

As is suggested in [58], the coefficients computed on the training set are used for training a support vector machine (SVM). RBF kernel has been used for the SVM and the optimal parameters of the SVM, i.e., the optimal kernel width γ^* and trade-off parameter C^*, are found by grid search and 5-fold cross-validation on the training set[6]. The coefficients computed on the test set are then submitted to this trained SVM to label unseen test examples.

Two measures are considered to evaluate the performance of the classification systems: classification error and balanced classification error, which are defined as follows:

$$\text{E} = \frac{n_{\text{wr}}}{n}, \tag{25}$$

$$\text{BE} = \frac{1}{c}\sum_{i=1}^{c}\frac{n_{\text{wr}}^i}{n_i}, \tag{26}$$

where E and BE are classification error and balanced error, respectively; n_{wr} is the total wrongly-classified data samples; n is the total number of data samples; c is the number of classes; n_{wr}^i is the number of wrongly-classified objects in class i; and n_i is the number of data samples in class i. According to this definition, E is the total number of wrongly-classified data samples over the total number of objects. Hence, if there are fewer objects in one class, wrongly-classified objects in that class contribute less towards the overall classification system error. The definition of BE, however, gives the same weight to all classes irrespective of the number of objects in each class. To further clarify the difference between these two measures, we consider an extreme case. Suppose that in a two-class problem, there are 98 objects in one class and 2 objects in another class. If all 98 objects are correctly classified in class one, and out of 2 objects in class two, only one is correctly classified, the classification error is $\text{E} = 1/100 = 1\%$, whereas the balanced error is $\text{BE} = (1/2 + 0/98)/2 = 25\%$. If,

[4]We have used λ_0 here as it is different from λ in the *lasso* and soft-thresholding.

[5]One λ^* is computed for each data point in the training set. However, the averaged λ^* over the whole training set is used to compute the coefficients on the training and test sets as it yields better generalization.

[6]10-fold cross-validation yields very close results. Thus to avoid higher computation load, 5-fold cross-validation is adopted.

TABLE I: The datasets used in this paper.

Dataset	Dataset Info.					
	Samples	Samples per Class	Training Size	Test Size	Classes	Dim.
Face (Olivetti)[a]	400	119, 281	200	200	2	4096
Digit (USPS)[b]	9298	-	7291	2007	10	256
Sonar[c]	208	97, 111	104	104	2	60
Ionosphere[c]	351	225, 126	176	175	2	34
Texture (I)[d]	5500	500	2750	2750	11	40
Satimage[d]	6435	1533, 703, 1358, 626, 707, 1508	3218	3217	6	36
Texture (II)[e]	600	300	300	300	2	256

[a]http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
[b]http://www-i6.informatik.rwth-aachen.de/∼keysers/usps.html
[c]http://archive.ics.uci.edu/ml/
[d]http://www.dice.ucl.ac.be/neural-nets/Research/Projects/ELENA/databases/REAL/
[e]http://www.ux.uis.no/∼tranden/

for example, this classification system is supposed to classify healthy versus unhealthy cases, BE is a better measure to evaluate the classification system, because both classes equally contribute towards the estimation of error irrespective of the number of data samples in each. Since as indicated on the third column of Table I, some datasets used in our experiments, such as Face, Sonar, Ionosphere, and Satimage, are not balanced[7], we have provided both E and BE for them in next subsections.

B. Face Data

In this experiment, our main goal is to show the compactness of our proposed dictionary. We use the Olivetti face dataset of AT&T [59]. This data consists of 400 face images of 40 distinct subjects, i.e., 10 images per subject, with varying lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). The original size of each image is 92×112 pixels, with 256 gray levels per pixel. However, in our experiments, each image has been cropped from the center to be 64×64 pixels.

The main task in our experiments is to classify the faces into glasses/no-glasses classes. To this end, the images are labeled to indicate these two classes, with 119 in the glasses class and 281 in the no-glasses. Typical images of these two classes are shown in Fig. 1. All images are normalized to have zero mean and unit ℓ_2-norm. Half of the images are randomly selected for training, and the other half for testing; the experiments are repeated 10 times, and the average error (E) and balanced error (BE) are reported in Table II. The experiments are performed on varying dictionary sizes, including 2, 4, 8, 16, and 32. The results are compared with several unsupervised and supervised dictionary learning approaches, as shown in Table II. For K-SVD, the fast implementation provided by Rubinstein [60] has been used. We have implemented DK-SVD with K-SVD as the core. The difference between supervised and unsupervised k-means is that in unsupervised k-means, the dictionary is learned on the whole training set, whereas in the supervised one, one dictionary is learned per class as suggested in the texton-based approach by Varma and Zisserman [17], [32]. The code for metaface approach has been provided by the authors [33]. The same as our approach, the parameter(s) of all these rival approaches are tuned using 5-fold cross-validation on the training set.

[7]The USPS digit dataset is also somewhat imbalanced. However, since in the literature, particularly in [41] with which our results are compared, only classification error (E) is provided, we also present our results using this measure only. Also since the publically available USPS data comes in separate training and test sets, and representing the number of instances per class takes space for 10 classes, we have not provided this information for the USPS dataset in Table I.

Fig. 1: Typical face images from the Olivetti face dataset in two classes of glasses vs. no-glasses.

As can be seen in Table II, our approach performs the best among these approaches. The compactness of the dictionary learned using the proposed SDL is noticeable from the results at small dictionary size. For example, at the dictionary size of two, while the error of our approach is 12.60%, unsupervised k-means yields a 27.4% error, which is more than twice as large as our approach. The best result obtained by other supervised dictionary approaches (here metaface) yields a 17.55% error at this dictionary size, which is more than 5% above the error generated by the proposed SDL. The same conclusion can be made using balanced error. Interestingly, supervised k-means performs significantly better than the unsupervised one, particularly at small dictionary sizes. The main conclusion of this experiment is that the proposed SDL generates a very discriminative and compact dictionary, compared to well-known unsupervised and supervised dictionary learning approaches.

