Support Discrimination Dictionary Learning for Image Classification

Yang Liu [1], Wei Chen [2,1], Qingchao Chen [3], Ian Wassell [1]

[1] Computer Laboratory, University of Cambridge
[2] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University
[3] Department of Electronic and Electrical Engineering, University College London
{yl504, wc253, ijw24}@cam.ac.uk, qingchao.chen.13@ucl.ac.uk

Abstract. Dictionary learning has been successfully applied in image classification. However, many dictionary learning methods that encode only a single image at a time while training, ignore correlation and other useful information contained within the entire training set. In this paper, we propose a new principle that uses the support of the coefficients to measure the similarity between the pairs of coefficients, instead of using Euclidian distance directly. More specifically, we proposed a support discrimination dictionary learning method, which finds a dictionary under which the coefficients of images from the same class have a common sparse structure while the size of the overlapped signal support of different classes is minimised. In addition, adopting a shared dictionary in a multi-task learning setting, this method can find the number and position of associated dictionary atoms for each class automatically by using structured sparsity on a group of images. The proposed model is extensively evaluated using various image datasets, and it shows superior performance to many state-of-the-art dictionary learning methods.

1 Introduction

Sparse representation has been successfully applied to a variety of problems in image processing and computer vision, e.g., image denoising, image restoration and image classification. In the framework of sparse representation, an image can be represented as a linear combination of a few bases selected sparsely from an over-complete dictionary. The dictionaries can be predefined by the use of some off-the-shelf basis, such as the Discrete Fourier Transform (DFT) matrix and the wavelet matrix. However, it has been shown that learning the dictionary from the training data enables a more sparse representation of the image in comparison to using a predefined one, which can lead to improved performance in the reconstruction task. Some typical reconstruction dictionary learning methods include the Method of optimal direction (MOD) [1], and K-SVD [2].

Sparse representation has also been considered in pattern recognition applications. For example, it has been used in the Sparse representation classifier (SRC) [3], which achieves competitive recognition performance in face recognition. In contrast to image reconstruction which only concerns the sparse representation of an image, in pattern recognition, the main goal is to find the correct label for the query sample, consequently the discriminative capability of the learned dictionary is crucial. A variety of discriminative dictionary learning methods have recently been proposed, that involve two different strategies.

One strategy is to learn a class-specific dictionary, which discriminates different classes of images via a sparse representation residual. Instead of learning a dictionary shared by all classes, it seeks to learn a sub-dictionary for each class. Yang et al. [4] first sought to learn a dictionary for each class, and applied it to image classification. In [5], instead of considering the dictionary atoms individually at the sparse coding stage, the atoms are selected in groups according to some priors to guarantee the block sparse structure of each coding coefficient. In [6], a group-structured dirty model is used to achieve a hierarchical structure of each coding coefficient via estimating a superposition of two coding coefficients and regularising them differently. It is worth noting that the multi-task setting is adopted in [6]. However, the sub-dictionaries in all these methods are disjoint to each other, and how many and which atoms belong to each class is fixed during the entire dictionary learning process. In addition, although class-specific setting of the dictionary works well when the number of training samples in each class is sufficient, it is not scalable to the problem with a large number of classes.

Another strategy is to learn a dictionary that is shared by all classes. Commonly, a classifier based on the coding vectors is learned together with the shared dictionary by imposing some class-specific constraints on the coding vector. Rodriguez et al. [7] proposed that samples of the same class should have similar sparse coding vectors which are achieved by using linear discriminant analysis. Yang et.al. [8] proposed Fisher discrimination dictionary learning (FDDL) where the Fisher discrimination criterion is imposed on the coding vectors to enhance class discrimination. Cai et.al. proposed support vector guided dictionary learning methods (SVGDL) [9], which is a generalised model of FDDL, that considers the squared distances between all pairs of coding vectors. In all these methods, the similarity between two coding vectors is measured by the Euclidean distance, which allows two images of different classes to be represented by using the same set of dictionary atoms. To our knowledge so far, no multi-task setting has been used in the shared dictionary, since it is difficult to discriminate groups of coefficients between different classes owing to the lack of prior knowledge concerning subdictionary structure.

In recent years, it has been shown that adding structural constraints to the supports of coding vectors can result in improved representation robustness and better signal interpretation [10][11][12]. In this paper, the multi-task setting adopts a shared dictionary, however, instead of learning the dictionary with discrimination based on the Euclidean distance between the coefficients for different classes, we consider a different principle: **The support of the coding vectors from one class should be similar, while the support of the coding vectors from different classes should be dissimilar.** Here the support of a coding vector denotes the indices of the non-zero elements of the image sparse representation under some dictionary.

More specifically, we propose a support discrimination dictionary learning method (SDDL), that finds a dictionary under which the coefficients of images from the same class have a common sparse structure while the size of the overlapped signal support of different classes is minimised. Informed by the multitask learning framework [13], and the multiple measurement vector (MMV) model [14] in the signal processing field, an effective way to encourage a group of signals to share the same support is to simultaneously encode those samples. Based on this idea, we encode multiple images from the same class, requiring that their coefficient matrix is largely 'row sparse', where only a few rows have non-zero elements. In addition to the similarity of intra-class coding vectors, the main contribution of our work is that we also design a new discriminative term to guarantee the dissimilarity of inter-class coding vectors by reducing the overlapped signal support from different classes. This can be achieved by minisation of the ℓ_0 norm of the Hadamard product between any pair of coefficients in different classes. An iterative reweighting scheme that produces more focal estimates is adopted as the optimization progresses.