C. Digit Recognition

The second experiment is performed on the task of handwritten digit classification on the USPS dataset [61]. This dataset consists of handwritten digits, each with the size of 16×16 pixels with 256 gray levels. There are 7291 and 2009 digits in the training and test sets, respectively.

We compare our results with the most recent SDL technique, which yields the best results published so far on this

TABLE II: Classification error (E) and balanced error (BE) on test set for Olivetti face data using the proposed SDL. The results are compared with several other dictionary learning approaches in the literature. The best results obtained are highlighted.

Approach		Dictionary Size									
		2		4		8		16		32	
		E	BE	E	BE	E	BE	E	BE	E	BE
Unsupervised	*k*-means	27.40 ±2.04	39.35 ±4.01	22.60 ±5.18	29.05 ±5.97	13.15 ±2.38	17.36 ±4.17	8.15 ±1.81	10.71 ±3.12	5.75 ±1.70	8.40 ±2.62
	K-SVD [28]	28.20 ±2.45	41.40 ±4.60	20.60 ±2.41	27.36 ±5.23	9.65 ±1.62	12.97 ±2.52	7.75 ±2.06	11.07 ±3.13	4.05 ±1.23	6.06 ±2.26
Supervised	Proposed SDL	**12.60** **±3.51**	**16.60** **±4.16**	10.30 ±2.59	12.70 ±3.50	**5.30** **±2.11**	**6.21** **±2.92**	**4.95** **±1.14**	**6.06** **±1.88**	**3.55** **±1.59**	**4.68** **±2.38**
	DK-SVD [38]	17.80 ±3.06	19.36 ±3.52	**10.25** **±2.48**	**10.15** **±3.04**	8.75 ±2.02	11.25 ±4.06	7.05 ±2.11	8.70 ±1.31	6.75 ±1.53	10.13 ±2.92
	k-means[a] [17]	17.75 ±3.65	23.45 ±5.71	10.40 ±2.56	14.01 ±2.76	7.40 ±1.90	10.57 ±2.93	5.55 ±1.62	7.84 ±2.58	3.65 ±1.20	5.45 ±2.01
	Metaface [33]	17.55 ±2.87	19.39 ±3.02	11.25 ±2.35	15.35 ±2.61	9.75 ±3.58	14.58 ±5.88	7.60 ±1.39	11.74 ±1.91	5.45 ±0.96	9.28 ±1.46

[a]Supervised *k*-means learns one sub-dictionary per class and then compose all learned sub-dictionaries into one.

dataset [41]. To facilitate a direct comparison with what is published in [41], we use the same setup as they have reported. To this end, since the most effective techniques on digit recognition deploy shift invariant features [62], and since neither our approach nor the one reported in [41] benefit from these kind of features, as suggested in [41], the training set is artificially augmented by adding digits which are shifted version of original ones, moved by one pixel in all four directions. Although this is not an optimal and sophisticated way of introducing shift invariance to the SDL techniques, it takes into account this property in a fairly simple approach. Each digit in training and test sets is normalized to have zero mean and unit ℓ_2-norm.

Table III shows the results obtained using the proposed approach in comparison with the unsupervised and supervised dictionary learning techniques reported in [41]. As can be seen, again our approach introduces a very compact dictionary such that its performance at dictionary size of 50 is the same as the performance of the system reported in [41] using a dictionary of 100 atoms. With increasing the dictionary size, the performance of our approach slightly degrades. This is mainly because the bases or dictionary atoms in our approach are associated with the directions of maximum separability of the data, as has been enforced by the optimization problem in (14). Nevertheless, the number of useful bases depends on the intrinsic dimensionality of the subspace, which in turn depends on the nature of the data. If the number of dictionary atoms goes beyond this intrinsic dimensionality, then adding more atoms does not improve the performance but may degrade it, as they are not associated with separable directions but related to noise. On the other hand, it is important to notice that we can achieve a reasonable performance using much less complexity than the best rival. It should be also noted that the best performance achieved by our approach (happening at a small dictionary size of 50) is just 0.25% worse than the best results obtained by [41] (happening at dictionary size of 300, i.e., with much higher complexity). This means that our approach misclassifies only 5 more digits compared to the best results obtained in [41], whereas for the same dictionary size (50), our approach performs 0.55% better, i.e., classifies 11 more digits correctly. On the other hand, w.r.t. the complexity, our proposed approach offers a much simpler solution for SDL than the approach in [41]: there are fewer parameters to tune, the dictionary can be computed in closed form, and there is no need to solve a complicated nonconvex optimization problem as used in [41] by iteratively and alternately optimizing classifier, dictionary, and coefficient learning.

TABLE III: Classification error on test set for digit recognition on USPS data using proposed SDL compared with the most effective SDL approach reported in the literature on the same data [41]. Highlighted entries represent the best results obtained at each dictionary size.

Approach	Dictionary Size			
	50	100	200	300
Unsupervised [41]	8.02	6.03	5.13	4.58
Supervised [41]	3.64	**3.09**	**2.88**	**2.84**
Proposed SDL	**3.09**	3.19	3.24	-

As a final remark, due to the orthonormality constraint in the optimization problem of our proposed SDL as given in (14), overcompleteness is not possible in our proposed SDL. This is the reason that in Table III, no results are reported for a dictionary size of 300 for our approach. However, as mentioned above, due to the compactness of our dictionary, good results are obtained at a much smaller dictionary size, which is a desired attribute as it decreases the computational load. Also, the proposed kernelized version of our proposed approach given in (21) and Algorithm 2 can learn dictionaries as large as n, i.e., the number of data points used for training, which is usually greater than the dimensionality of the data p (see Table I for the relative size of p and n for the data used in our experiments).

D. Other Real-World Data

In the two previous sections, the classification task was performed on the pixels of images directly. In this section, we evaluate the performance of the proposed approach on the classification of some real-world data using features extracted. Four datasets with varying complexity from 2- to 11-class,

with the dimensionality of up to 60 features, and also with as many as 6435 data samples are used in these experiments (refer to Table I for detailed information on these datasets). All data are preprocessed to have zero mean and unit ℓ_2-norm, except Satimage dataset, where the features are normalized to be in the range of [0, 1] due to the large variation of feature values.

Since the rival approaches are the same as that used for face data, their implementations are the same as was explained in Subsection IV-B. There is one additional remark here on the implementation of supervised *k*-means on datasets with more than two classes, such as the Texture and Satimage datasets. We have implemented this approach in a way to ensure that the dictionary atoms are evenly computed over different classes as much as possible. For example, in the case of dictionary size of 8 and for the Texture dataset that has 11 classes, we have first selected 11 dictionary atoms, one from each class, then 8 of them are randomly selected.

On all datasets, the experiments are repeated ten times over a random split of data into half for training and half for testing. The average and standard deviation of classification error (E) and balanced error (BE) are reported in Tables IV and V, respectively, in comparison with several other unsupervised and supervised dictionary learning approaches. Since the texture dataset is balanced, the error and balanced error are the same, therefore, the balanced error has not been reported for this dataset in Table V. We have also included the results of classification using a kernelized version of our proposed SDL with radial basis function (RBF) as the kernel. The width of the RBF kernel has been selected based on a self-tuning approach [63].

As can be seen from Tables IV and V, the proposed SDL or its kernelized version performs the best in all cases, except for the dictionary size of 8 and 16 on the Sonar data. The better performance of supervised *k*-means at the dictionary sizes of 8 and 16 is not significant, as the resultant standard deviation is very high. DK-SVD performs poorly (even worse than the unsupervised K-SVD approach) on these datasets mainly because, by design, it uses a linear classifier (refer to Subsection II-B and [38] for more description on this approach). The poor performance of metaface is because it usually performs well at very large dictionary size. Hence, at reported dictionary sizes, its training is not sufficient to capture the underlying data structure. For example, for Sonar data, while the proposed SDL can achieve an error of 20.77$\pm$4.67 at the dictionary size of 32, the metaface approach can only achieve this accuracy at the dictionary size of 64 (error 20.00$\pm$4.75). However, using large dictionary size adds to the computational load of the approach.

As a final remark on the results presented in this subsection, we would like to comment on the relative performance of proposed SDL and its kernelized version KSDL. The relative performance of these two approaches mainly depends on the nature of the data to be classified, and whether it has a linear or nonlinear behavior. In other words, it depends whether the data can be represented as a subspace or a submanifold. In the former case, the proposed SDL should be sufficient to model the data, while in the latter case, the KSDL should potentially perform better. However, the success of KSDL depends on the proper selection of the kernel and its parameter(s). In fact, even if the data has a linear nature and can be represented in a subspace, the KSDL should also perform as well as SDL, but this again depends on proper kernel and model selection. In the next subsection, we will show how choosing a proper kernel can significantly improve the results using KSDL approach for a rather complicated dataset.

E. Patch Classification on Texture Data

To show the benefit of using data-dependent kernels such as kernels computed using normalized compression distance [25], in this section, we perform classification on patches extracted from texture images. We compare our results with and without kernels using the proposed approach, and also compare them to the results published in [15], i.e., two supervised dictionary learning approaches called SDL-G BL (G for generative and BL for bilinear model) and SDL-D BL (D for discriminative). To ease the comparison, we use the same data as in [15], i.e., classification on texture pair of D5 and D92 from the Brodatz album, shown in Fig. 2. Also the same as [15], 300 patches are randomly extracted from the left half of each texture image for training and 300 patches from the right half for testing. This is to ensure that there is no overlap among the patches used in the training and test sets.

We have used the RBF kernel and two data-dependent compression-based kernels as reported in [64] (CK-1) and [26] (d_{N}) as the kernel for the proposed kernelized SDL. The latter deploys MPEG-1 as the compressor as suggested in [64] for the computation of normalized compression distance [25]. However, compared to the measure proposed in [64] (CK-1), it proposes a novel compression-based dissimilarity measure (d_{N}) that performs well on both small and large patch sizes (as shown in [26], CK-1 does not work properly on small patch sizes). Besides, d_{N} is a semi-metric.

Table VI provides the results of classification using the proposed SDL with and without kernels. It also compares the results with *k*-means as an unsupervised approach to compute the dictionary, and with the results published in [15] for the same number of patches (300) and the same dictionary size, i.e., 64. The sparsity of the coefficients, i.e., the number of nonzero coefficients, are also provided in this table (this is not reported for SDL-G BL and SDL-D BL in [15]). As can be seen, using a compression-based data-dependent kernel based on d_{N} dramatically improves the results. The classification error is even lower than the one obtained by the SDL-D BL approach using 30000 patches for training, which yields the best results on this data with the classification error = 14.26% in [15]. Moreover, as the sparsity of the coefficients indicate, the proposed approach with data-dependent kernel d_{N} deploys the smallest number of dictionary atoms in the reconstruction of the signal, i.e., benefits the most from the sparse representation, as it uses almost half of the dictionary elements compared to other approaches. This has a great impact on the computation load of the classification task, especially in the stage of training and testing of the classifier. Our experiments show (not reported in Table VI) that by using

TABLE IV: The results of classification *error* (%) on various real-world datasets using different methods and in different dictionary sizes. The best results obtained are highlighted.