The SDDL provides the following advantages. Fistly, the previous multi-task setting based dictionary learning methods all use disjoint sub-dictionaries, in which how many and which atoms belong to each class is fixed during the entire dictionary learning process. In contrast, a multi-task setting using a shared dictionary is adopted in SDDL. Our approach can automatically identify overlapped sub-dictionaries for different classes, where the size of each sub-dictionary is adjusted appropriately during the learning process to suit the training dataset. Furthermore, our approach is scalable to allow for a large number of classes, while the previous sub-dictionary based approaches cannot. Secondly, instead of using the Euclidean distance to measure the similarity and dissimilarity between different coefficients, we achieve discrimination via the support. The structural sparse constraints eases the difficulty in solving the ill-posed inverse problem in comparison to the conventional element-sparse structure [15]. The superior performance of the proposed approach in comparison to the state-of-art is demonstrated using both face and object datasets.

The paper is organised as follows. In section 2, we propose the novel support discrimination dictionary learning method for classification, including the optimisation algorithm and the classification scheme. In section 3, extensive experiments are performed on both face and object datasets to compare the proposed method with other state-of-art dictionary learning methods. Conclusions are drawn in section 4.

2 Support Discrimination Dictionary Learning

2.1 Problem Formulation

Assume that $\boldsymbol{x} \in \mathbb{R}^m$ is a m dimensional image with class label $c \in \{1, 2, ..., C\}$, where C denotes the number of classes. The training set with n images is denoted as $\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_n}] = [\boldsymbol{X_1}, \boldsymbol{X_2}, ..., \boldsymbol{X_C}] \in \mathbb{R}^{m \times n}$, where $\boldsymbol{X_c}$ includes n_c training images of class c. The learned dictionary is denoted by $\boldsymbol{D} =$

$[\boldsymbol{d_1}, \boldsymbol{d_2}, ..., \boldsymbol{d_K}] \in \mathbb{R}^{m \times K} (K < n)$, where $\boldsymbol{d_k}$ denotes the k^{th} atom of the dictionary. $\boldsymbol{A} = [\boldsymbol{A_1}, \boldsymbol{A_2}, ..., \boldsymbol{A_C}] = [\boldsymbol{a_1}, \boldsymbol{a_2}, ..., \boldsymbol{a_n}] \in \mathbb{R}^{K \times n}$ are the coding coefficients of $\boldsymbol{X}$ over $\boldsymbol{D}$. Our dictionary learning problem can be described as

$$\min_{\boldsymbol{D},\boldsymbol{A}} R(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{A}) + w_1 g(\boldsymbol{A}) + w_2 f(\boldsymbol{A}), \tag{1}$$

where $R(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{A})$ denotes the reconstruction residuals for all the images $\boldsymbol{X}$ with the sparse representation matrix $\boldsymbol{A}$ under the dictionary $\boldsymbol{D}$, $g(\boldsymbol{A})$ is a regulariser to promote intra-class similarity, $f(\boldsymbol{A})$ is the inter-class discriminative term based on the coding vectors $\boldsymbol{A}$, and $w_1 > 0$ and $w_2 > 0$ denote the weights for the final two terms in (1). In this optimisation problem, we learn a single dictionary shared among all classes while exploring the discrimination of the coding vectors.

In a common multi-task learning setting, a group of tasks share certain aspects of some underlying distribution. Here we assume the intra-class coding vectors share a similar sparse structure. In our fomulation, we use the joint sparsity regulatisation ℓ_p/ℓ_q norm of a coefficient matrix corresponding to one class, rather than encoding each training image separately. More specificaly, we set $p = 2, q = 0$, which means that the intra-class coefficient matrix should be 'row sparse', i.e., where each row is either all zero or mostly non-zero, and the number of non-zero rows is low. In this way, we can find the shared nonzero supports for each class automatically, rather than predefining their number and position. However, minimizing the ℓ_2/ℓ_0 norm is NP hard, so in this paper, we use ℓ_2/ℓ_1 norm instead. In this way, we can design a regulariser to promote intra-class similarity as:

$$g(\boldsymbol{A}) = \sum_{i=1}^{C} \|\boldsymbol{A_i}\|_{2,1} = \sum_{i=1}^{C} \sum_{k=1}^{K} \left\|\boldsymbol{a^{(ik)}}\right\|_2, \tag{2}$$

where $\boldsymbol{A_i}$ represents the coefficient matrix for the i^{th} class and $\boldsymbol{a^{(ik)}}$ denotes the k^{th} row of coefficient matrix $\boldsymbol{A_i}$.

In general, discrimination for different classes can be assessed by the similarity of the intra-class coding vectors and the dissimilarity of inter-class ones. As mentioned previously, the similarity of intra-class coding vectors is promoted by the ℓ_2/ℓ_1 regulariser. To encourage dissimilarity of the inter-class coding vectors, we design the following discriminative term:

$$f(\boldsymbol{A}) = \sum_{i=1}^{C} \sum_{p} \sum_{q} \left\|\boldsymbol{a_{i,p}} \circ \boldsymbol{a_{/i,q}}\right\|_0, \tag{3}$$

where $\circ$ denotes the Hadamard (elementwise) product between two vectors $\boldsymbol{a_{i,p}}$ and $\boldsymbol{a_{/i,q}}$, where $\boldsymbol{a_{i,p}}$ and $\boldsymbol{a_{/i,q}}$ are the p^{th} column of $\boldsymbol{A_i}$ and the q^{th} column of $\boldsymbol{A_{/i}}$ respectively. $\boldsymbol{A_i} \in \mathbb{R}^{K \times n_i}$ represents the coefficient matrix for the i^{th} class, while $\boldsymbol{A_{/i}} \in \mathbb{R}^{K \times (n - n_i)}$ denotes a sub-matrix of $\boldsymbol{A} \in \mathbb{R}^{K \times n}$ without the columns in $\boldsymbol{A_i}$. Alternatively, the value of $\left\|\boldsymbol{a_{i,p}} \circ \boldsymbol{a_{/i,q}}\right\|_0$ is the size of the overlapped support between the p^{th} image of the i^{th} class and the q^{th} image that is not in class i. Therefore, $f(\boldsymbol{A})$ denotes the summation of overlapped supports between images in different classes. However, minimising $f(\boldsymbol{A})$ in Eq.(3) is an NP hard problem. Enlightened by many recent sparse approximation algorithms that rely on iterative reweighting schemes [16][17][18] to produce more focal estimates as optimization progresses, we use the iterative reweighted ℓ_2 minimization to approximate the ℓ_0 norm.