Approach		Sonar			Ionosphere			Texture			Satimage		
		8	16	32	8	16	32	8	16	32	8	16	32
Unsupervised	*k*-means	28.56 ±5.53	24.52 ±5.43	24.42 ±3.77	7.37 ±2.48	7.71 ±1.41	8.06 ±1.86	2.49 ±0.66	1.12 ±0.25	0.97 ±0.29	13.36 ±0.47	13.02 ±0.64	12.87 ±0.72
	K-SVD [28]	27.31 ±2.69	24.81 ±6.69	28.56 ±4.25	8.69 ±4.12	9.09 ±1.73	8.00 ±1.50	1.54 ±0.30	0.81 ±0.27	0.83 ±0.19	10.42 ±0.43	10.70 ±0.73	11.92 ±0.36
Supervised	Proposed SDL	27.79 ±3.47	22.50 ±2.73	**20.77 ±4.67**	5.94 ±1.66	**5.60 ±1.41**	**5.43 ±1.41**	**1.44 ±0.38**	**0.45 ±0.12**	**0.31 ±0.10**	11.25 ±0.36	10.58 ±0.40	10.66 ±0.41
	KSDL-RBF[a]	28.75 ±3.88	27.31 ±4.40	26.35 ±3.22	**5.66 ±1.97**	5.89 ±2.03	6.17 ±2.07	1.68 ±0.26	1.20 ±0.23	1.19 ±0.22	**10.18 ±0.36**	**9.81 ±0.38**	**9.66 ±0.33**
	DK-SVD [38]	32.40 ±4.53	32.69 ±4.32	29.04 ±4.15	16.11 ±1.88	18.00 ±3.51	15.89 ±2.50	27.91 ±3.87	6.15 ±0.82	7.28 ±1.86	35.36 ±13.29	20.15 ±1.38	28.89 ±4.19
	k-means [17]	**24.62 ±5.31**	**22.31 ±4.27**	22.88 ±5.98	7.54 ±1.39	9.54 ±1.59	10.00 ±2.35	2.11 ±0.42	0.95 ±0.14	0.82 ±0.22	13.61 ±0.36	12.65 ±0.32	12.98 ±0.65
	Metaface [33]	26.74 ±3.17	27.89 ±5.22	23.17 ±4.43	18.29 ±1.62	21.54 ±2.89	16.29 ±2.52	9.76 ±0.55	10.03 ±1.88	4.64 ±0.57	23.43 ±1.38	27.14 ±1.05	24.85 ±1.53

[a]Proposed kernel SDL with RBF kernel.

TABLE V: The results of classification *balanced error* (%) on various real-world datasets (except texture data, for which the error and balanced error are the same as the dataset is balanced) using different methods and in different dictionary sizes. The best results obtained are highlighted.

Approach		Sonar			Ionosphere			Satimage		
		8	16	32	8	16	32	8	16	32
Unsupervised	*k*-means	28.41 ±5.66	24.26 ±4.93	24.25 ±3.97	8.36 ±3.11	9.40 ±1.56	9.25 ±2.97	16.99 ±0.65	16.39 ±0.86	16.50 ±0.73
	K-SVD [28]	27.09 ±2.61	24.87 ±6.36	28.73 ±4.27	10.35 ±4.97	10.63 ±2.05	8.46 ±1.60	13.31 ±0.54	13.44 ±0.85	15.16 ±0.48
Supervised	Proposed SDL	27.75 ±3.50	22.65 ±2.87	**20.86 ±4.84**	7.06 ±1.57	**6.22 ±1.39**	**6.20 ±1.62**	14.07 ±0.33	13.12 ±0.54	13.25 ±0.49
	KSDL-RBF[a]	28.64 ±3.91	27.12 ±4.79	26.03 ±3.28	**6.06 ±1.92**	6.33 ±2.13	6.58 ±2.20	**12.96 ±0.40**	**12.32 ±0.60**	**12.14 ±0.46**
	DK-SVD [38]	33.61 ±3.76	33.79 ±4.46	29.56 ±4.11	18.98 ±3.54	20.17 ±4.67	18.78 ±4.49	35.64 ±8.80	22.22 ±1.51	29.85 ±3.42
	k-means [17]	**24.74 ±5.06**	**22.32 ±4.26**	22.48 ±5.65	8.61 ±1.92	11.29 ±1.94	11.79 ±3.34	17.21 ±0.48	16.01 ±0.52	16.55 ±0.88
	Metaface [33]	27.35 ±4.13	30.20 ±3.44	23.76 ±6.19	25.79 ±2.98	29.67 ±3.64	24.43 ±3.32	30.26 ±1.94	32.89 ±2.01	32.26 ±2.27

[a]Proposed kernel SDL with RBF kernel.

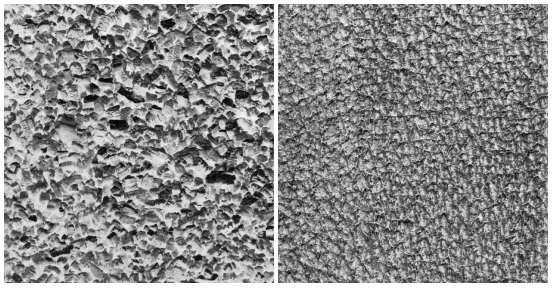


Fig. 2: Texture images of D5 and D92 from Brodatz album.

a slightly larger regularization parameter λ in soft thresholding such that the reconstruction error is within one standard deviation of the minimum, the sparsity of coefficients can be significantly increased. That is, the average number of nonzero coefficients can be reduced to about 5% of the total number of coefficients without compromising the classification error. The classification error is 9.90±1.43 in this case, which is even slightly better than what is reported in Table VI.

F. The Effect of Noisy Labels on the Performance of the Proposed SDL

Since in supervised dictionary learning approaches the information category is used in the learning of the dictionary, one main question will be: "to what extent are these approaches sensitive to noisy labels?". In this subsection we will try to address this question.

As defined in Subsection III-B, the labels $\mathbf{Y} \in \{0,1\}^{c\times n}$ can only take the values 0 or 1. Therefore, what we mean by noisy labels is that 0 might be converted to 1, or vice versa. We assume that in each column of noisy labels $\hat{\mathbf{Y}}$, there is still only one nonzero element, which indicates the class of the corresponding object.

Almost all the categories of SDL mentioned in Subsection II-B utilize the labels directly or indirectly in the learning of the dictionary. For example, in the first SDL category, one dictionary is learned per class. Therefore, if one object is wrongly assigned to a class, this object will contribute to learning dictionary atoms in the wrong class, which consequently may lead to reducing the efficiency of the learned dictionary in the classification task. In our proposed approach, as indicated

TABLE VI: Classification error and the number of nonzero coefficients on the test set for texture pair D5-D92 of Brodatz album. The dictionary size is 64. Using data-dependent kernels and the proposed kernelized SDL can significantly improve the results.

Approach		Average No. of Nonzero Coefficients		Classification Error (%)
		Train Set	Test Set	
k-means		47.85	48.99	27.75±2.29
Proposed SDL		59.80	59.85	26.43±2.95
Proposed kernel SDL	RBF	62.88	62.51	28.85±1.84
	CK-1 [64]	64	64	26.05±1.07
	d_{N} [26]	**33.46**	**31.53**	**10.15±1.30**
SDL-G BL[a] [15]		-	-	26.34
SDL-D BL[a] [15]		-	-	26.34

[a]The average no. of nonzero coefficients is not provided for this approach in [15].

in the optimization problem (14), a linear kernel over the labels is used to include the category information in the learning of the dictionary. Hence, it is natural that we expect that noisy labels degrade the efficacy of the dictionary learned in the classification task.

To address the question raised in the beginning of this subsection, we have performed experiments on the Olivetti face dataset. In these experiments, we have included a certain percentage of wrong labels in the learning of the supervised dictionary, and then performed the classification task using this dictionary. Since our main concern is to see how sensitive the dictionary is to noisy labels, correct labels are used in the classifier over the training set. In other words, in our experiments, noisy labels are only used in the learning of the dictionary, and correct labels in the classifier. This may not be a realistic setup as when we have wrong labels, we do not have the correct labels, otherwise we could also use them in the learning of the dictionary. However, if we use wrong labels in the classifier as well, we do not know to what extent the dictionary is affected by wrong labels because wrong classification might be also due to misguiding the classifier.

The results are shown in Fig. 3 for the dictionary sizes of 2, 4, 8, and 16, and for various supervised dictionary learning approaches as used in the experiments on the Olivetti face dataset (refer to Table II). As can be seen from this figure, our proposed SDL is the least sensitive to noisy labels. Also, by increasing the dictionary size, the sensitivity to noisy labels is reduced for our proposed SDL as well as for the supervised k-means. It makes sense to see lower sensitivity to noisy labels at larger dictionary sizes for the proposed SDL, because noisy labels will cause the discriminative directions to move away from leading atoms or bases in the learned dictionary, which degrades the effectiveness of the dictionary at small dictionary sizes, while at larger dictionary sizes, these discriminative directions will appear again, although not in leading atoms. Also in supervised k-means, by increasing the dictionary size, it is more likely that some of the cluster centers in each class, which are the dictionary atoms in that class, correspond to the correctly-labeled data samples. For example, if the dictionary size is two in a two-class problem, there is only one dictionary atom per class. Hence, if this dictionary atom represents wrong data samples due to noisy labels, the dictionary learned completely fails to model the data samples correctly. However, by increasing the dictionary size, this catastrophic failure is less likely to happen.

However, this phenomenon cannot be observed for the DK-SVD and the metaface approaches. DK-SVD does not follow this behavior mainly because the learning of the dictionary and classifier is performed in one optimization problem, as explained in Subsection II-B and in [38]. Hence, noisy labels also affect the learning of the linear classifier involved, and we could not find any way to include the noisy labels only in the learning of the dictionary, not the classifier.

Similarly, in the metaface approach, the class labels used during learning the dictionary are used to tag each dictionary as to what class it belongs to. This tag is later used to indicate the class label of the test object that minimizes the residue obtained using the reconstruction error computed on the subdictionary elements belonging to a class and a test object. Therefore, similar to the DK-SVD approach, there is no way to include the noisy class labels only in the learning of the dictionary, and not in the classifier. This explains why noisy labels have greater impact on DK-SVD and metaface approaches, as they affect both the dictionary learning and training of the classifiers. Based on these explanations, we admit that comparing the effect of noisy labels on our proposed SDL with DK-SVD or metaface is not completely fair, as in our approach (as well as in supervised k-means) we deliberately avoided the impact of noise on the training of classifiers, whereas we could not avoid it in the DK-SVD and metaface approaches.

V. Discussions and Conclusions

In this paper we proposed a novel supervised dictionary learning. The proposed approach learns the dictionary in a space where the dependency between the data and category information is maximized. Maximizing this dependency has been performed based on the concept of the Hilbert Schmidt independence criterion (HSIC). This introduces a data decomposition that represents the data in a space with maximum dependency with category information. We showed that both the dictionary and sparse coefficients can be learned in closed form. Our experiments using real-world data with varying complexity shows that the proposed approach is very efficient in classification tasks, and outperforms other unsupervised and supervised dictionary learning approaches in the literature. Also, the proposed approach is very fast and efficient in computation.