We use the vector $\boldsymbol{a}^{\odot 2}$ to represent the element by element square of vector $\boldsymbol{a}$, which equals to $\boldsymbol{a} \circ \boldsymbol{a}$. We define the weight term $\boldsymbol{w_{p,q}}$ for a given pair of coefficient $(\boldsymbol{a_{i,p}}, \boldsymbol{a_{/i,q}})$ at each iteration as a function of those coefficients from the previous iteration as

$$\boldsymbol{w_{i,p,q}} = \frac{1}{(\boldsymbol{a}'_{\boldsymbol{i,p}} \circ \boldsymbol{a}'_{\boldsymbol{/i,q}})^{\odot 2} + \epsilon} \tag{4}$$

where $\boldsymbol{a}'_{\boldsymbol{i,p}}$ and $\boldsymbol{a}'_{\boldsymbol{/i,q}}$ are the coefficients from the previous iteration and ϵ is a regularization factor that is reduced to zero as the number of iterations increases. In this case, the descrimination term $f(\boldsymbol{A})$ can be rewritten as

$$\begin{aligned} f(\boldsymbol{A}) &= \sum_{i=1}^{C} \sum_{p} \sum_{q} \left\| \boldsymbol{a_{i,p}} \circ \boldsymbol{a_{/i,q}} \right\|_0 = \sum_{i=1}^{C} \sum_{p} \sum_{q} \sum_{k} w_{i,p,q}^{(k)} \cdot (a_{i,p}^{(k)} \circ a_{/i,q}^{(k)})^2 \\ &= \sum_{i=1}^{C} \sum_{p} \sum_{q} \sum_{k} [w_{i,p,q}^{(k)} \cdot (a_{/i,q}^{(k)})^2] \circ (a_{i,p}^{(k)})^2 \\ &= \sum_{i=1}^{C} \sum_{p} \sum_{q} diag([\boldsymbol{w_{i,p,q}} \circ (\boldsymbol{a_{/i,q}})^{\odot 2}] \cdot (\boldsymbol{a_{i,p}})^{\odot 2} = \sum_{i=1}^{C} \sum_{p} \| \boldsymbol{\Omega_{i,p}} \boldsymbol{a_{i,p}} \|_F^2 , \end{aligned} \tag{5}$$

where k represents the index of the corresponding vector and

$$\boldsymbol{\Omega_{i,p}} = diag(\sqrt{\sum_{q} (\sqrt{\boldsymbol{w_{i,p,q}}} \circ \boldsymbol{a_{/i,q}})^{\odot 2}}). \tag{6}$$

However, minimising the above $f(\boldsymbol{A})$ is both time and memory consuming since we need to calculate a weight vector $\boldsymbol{w_{i,p,q}}$ and thus a distinct weight matrix $\boldsymbol{\Omega_{i,p}}$ for each $\boldsymbol{a_{i,p}}$. Considering the effect of the ℓ_2/ℓ_1 regulariser, different coefficients in the same class should have a similar sparse pattern, hence we use the average $(\tilde{\boldsymbol{a}}'_{\boldsymbol{i}})^{\odot 2}$ instead of $(\boldsymbol{a}'_{\boldsymbol{i,p}})^{\odot 2}$ in Eq.(4), where

$$\forall p, \;\; (\boldsymbol{a}'_{\boldsymbol{i,p}})^{\odot 2} \approx (\tilde{\boldsymbol{a}}'_{\boldsymbol{i}})^{\odot 2} = \sum_{p} (\boldsymbol{a}'_{\boldsymbol{i,p}})^{\odot 2} / n_i. \tag{7}$$

That is, all p images of the class i share the same weight vector $\boldsymbol{w_{\tilde{i},q}}$ as

$$\boldsymbol{w_{\tilde{i},q}} = \frac{1}{(\tilde{\boldsymbol{a}}'_{\boldsymbol{i}})^{\odot 2} \circ (\boldsymbol{a}'_{\boldsymbol{/i,q}})^{\odot 2} + \epsilon}. \tag{8}$$

Finally Eq.(5) can be rewritten as:

$$f(\boldsymbol{A}) = \sum_{i=1}^{C} \sum_{p} \left\| \boldsymbol{\Omega_{i,p} a_{i,p}} \right\|_F^2 = \sum_{i=1}^{C} \left\| \boldsymbol{\tilde{\Omega}_i A_i} \right\|_F^2, \tag{9}$$

where

$$\boldsymbol{\tilde{\Omega}_i} = diag(\sqrt{\sum_q (\sqrt{\boldsymbol{w_{\tilde{i},q}}} \circ \boldsymbol{a_{/i,q}})^2}). \tag{10}$$

By substituting the discrimination term given by Eq.(9) into (1), we can rewrite the dictionary learning problem as

$$\min_{D,A} \sum_{i=1}^{C} \left\| \boldsymbol{X_i} - \boldsymbol{D A_i} \right\|_F^2 + w_1 \left\| \boldsymbol{A_i} \right\|_{2,1} + w_2 \left\| \boldsymbol{\tilde{\Omega}_i A_i} \right\|_F^2. \tag{11}$$

Although the objective function in (11) is not jointly convex to $(\boldsymbol{D}, \boldsymbol{A})$, it is convex with respect to $\boldsymbol{D}$ and $\boldsymbol{A}$ when the other is fixed. In the sequel, we provide an algorithm which alternately optimises $\boldsymbol{D}$ and $\boldsymbol{A}$.

2.2 Optimisation

Finding the solution of the optimisation problem in (11) involves two subproblems, i.e., to update the coding coefficients $\boldsymbol{A}$ with fixed $\boldsymbol{D}$, and to update $\boldsymbol{D}$ with fixed coefficients $\boldsymbol{A}$.

First suppose that $\boldsymbol{D}$ is fixed, and the optimisation problem can be reduced to a sparse coding problem to calculate $\boldsymbol{A} = [\boldsymbol{A_1}, \boldsymbol{A_2}, .., \boldsymbol{A_C}]$ with two constraints. Here we compute the coefficients matrix $\boldsymbol{A_i}$ class by class. More specifically, all $\boldsymbol{A_j (j \neq i)}$ are fixed thus $\boldsymbol{\tilde{\Omega}_i}$ is fixed when computing the $\boldsymbol{A_i}$. In this way, the objective function can be further reduced to

$$\min_{\boldsymbol{A_i}} \left\| \boldsymbol{X_i} - \boldsymbol{D A_i} \right\|_F^2 + w_1 \left\| \boldsymbol{A_i} \right\|_{2,1} + w_2 \left\| \boldsymbol{\tilde{\Omega}_i A_i} \right\|_F^2. \tag{12}$$

We choose the alternating direction method of multipliers (ADMM) as the optimisation approach because of its simplicity, efficiency and robustness [15][19][20]. By introducing one auxiliary variable $\boldsymbol{Z_i} = \boldsymbol{A_i} \in \mathbb{R}^{K \times n_c}$, this problem can be reformulated as

$$\min_{\boldsymbol{A_i}, \boldsymbol{Z_i}} \left\| \boldsymbol{X_i} - \boldsymbol{D A_i} \right\|_F^2 + w_1 \left\| \boldsymbol{Z_i} \right\|_{2,1} + w_2 \left\| \boldsymbol{\tilde{\Omega}_i A_i} \right\|_F^2 \quad s.t. \ \boldsymbol{A_i} - \boldsymbol{Z_i} = 0. \tag{13}$$

Therefore, the augmented Lagrangian function with respect to $\boldsymbol{A_i}, \boldsymbol{Z_i}$ can be formed as

$$\begin{aligned} L_u(\boldsymbol{A_i}, \boldsymbol{Z_i}) = {} & \left\| \boldsymbol{X_i} - \boldsymbol{D A_i} \right\|_F^2 + w_1 \left\| \boldsymbol{Z_i} \right\|_{2,1} + w_2 \left\| \boldsymbol{\tilde{\Omega}_i A_i} \right\|_F^2 \\ & - \boldsymbol{\Lambda_1^T} (\boldsymbol{Z_i} - \boldsymbol{A_i}) + \frac{u_1}{2} \left\| \boldsymbol{Z_i} - \boldsymbol{A_i} \right\|_2^2, \end{aligned} \tag{14}$$

Algorithm 1: Sparse coding using ADMM

```
Input: Training Data $\boldsymbol{X} \in \mathbb{R}^{m \times n}$, learned dictionary $\boldsymbol{D} \in \mathbb{R}^{m \times K}$, Number of
       classes $C$, regularisation parameters $w_1, w_2$, penalty parameter $u_1$ and
       step length $\gamma_1$.
Initialising $\boldsymbol{A}^0 = 0, \boldsymbol{\Lambda_1^0} = 0$, Iteration number $k = 0$ ;
for $i = 1 : C$ do
    while until converge do
        Set the matrix $\tilde{\boldsymbol{\Omega}}_{\boldsymbol{i}}^{\boldsymbol{k}}$:
            $\tilde{\boldsymbol{\Omega}}_{\boldsymbol{i}}^{\boldsymbol{k}} = diag(\sqrt{\sum_q (\sqrt{\boldsymbol{w}^{\boldsymbol{k}}_{\tilde{\boldsymbol{i}},\boldsymbol{q}}} \circ \boldsymbol{a}^{\boldsymbol{k}}_{/\boldsymbol{i},\boldsymbol{q}})^2})$        (16)
        Fix $\boldsymbol{A_i}$ and update $\boldsymbol{Z_i}$ by row-wise shrinkage
            $\boldsymbol{Z_i^{k+1}} = Shrink(\boldsymbol{A_i^k} + \frac{1}{u_1}\boldsymbol{\Lambda_1^k}, \frac{1}{u_1} w_1)$        (17)
        Fix $\boldsymbol{Z_i}$ and update $\boldsymbol{A_i}$ by:
            $\boldsymbol{A_i^{k+1}} = arg \min_{A_i} L_u(\boldsymbol{A_i}, \boldsymbol{Z_i^{k+1}})$
            $= (\boldsymbol{D}^T\boldsymbol{D} + w_2 \tilde{\boldsymbol{\Omega}}_{\boldsymbol{i}}^{\boldsymbol{kT}} \tilde{\boldsymbol{\Omega}}_{\boldsymbol{i}}^{\boldsymbol{k}} + u_1\boldsymbol{I})^{-1}(\boldsymbol{D}^T\boldsymbol{X} + u_1\boldsymbol{Z_i^{k+1}} - \frac{1}{2}\boldsymbol{\Lambda_1^k})$        (18)
        Update Lagrange multipliers $\boldsymbol{\Lambda_1}$:
            $\boldsymbol{\Lambda_1^{k+1}} = \boldsymbol{\Lambda_1^k} - \gamma_1 u_1(\boldsymbol{Z_i^{k+1}} - \boldsymbol{A_i^{k+1}})$        (19)
        Increment $k$.
Output: Estimated sparse code $\boldsymbol{A}$
```

where $\boldsymbol{\Lambda_1} \in \mathbb{R}^{K \times m}$ are the Lagrangian multipliers for equality constraints and $u_1 > 0$ is a penalty parameter. The Augmented Lagrangian function can be minimised over $\boldsymbol{A_i}, \boldsymbol{Z_i}$ by fixing one variable at a time and updating the other one. The entire procedure is summarised in Algorithm 1. The *Shrink* function in Eq.(17) updates $\boldsymbol{Z_i}$ by using row-wise shrinkage, which can be represented as

$$\boldsymbol{z^r} = max\{\|\boldsymbol{q^r}\|_2 - \frac{w_1}{u_1}, 0\}\frac{\boldsymbol{q^r}}{\|\boldsymbol{q^r}\|_2}, r = 1, ..., K, \tag{15}$$

where $\boldsymbol{q^r} = \boldsymbol{a^r} + \frac{\boldsymbol{\lambda_1^r}}{u_1}$ and $\boldsymbol{z^r}, \boldsymbol{a^r}, \boldsymbol{\lambda_1^r}$ represent the r^{th} row of matrix $\boldsymbol{Z_i}, \boldsymbol{A_i}, \boldsymbol{\Lambda_i}$ respectively.