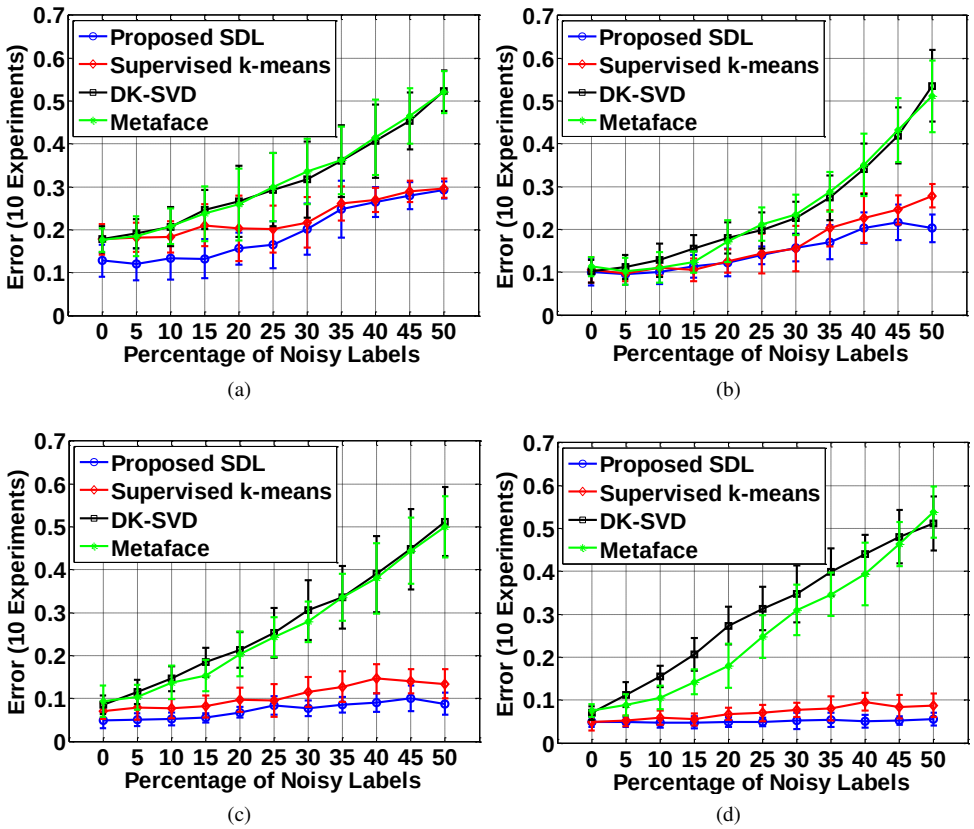


Fig. 3: The error rate of the classification system for Olivetti face recognition system to discriminate between faces with and without glasses. The effect of noisy labels in learning dictionary are shown for the dictionary sizes of (a) 2, (b) 4, (c) 8, and (d) 16.

Moreover, we showed how the proposed SDL can be kernelized. This enables the proposed SDL to benefit from data-dependent kernels. It was shown using some experiments that the proposed kernelized SDL can significantly improve the results in difficult classification tasks compared to other SDL approaches in the literature. To the best of our knowledge, this is the first SDL in the literature that can be kernelized, and thus benefit from data-dependent kernels embedded into the SDL.

The proposed approach learns a very compact dictionary, in the sense that it significantly outperforms other approaches when the size of the dictionary is very small. This shows that the proposed SDL can effectively encode the category information into the learning of the dictionary such that it can perform very well in classification tasks using few atoms. In the dictionary learning literature, usually the dictionary learned is overcomplete, i.e., the number of elements in the learned dictionary is larger than the dimensionality of the data/dictionary. In our proposed SDL, due to the orthonormality constraint on the dictionary atoms as detailed in (14), the dictionary cannot be overcomplete. However, there are two remarks here: first, as discussed above, our dictionary is very compact and as the experiments show, the proposed SDL performs very well at small dictionary size, which is usually below even complete dictionary size. This is a main advantage of the proposed approach, as small dictionary size means lower computational cost. Second, the kernelized version of the proposed approach can easily learn dictionaries as large as n, the number of data samples in the training set. This is because the kernel computed on the data is of the dimensionality of n, which is usually greater than the dimensionality of the data (p). Note that for all datasets provided in this paper except the Olivetti face dataset, the number of data in the training set is larger than the dimensionality of data (refer to Table I). For the face dataset, it is worth noting that a dictionary as small as 32 atoms leads to extremely good results using the proposed SDL, and overcompleteness is not necessary here.

Another advantage of the proposed approach is that there is only one parameter to be tuned, which is the regularization parameter λ in soft thresholding. Since the dictionary is learned in closed form, it is extremely fast to tune this parameter within the classification task or by minimizing the reconstruction error. Other SDL approaches in the literature usually have several parameters to be tuned, and since learning the dictionary and coefficients have to be performed alternately and iteratively, it is very time-consuming to tune these parameters using a cross-validation on the training set.

Through some experiments, we showed that our proposed approach is less sensitive to noisy labels compared to other SDL approaches. It was also shown that by increasing the number of atoms in the dictionary, the proposed approach becomes less sensitive to noisy labels.

In this research, we proposed to use $\mathbf{L} = \mathbf{Y}^\top \mathbf{Y} + \mathbf{I}$ as the kernel on the labels. As proposed in [65], [66], it is possible to encode the relationship among the classes into a matrix $\mathbf{M} \in \mathbb{R}^{c \times c}$, where c is the number of classes, and use $\mathbf{L} = \mathbf{Y}^\top \mathbf{M} \mathbf{Y} + \mathbf{I}$ instead to build up the kernel on the labels. This may consequently better encode the data structure into the learning of the dictionary, and also reduce the sensitivity of the proposed approach to noisy labels. As a future work, we will implement this new kernel in the formulation provided for Algorithm 1.