Since the above ADMM scheme computes the exact solution for each subproblem, its convergence is guaranteed by the existing ADM theory [21][22]. After we obtain the sparse coding, we secondly update dictionary $\boldsymbol{D}$ column by column with fixed $\boldsymbol{A}$. When updating $\boldsymbol{d_i}$, all the other columns $\boldsymbol{d_j}, j \neq i$ are fixed. Now the objective function in Eq.(13) is reduced to

$$\min_{\boldsymbol{D}} \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F^2, s.t. \|\boldsymbol{d_i}\|_2 = 1. \tag{20}$$

In general, we require that each column of the dictionary $\boldsymbol{d_i}$ is a unit vector. Eq.(20) is a quadratic programming problem and it can be solved by using the K-SVD algorithm, which updates $\boldsymbol{d_i}$ atom by atom. In practice, the exact solution by K-SVD can be computationally demanding, especially when the number of training images is large. As an alternative, in the following experiments, we use the approximate KSVD to reduce the complexity of this task [23]. The detailed derivation can be found in Algorithm 5 in [24].

2.3 The Classification Scheme

After obtaining the learned dictionary $\boldsymbol{D}$, a test sample $\boldsymbol{y}$ can be classified based on its sparse coefficients over $\boldsymbol{D}$. We choose a linear classifier both for its simplicity and for the purpose of fair comparison with other dictionary learning methods, although we note that better classifier design (e.g. SRC) can potentially improve the performance further. We design the linear classifier $\boldsymbol{W} \in \mathbb{R}^{C\times K}$ as [6][25]:

$$\boldsymbol{W^T} = (\boldsymbol{A}\boldsymbol{A^T} + \eta \boldsymbol{I})^{-1}\boldsymbol{A}\boldsymbol{L^T}, \tag{21}$$

where $\boldsymbol{A} \in \mathbb{R}^{K\times n}$ is the final rounded coefficients of the training set. The matrix $\boldsymbol{L} \in \mathbb{R}^{C\times n}$ contains the label information of the training set. If the training data $\boldsymbol{x_i}$ belongs to the class c, the element $L_{c,i}$ in vector $\boldsymbol{l_i}$ is one and all the other elements in the same columns are zero. The parameter η controls the tradeoff between the classification accuracy and the smoothness of the classifier.

Next, we can compute the sparse coefficients of the each test sample y using the following objective function:

$$\min_{\boldsymbol{a}} \|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{a}\|_F^2 + w_3 \|\boldsymbol{a}\|_1, \tag{22}$$

where w_3 is a constant. Finally we apply the linear classifier $\boldsymbol{W}$ to the sparse coding of a test sample to get the label vector $\boldsymbol{l_y}$ and assigned it to the class $c = \arg\max_c \boldsymbol{l_y}$. The overall procedure is summarised in Algorithm 2.

3 Experimental Validation

In this section, we compare our proposed Support discrimination dictionary learning (SDDL) method with some other existing Dictionary learning (DL) based classification approaches. We verify the classification performance on various datasets, such as face recognition and object classification. The classification performance is measured by the percentage of correctly classified test data. The public datasets used are the Extended-Yale B Face Dataset [26], the AR Face Dataset [27] and the Caltech 101 object dataset [28]. The benchmark algorithms for comparison are the Sparse Representation Classification (SRC) [3], K-SVD

Algorithm 2: Overall Framework

```
Input: Training Data $\boldsymbol{X}$, learned dictionary $\boldsymbol{D}$, Number of classes $C$, test
       sample $\boldsymbol{y}$ and regularisation parameters $w_1, w_2, w_3$.
Initialising $k = 0$ ;
while until converge do
    Fix $\boldsymbol{D}^k$ and update $\boldsymbol{A}^{k+1}$ by Algorithm 1;
    Fix $\boldsymbol{A}^{k+1}$ and update $\boldsymbol{D}^{k+1}$ by approximate K-SVD in [24];
    Increment k.
Use $\boldsymbol{A}^{k+1}$ of $\boldsymbol{X}$ to train a linear classifier $\boldsymbol{W}$
Calculate the sparse coefficient $\boldsymbol{a_{test}}$for $\boldsymbol{y}$ by Eq.(22)
Classify the test sample $\boldsymbol{y}$ by $c = arg \max_c \boldsymbol{W} \boldsymbol{a_{test}}$ .
Output: Classification result
```

[2], Label-Consistent K-SVD (LC-KSVD) [25], Fisher Discrimination Dictionary Learning (FDDL) [8], Support Vector Guided Dictionary Learning (SVGDL) [9] and Group-structured Dirty Dictionary Learning method (GDDL) [6]. For all the competing methods, we tune their parameters for the best performance.

3.1 Parameter Selection

Dictionary size: In all experiments, the initialised dictionary is randomly selected from the training data. As shown in [25][8], the larger the size of the dictionary, the better is the performance it can achieve. The disadvantage of a large dictionary is that the problem size becomes large, which is computationally demanding. Therefore, the ideal dictionary learning method should achieve an acceptable level of performance using a relatively small size of dictionary. Here we use the Caltech 101 object dataset as an example. For each class, we randomly choose 30 images for training and the rest for testing. The number of dictionary atoms per class varies from 10 to 30. As shown in Fig.1, all the DL methods tested improve performance when the dictionary size becomes larger. Also, our proposed SDDL method achieves high classification accuracy and consistently outperforms all the other DL-based methods. The basic reason for good recognition performance, even with only a small size dictionary, is that SDDL learns a shared dictionary for all classes, while it can automatically identify sub-dictionaries for different classes, where the size of each sub-dictionary is adjusted appropriately during the learning process.

Regularisation parameters: There are 3 regularisation parameters w_1, w_2, w_3 that need to be tuned, two in the dictionary learning stage and one in the classifier. In this paper, we employ cross validation to find the regularisation parameters that give the best result.

Stopping criterion: The proposed algorithm will stop either if the values of the objective function in Eq.(11) in adjacent iterations are sufficiently close in value, or if the maximum number of iterations is reached. In Fig.2 we show empirically the value of the objective function as the number of iterations in-

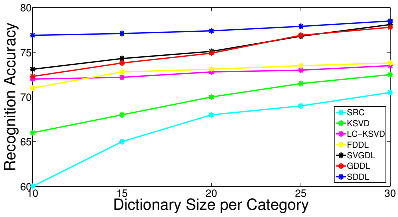

Fig. 1. Effect of dictionary size on the classification performance of various DL methods. For the Caltech 101 dataset, the size of training samples per class is fixed to 30. The dictionary atoms per class is varied from 10 to 30. As can be seen, our proposed method outperforms the other DL-based methods.

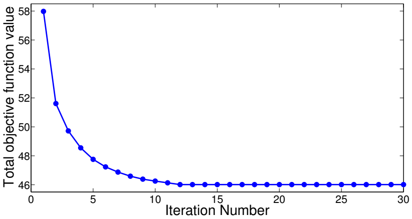

Fig. 2. The convergence curve of objective function on the AR database.

creases using the AR dataset, where we can see that the SDDL method converges rapidly.

3.2 Factors Affecting Performance

We will now investigate how the performance is affected by different factors in the proposed method using the face datasets, i.e., the Extended Yale B dataset and the AR dataset. We will discuss two factors as follows:

Factor 1: Function of the ℓ_2/ℓ_1 regularisation term

As mentioned in section 2.1, the ℓ_2/ℓ_1 regularisation term is adopted to make the coefficients from the same class share a similar sparse structure. In this section, we provide a visual illustration to see if the ℓ_2/ℓ_1 regularisation term can be truly helpful in the representation of the images from the same class. We compare the sparse codings of the same test samples from two dictionaries, where one is learned with ℓ_1 regularisation while the other with ℓ_2/ℓ_1 regularisation. Fig.3(a) shows 4 test samples of the 2^{nd} subject in the Extended Yale B database; Fig.3(b) and Fig.3(c) show the four coefficients corresponding with the two dictionaries respectively. Looking at the coefficients in Fig.3(b), in which the dictionary is learned with ℓ_1 regularisation, it can be seen that the coding vectors corresponding to the fourth image are significantly different to the other three coding vectors of the same class, which is not discriminative, owing to the poor quality of the image. However, in the Fig. 3(c), the coding

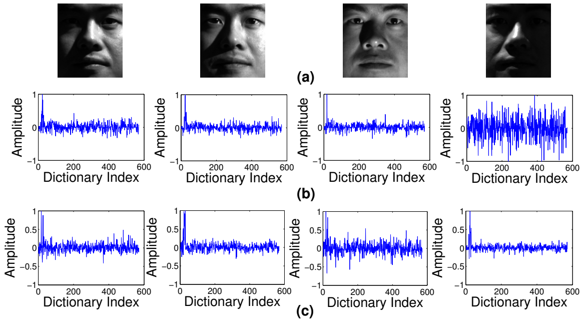

Fig. 3. An example for 4 test images and their corresponding coefficients. (a) shows 4 training samples of the 2^{nd} subject in Extended Yale B database; (b) and (c) show the four coefficients corresponding with two dictionaries, where one is learned with ℓ_1 regularisation while the other with ℓ_2/ℓ_1 regularisation respectively.

vector of the fourth image now look more similar to the other coding vectors in the class, which has a high probability of being classified correctly. A benefit of such a multi-task learning framework is that 'good quality' images help constrain the coding vector of 'poor quality' ones in the training stage. In this way, even the 'poor quality' images contribute appropriately to the dictionary update.

Factor 2: Function of the discriminative term $f(\boldsymbol{A})$

As described in section 2.1, the term $f(\boldsymbol{A})$ is utilised in the objective function to guarantee the discrimination of coding vectors from different classes. In this section, we illustrate both visually and numerically the influence of the discriminative term $f(\boldsymbol{A})$ with an example from the AR database, as shown in Fig.4 and Fig.5.

To clearly show the discrimination of coding vectors between subjects in the AR database (100 subjects in total), we calculate a symmetric scatter matrix $\boldsymbol{S} \in \mathbb{R}^{100\times100}$, in which each element S_{ij} represent the similarity between sparse codings $\boldsymbol{A_i}$, $\boldsymbol{A_j}$ of i^{th} and j^{th} subject $(i, j \in [1, 100])$:

$$S_{ij} = \sum_p \sum_q \|\boldsymbol{a_{i,p}} \circ \boldsymbol{a_{j,q}}\|_1 \, , \tag{23}$$

where $\boldsymbol{a_{i,p}}$ and $\boldsymbol{a_{j,q}}$ are the p^{th} column of $\boldsymbol{A_i}$ and the q^{th} column of $\boldsymbol{A_j}$ respectively. Following this, two scatter matrices are calculated based on the sparse codings of the same test samples from two dictionaries, where one is learned using the discriminative term while the other is not. Then for both scatter matrices, we normalise the largest element of each column or row to unity to permit comparison and plot them in Fig.4. Accordingly, the diagonal elements represent the similarity of intra-class sparse codings while the off-diagonal elements shows the similarity of the between-subject ones. We see that, the diagonal elements of both figures are the largest, and that there is obviously more between-subject similarity in Fig.4(a) than in Fig.4(b). By summing the elements in the columns