Also, the kernel $\mathbf{L}$ is a general kernel over the labels. However, to avoid the need for tuning the kernel on different datasets and for consistency, we limited ourselves to linear kernels in this paper. In future work, we will show how by proper selection of the kernels over the labels, we can benefit the most from the supervised dictionaries learned, which are particularly designed for a specific application in hand.

Moreover, we have used an SVM with an RBF kernel on the sparse coefficients learned for performing the classification task. However, model selection is still an open research problem [67]. For example, the RBF kernel may not fully utilize the sparsity of the coefficients. In future work, we will consider other kernels for the SVM or other classifiers that can benefit more from the sparse nature of data points submitted for classification, as suggested in [58].

ACKNOWLEDGMENT

The authors gratefully acknowledge the comment by the reviewer who pointed out the separability problem and consequently closed-form solution on the coefficients.

REFERENCES

[1] S. Mallat, *A Wavelet Tour of signal Processing: The Sparse Way*, 3rd ed. Academic Press, 2009.

[2] B. Olshausen and D. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, Mar. 1996.

[3] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: John Wiley & Sons, 2001.

[4] R. Tibshirani, "Regression shrinkage and selection via the *lasso*," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[5] D. Lee and H. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, Oct. 1999.

[6] M. Biggs, A. Ghodsi, and S. Vavasis, "Nonnegative matrix factorization via rank-one downdate," in *Proceedings of the* 25th *International Conference on Machine Learning (ICML)*, 2008, pp. 64–71.

[7] C. Jutten and J. Herault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, no. 1, pp. 1–10, 1991.

[8] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[9] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[10] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in 27th *Asilomar Conference on Signals, Systems and Computers*, 1993, pp. 40–44.

[11] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[12] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, June 2010.

[13] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, vol. 51, no. 1, pp. 34–81, 2009.

[14] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York: Springer, 2010.

[15] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *Advances in Neural Information Processing Systems (NIPS)*, 2008, pp. 1033–1040.

[16] M. J. Gangeh, A. Ghodsi, and M. S. Kamel, "Dictionary learning in texture classification," in *Proceedings of the* 8th *international conference on Image analysis and recognition - Volume Part I*. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 335–343.

[17] M. Varma and A. Zisserman, "A statistical approach to material classification using image patch exemplars," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 2032–2047, Nov. 2009.

[18] C. Zhong, Z. Sun, and T. Tan, "Robust 3D face recognition using learned visual codebook," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–6.

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

[20] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in 13th *IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 543–550.

[21] 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, Dec. 2006.

[22] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, Jan. 2008.

[23] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf, "Measuring statistical dependence with hilbert-schmidt norms," in *Proceedings of the* 16th *international conference on Algorithmic Learning Theory (ALT)*, 2005, pp. 63–77.

[24] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf, "Kernel methods for measuring independence," *Journal of Machine Learning Research*, vol. 6, pp. 2075–2129, Dec. 2005.

[25] R. Cilibrasi and P. Vitányi, "Clustering by compression," *IEEE Trans. Information Theory*, vol. 51, no. 4, pp. 1523–1545, 2005.

[26] M. J. Gangeh, A. Ghodsi, and M. S. Kamel, "Supervised texture classification using a novel compression-based similarity measure," in *Proceedings of the International Conference on Computer Vision and Graphics (ICCVG)*, 2012.

[27] K. Huang and S. Aviyente, "Sparse representation for signal classification," in *Advances in Neural Information Processing Systems (NIPS)*, 2007, pp. 609–616.

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

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

[30] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, Feb. 2010.

[31] T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *International Journal of Computer Vision*, vol. 43, pp. 29–44, June 2001.

[32] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *International Journal of Computer Vision: Special Issue on Texture Analysis and Synthesis*, vol. 62, no. 1-2, pp. 61–81, 2005.

[33] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Metaface learning for sparse representation based face recognition," in 17th *IEEE International Conference on Image Processing (ICIP)*, 2010, pp. 1601–1604.

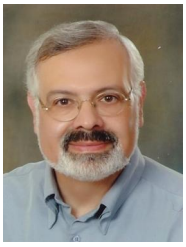
[34] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 3501–3508.

[35] B. Fulkerson, A. Vedaldi, and S. Soatto, "Localizing objects with smart dictionaries," in *Proceedings of the* 10th *European Conference on Computer Vision (ECCV): Part I*, 2008, pp. 179–192.