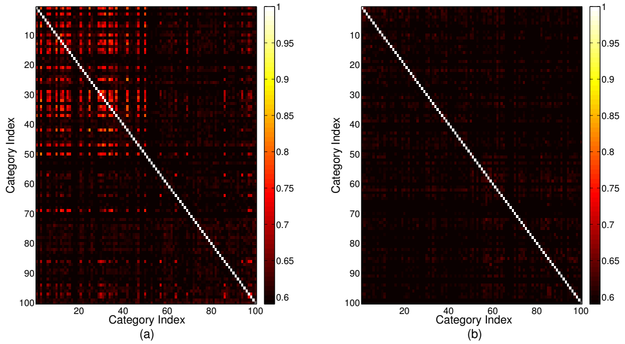

Fig. 4. Comparison between the scatter matrices calculated based on the sparse coding of the same test samples from two different dictionaries. In (a), the dictionary is learned without the discrimination term, and in (b), the dictionary is learned using the discrimination term.

of the scatter matrix to quantify the similarity index for each subject, we then plot them in Fig.5. The lower the similarity index, the less overlap there is between the pairs of coefficients between this subject and the others, i.e., the better is the discrimination of the coding coefficient. As shown in Fig.5, the red curve learned using the discrimination term is lower than the blue one learned without the discrimination term for all the 100 subjects, which shows that learning the dictionary with the help with $f(\boldsymbol{A})$ can decrease the coefficient overlap between different subjects. These visual and numerical results both show that the dictionary learned with the $f(\boldsymbol{A})$ term can significantly enhance the discrimination performance of the coefficients. We use the Extended Yale and AR face databases to illustrate how this term can help to improve classification performance. With the help with the discrimination term $f(\boldsymbol{A})$, the recognition rate for the Extended Yale B is enhanced from 96.20% to 98.50%, and the recognition rate for the AR database is increased from 95.90% to 98.00%. The experimental setting used to obtain these result will be presented fully in section 3.4.

3.3 Object Classification

The Caltech 101 dataset is one of the benchmark datasets used in object classification. It consists of 9144 images, split between 101 distinct object classes including animals, vehicles, as well as a background class. The sample from each class has significant shape variability. In the following experiments, the spatial pyramid features are used as the input for the classifier, which is the same as used in [25][8][9]. Following [25], We vary the number of training samples per class from 10 to 30. The size of the dictionary in SDDL is K=510, that is the same as the experimental setting in [9]. The experiments are carried out 10 times with differently chosen partitions. The average classification accuracy of the proposed method (SDDL) compared with other existing dictionary learning based

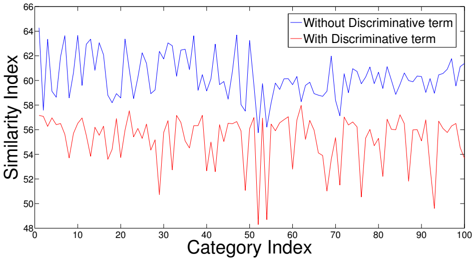

Fig. 5. The comparison between the similarity index calculated based on the sparse coding of the same test samples from two different dictionaries.The red line represents the similarity index calculated by the dictionary learned using the discrimination term, while the blue line represents the similarity index without.

Table 1. Recognition Rates (%) for Object Classification

No.Training	SRC	KSVD	LC-KSVD	FDDL	SVGDL	GDDL	SDDL
10	58.89	59.80	62.40	63.10	63.10	62.30	**66.80**
15	63.80	64.20	66.90	66.60	68.80	66.20	**71.60**
20	67.20	68.70	69.50	69.80	70.00	69.80	**73.60**
25	68.60	70.20	71.80	72.30	73.50	72.30	**76.50**
30	70.30	73.40	73.30	73.10	74.10	73.40	**76.90**

methods is shown in Table.1. The regularisation parameters for the Caltech 101 dataset are $w_1 = 0.2, w_2 = 10, w_3 = 0.05$. The DL-based methods perform better than SRC, which shows that better performance can be achieved by learning a discriminative dictionary. Our proposed method consistently outperforms the other existing DL based methods, by at least 2.8 percentage points.

3.4 Face Classification

The two benchmark face datasets are the Extended Yale B dataset and the AR dataset. With different illumination conditions and facial expressions, the Extended Yale B dataset consists of 2414 frontal images of 38 subjects (about 64 images per subject). We randomly select half as the training set and the rest as the test set for each class. As in the experimental setting in [25] [6], we crop each image to 192 × 168 pixels, and then normalise and project it to a 504 dimension vector using a random Gaussian matrix. The dictionary size of the Extended Yale B dataset is 570, which corresponds to an average of 15 atoms per subject. As discussed previously, there is no explicit correspondence between the dictionary atoms and the labels of the individual at the training stage.

Similarly, the AR face dataset consists of over 4000 images of 126 subjects, which is more challenging owing to more variation, i.e., different illumination, expressions and facial occlusion (e.g., sunglasses, scarf). As in the experimental setting in [25] [6], we use the subset of the dataset which contains 2600 images

Table 2. Recognition Rates (%) for Face Classification

Method	SRC	KSVD	LC-KSVD	FDDL	SVGDL	GDDL	SDDL
Extended Yale	80.54	93.40	94.50	94.92	95.70	96.80	**98.50**
AR	66.57	86.30	93.70	94.10	96.00	96.00	**98.00**

for 50 male and 50 female subjects. For each subject, we randomly select 20 and 6 images for training and testing respectively. We crop each image to 165 × 120 pixels, and then normalise and project it to a 540 dimension vector using a Gaussian matrix. The dictionary size of the AR dataset is 500, that corresponds to an average of 5 atoms per subject. The dictionary is shared by all subjects.