[36] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in 10th *IEEE International Conference on Computer Vision (ICCV)*, 2005, pp. 1800–1807.
[37] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.
[38] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 2691–2698.
[39] D.-S. Pham and S. Venkatesh, "Joint learning and dictionary construction for pattern recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.
[40] L. Yang, R. Jin, R. Sukthankar, and F. Jurie, "Unifying discriminative visual codebook generation with classifier training for object category recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.
[41] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 791–804, Apr. 2012.
[42] S. Lazebnik and M. Raginsky, "Supervised learning of quantizer codebooks by information loss minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 7, pp. 1294–1309, July 2009.
[43] F. Moosmann, B. Triggs, and F. Jurie, "Fast discriminative visual codebooks using randomized clustering forests," in *Advances in Neural Information Processing Systems (NIPS)*, 2006, pp. 985–992.
[44] F. Rodriguez and G. Sapiro, "Sparse representations for image classification: learning discriminative and reconstructive non-parametric dictionaries," in *IMA Preprint Series 2213*, 2007.
[45] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
[46] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York: John Wiley & Sons, 2006.
[47] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
[48] F. R. Bach and M. I. Jordan, "Kernel independent component analysis," *Journal of Machine Learning Research*, vol. 3, pp. 1–48, 2002.
[49] L. Song, J. Bedo, K. Borgwardt, A. Gretton, and A. Smola, "Gene selection via the bahsic family of algorithms," *Bioinformatics*, vol. 23, pp. i490–i498, July 2007.
[50] H. Shen, S. Jegelka, and A. Gretton, "Fast kernel-based independent component analysis," *IEEE Transactions on Signal Processing*, vol. 57, no. 9, pp. 3498–3511, Sept. 2009.
[51] N. Quadrianto, A. Smola, L. Song, and T. Tuytelaars, "Kernelized sorting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1809–1821, Oct. 2010.
[52] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample problem," Max Planck Institute for Biological Cybernetics, Technical Report 157, Apr. 2008.
[53] S. Jegelka, A. Gretton, B. Schölkopf, B. K. Sriperumbudur, and U. Von Luxburg, "Generalized clustering via kernel embeddings," in *Proceedings of the* 32nd *Annual German Conference on Advances in Artificial Intelligence*, 2009, pp. 144–152.
[54] H. Lütkepohl, *Handbook of Matrices*. John Wiley & Sons, 1996.
[55] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1200–1224, 1995.
[56] J. Friedman, T. Hastie, H. Hofling, and R. Tibshirani, "Pathwise coordinate optimization," *The Annals of Applied Statistics*, vol. 1, no. 2, pp. 302–332, 2007.
[57] J. L. Alperin, *Local Representation Theory: Modular Representations as an Introduction to the Local Representation Theory of Finite Groups*. New York: Cambridge University Press, 1986.
[58] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: transfer learning from unlabeled data," in *Proceedings of the* 24th *international conference on Machine learning (ICML)*, 2007, pp. 759–766.
[59] *Cambridge University Computer Laboratory, Olivetti Face Dataset AT&T*, 1994, http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.
[60] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," Dept. of Computer Science, Technion, Technical Report, 2008.
[61] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in Neural Information Processing Systems*. Morgan Kaufmann, 1990, pp. 396–404.
[62] M. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. LeCun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.
[63] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems (NIPS)*, 2004, pp. 1601–1608.
[64] B. Campana and E. Keogh, "A compression-based distance measure for texture," *Statistical Analysis and Data Mining*, vol. 3, no. 6, pp. 381–398, 2010.
[65] M. Blaschko and A. Gretton, "Taxonomy inference using kernel dependence measures," Max Planck Institute for Biological Cybernetics, Technical Report 181, Nov. 2008.
[66] L. Song, A. Smola, A. Gretton, and K. Borgwardt, "A dependence maximization view of clustering," in *Proceedings of the* 24th *international conference on Machine learning (ICML)*, 2007, pp. 815–822.
[67] D. Anguita, A. Ghio, L. Oneto, and S. Ridella, "In-sample and out-of-sample model selection and error estimation for support vector machines," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 9, pp. 1390 –1406, Sept. 2012.

Mehrdad J. Gangeh (M'05) received the Ph.D. degree in Electrical and Computer Engineering from University of Waterloo, Canada, in 2013. He is now a postdoctoral fellow jointly at the Dept. of Medical Biophysics, University of Toronto and the Dept. of Radiation Oncology, Odette Cancer Center, Sunnybrook Health Sciences Center. His current research is on developing machine learning algorithms for the assessment of cancer therapy effects using quantitative ultrasound spectroscopy. Before undertaking his Ph.D. studies, he was a lecturer at Multimedia University, Malaysia between 2000 and 2008, during which he established a joint research on scale space image analysis with the Dept. of Biomedical Engineering, Eindhoven University of Technology, and collaboration with Pattern Recognition Lab, Delft University of Technology in 2005. Dr. Gangeh's research interests are on multiview learning, dictionary learning and sparse representation, and kernel methods with the applications to medical imaging and texture analysis.

Ali Ghodsi is an Associate Professor in the Department of Statistics at the University of Waterloo. He is also cross-appointed with the school of Computer Science, a member of the Center for Computational Mathematics in Industry and Commerce and the Artificial Intelligence Research Group at the University of Waterloo. His research involves applying statistical machine-learning methods to dimensionality reduction, pattern recognition, and bioinformatics problems. Dr. Ghodsi's research spans a variety of areas in computational statistics. He studies theoretical frameworks and develops new machine learning algorithms for analyzing large-scale datasets, with applications to bioinformatics, data mining, pattern recognition, and sequential decision making.



Mohamed S. Kamel (S'74-M'80-SM'95-F'05) received the B.Sc. (Hons) EE (Alexandria University), M.A.Sc. (McMaster University), Ph.D. (University of Toronto). He joined the University of Waterloo, Canada in 1985 where he is at present Professor and Director of the Center for Pattern Analysis and Machine Intelligence at the Department of Electrical and Computer Engineering. Professor Kamel currently holds University Research Chair. Dr. Kamel's research interests are in Computational Intelligence, Pattern Recognition, Machine Learning and Cooperative Intelligent Systems. He has authored and co-authored over 500 papers in journals and conference proceedings, 13 edited volumes, 16 chapters in edited books, 4 patents and numerous technical and industrial project reports. Under his supervision, 88 Ph.D and M.A.SC students have completed their degrees. Dr. Kamel is member of ACM, PEO, Fellow of IEEE, Fellow of the Engineering Institute of Canada (EIC), Fellow of the Canadian Academy of Engineering (CAE) and Fellow of the International Association of Pattern Recognition (IAPR).