The experiments are carried out 10 times with different chosen partitions. The average classification accuracy of the proposed method compared with other existing dictionary learning based methods are shown in the Table.2. The regularisation parameters for the Extended Yale B dataset are $w_1 = 0.04, w_2 = 2, w_3 = 0.005$, and for the AR face database are $w_1 = 0.05, w_2 = 3, w_3 = 0.005$. We can see that the proposed SDDL method achieves an improvement of at least 1.7 and 2 percentage points over the next best scheme in terms of classification accuracy for the Extended Yale B and the AR datasets respectively.

4 Conclusion

We incorporate structured sparsity into the dictionary learning process and propose a support discrimination dictionary learning (SDDL) method for image classification. In contrast to other methods, we use the sparse structure, i.e., support, to measure the similarity between the pairs of coefficients, rather than the Euclidean distance which is widely adopted in many dictionary learning approaches for classification. The discrimination capability of the proposed method is enhanced in two ways. First, a row sparse regulariser is adopted so that a shared support structure for each class can be learned automatically. Second, we adopt a discriminative term to make the coefficients from different classes have minimum support overlap between each other. It can be achieved by minimisation of the ℓ_0 norm of the Hadamard product between any pair of coefficients in different classes. It worth noting that our approach can automatically identify overlapped sub-dictionaries for different classes, where the size of each sub-dictionary is adjusted appropriately during the learning process to suit the training dataset. In this way, this proposed approach is scalable to classification tasks having a large number of classes. Extensive experimental results on object recognition and face recognition demonstrate the proposed method can generate more discriminative sparse coefficients and that it has superior classification performance to a number of state-of-the-art dictionary learning based methods.

Acknowledgements. This work is supported by University of Cambridge Overseas Trust, EPSRC Research Grant (EP/K033700/1) and Natural Science Foundation of China (61401018).

References

1. Engan, K., Aase, S.O., Husøy, J.H.: Multi-frame compression: Theory and design. Signal Processing **80**(10) (2000) 2121–2140
2. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing **54**(11) (2006) 4311–4322
3. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(2) (2009) 210–227
4. Yang, J., Yu, K., Huang, T.: Supervised translation-invariant sparse coding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2010) 3517–3524
5. Elhamifar, E., Vidal, R.: Robust classification using structured sparse representation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2011) 1873–1879
6. Suo, Y., Dao, M., Tran, T., Mousavi, H., Srinivas, U., Monga, V.: Group structured dirty dictionary learning for classification. In: IEEE Transactions on Image Processing (ICIP). (2014) 150–154
7. Rodriguez, F., Sapiro, G.: Sparse representations for image classification: Learning discriminative and reconstructive non-parametric dictionaries. Technical report, DTIC Document (2008)
8. Yang, M., Zhang, L., Feng, X., Zhang, D.: Sparse representation based Fisher discrimination dictionary learning for image classification. International Journal of Computer Vision **109**(3) (2014) 209–232
9. Cai, S., Zuo, W., Zhang, L., Feng, X., Wang, P.: Support vector guided dictionary learning. In: Computer Vision–ECCV 2014. Springer (2014) 624–639
10. Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **68**(1) (2006) 49–67
11. Jenatton, R., Audibert, J.Y., Bach, F.: Structured variable selection with sparsity-inducing norms. The Journal of Machine Learning Research **12** (2011) 2777–2824
12. Eldar, Y.C., Mishali, M.: Robust recovery of signals from a structured union of subspaces. IEEE Transactions on Information Theory **55**(11) (2009) 5302–5316
13. Lounici, K., Pontil, M., Tsybakov, A.B., Van De Geer, S.: Taking advantage of sparsity in multi-task learning. arXiv preprint arXiv:0903.1468 (2009)
14. Cotter, S.F., Rao, B.D., Engan, K., Kreutz-Delgado, K.: Sparse solutions to linear inverse problems with multiple measurement vectors. IEEE Transactions on Signal Processing **53**(7) (2005) 2477–2488
15. Deng, W., Yin, W., Zhang, Y.: Group sparse optimization by alternating direction method. In: SPIE Optical Engineering+Applications, International Society for Optics and Photonics (2013) 88580R–88580R
16. Chartrand, R., Yin, W.: Iteratively reweighted algorithms for compressive sensing. In: IEEE international conference on Acoustics, speech and signal processing. (2008) 3869–3872
17. Candes, E.J., Wakin, M.B., Boyd, S.P.: Enhancing sparsity by reweighted 1 minimization. Journal of Fourier analysis and applications **14**(5-6) (2008) 877–905
18. Wipf, D., Nagarajan, S.: Iterative reweighted and methods for finding sparse solutions. IEEE Journal of Selected Topics in Signal Processing **4**(2) (2010) 317–329

19. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning **3**(1) (2011) 1–122
20. Yang, J., Zhang, Y.: Alternating direction algorithms for l1-problems in compressive sensing. SIAM journal on scientific computing **33**(1) (2011) 250–278
21. Glowinski, R., Le Tallec, P.: Augmented Lagrangian and operator-splitting methods in nonlinear mechanics. Volume 9. SIAM (1989)
22. Glowinski, R., Oden, J.: Numerical methods for nonlinear variational problems. Journal of Applied Mechanics **52** (1985) 739
23. Aharon, M., Elad, M.: Sparse and redundant modeling of image content using an image-signature-dictionary. SIAM Journal on Imaging Sciences **1**(3) (2008) 228–247
24. Rubinstein, R., Zibulevsky, M., Elad, M.: Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit. CS Technion **40**(8) (2008) 1–15
25. Jiang, Z., Lin, Z., Davis, L.S.: Label consistent K-SVD: Learning a discriminative dictionary for recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(11) (2013) 2651–2664
26. Georghiades, A.S., Belhumeur, P.N., Kriegman, D.J.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Transactions on Pattern Analysis and Machine Intelligence **23**(6) (2001) 643–660
27. Martinez, A.M.: The AR face database. CVC Technical Report **24** (1998)
28. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. Computer Vision and Image Understanding **106**(1) (2007) 59–70