International Journal of Machine Learning and Cybernetics (2020) 11:1717–1734
https://doi.org/10.1007/s13042-020-01067-w

ORIGINAL ARTICLE

Dual sparse learning via data augmentation for robust facial image classification

Shaoning Zeng[1,2] · Bob Zhang[1] · Yanghao Zhang[3] · Jianping Gou[4]

Received: 27 June 2019 / Accepted: 11 January 2020 / Published online: 29 January 2020

Abstract
Data augmentation has been utilized to improve the accuracy and robustness of face recognition algorithms. However, most of the previous studies focused on using the augmentation techniques to enlarge the feature set, while the diversity produced by the virtual samples lacked sufficient attention. In sparse dictionary learning-based face recognition, l_1-based sparse representation (SR) and SVD-based dictionary learning (DL) both have shown promising performance. How to utilize both of them in an enhanced training process by data augmentation is still unclear. This paper proposes a novel method that utilizes the sample diversity generated by data augmentation and integrates sparse representation with dictionary learning, to learn dual sparse features for robust face recognition. An additional feature set is created by applying sample augmentation via simply horizontal flipping of face images. The two sparse models, l_1-based SR and SVD-based DL, are integrated together using our new proposed objective function. Under two-level fusion of both data and classifiers, the diversity between two training sets is well learned and utilized, in three implementations, to obtain a robust face recognition. After conducting extensive experiments on some popular facial datasets, we demonstrate the proposed method can produce a higher classification accuracy than many state-of-the-art algorithms, and it can be considered as a promising option for image-based face recognition. Our code is released at GitHub.

Keywords Image classification · Sparse representation · Dictionary learning · l_1 Regularization

✉ Bob Zhang
bobzhang@um.edu.mo

Shaoning Zeng
yb77477@um.edu.mo

Yanghao Zhang
yz16n18@soton.ac.uk

Jianping Gou
goujianping@ujs.edu.cn

[1] Pattern Analysis and Machine Intelligence Group, Department of Computer and Information Science, University of Macau, Macau, China

[2] School of Computer Science and Engineering, Huizhou University, Huizhou, Guangdong, China

[3] Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

[4] College of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, Jiangsu, China

1 Introduction

Face recognition based on facial image classification is an important application of machine learning in computer vision and pattern recognition. Among different machine learning methods, Sparse Representation-based Classification (SRC) is one promising method competent to both close-set [38, 46] and open-set [5, 17] face recognition. In SRC, a sparse coefficient vector is firstly obtained from the training samples to represent the test sample. Since there are only a few of non-zero entries in the vector, the representation is so called sparse. The sparsity means that only part of the training samples will be utilized to represent the test sample. After that, the sparse model is finally determined by the l_1 normalization on the sparse coefficient vector, as well as the minimization of the reconstruction error. According to previous studies, the sparsity in the representation coefficient plays the most important role in producing a robust face recognition method [48, 54, 67]. However, the conditions of face recognition applications become more and more complicated, e.g., variational illumination and background when capturing the facial images, which in turn sets a rigorous requirement for robustness. Optimization, considering the sparsity of image representation, provides an opportunity to produce a powerful and robust classifier.

For this reason, how to improve sparse representation (SR) has become a hot topic in the last decade. First, different regularizations other than l_1 regularization are utilized to improve the sparse representation, for example, l_0 (non-convex) [68], l_q (or $l_{1/2}$) [15], l_2 [64], and even joint l_1 and l_2 [53]. All of them are looking for an efficient method to learn a robust sparse model. However, it is really hard to obtain a model with sufficient sparsity in one shot. Therefore, the second option is utilizing one or more other models to help produce a robust sparse representation. Many state-of-the-art methods, like NN [66], collaborative representation [58], and dictionary learning [26, 57], were combined with sparse representation for face recognition. On the other hand, dictionary learning-based face recognition is also one of the most successful applications related with sparse representation. Dictionary learning is usually a synonym for sparse representation in machine learning. However, there is a subtle difference between them. Sparse representation focuses on how to construct an impressive dictionary, while dictionary learning generates a dictionary from the samples. For example, dictionary learning using K-SVD showed a good performance on face recognition [65]. Considering the first proposed SRC by Wright [46] is based on l_1 regularization, we define the sparse representation using K-SVD as dictionary learning. The others based on l_1 regularization are sequentially called sparse representation. Rubinstein et al. developed an analysis of K-SVD to obtain the analyzed sparse model [35]. Xu et al. considered the sample diversity and representation effectiveness to create a new robust dictionary learning [47]. Recently, a locality-constrained and label embedding dictionary learning (LCLE-DL) obtained a promising result in face recognition [25]. These studies demonstrate that learning an optimal dictionary from training data rather than directly using the original data can lead to promising results in face recognition. Motivated by this, we believe using twofold features, from both sparse representation and dictionary learning, ought to help to produce more sufficient sparsity in representation.

Data augmentation is one of the most popular techniques to improve the accuracy and robustness of classifiers, which generates virtual samples by transforming the training data. Adding virtual samples has been proven to be helpful for neural networks for a long time [11]. It is very good at combatting the problem of overfitting [23]. For this reason, it has been applied to different domains, including face recognition [27, 42], leaves recognition [62], image classification [14], hyperspectral image classification based on the convolutional neural networks [40], etc. Furthermore, there is a powerful and open source image augmentation tool [4]. In order to gain more features, virtual samples are often generated to enrich the feature set. According to the recent trends of facial image-based pattern recognition [20, 43], plenty of virtual sample generation technologies have been proposed to achieve promising performances in face recognition [7]. Tang et al. expanded the training samples with virtual samples by randomly adding noise to the original samples, to obtain a more robust SRC [41]. Du et al. treated the multiplication of two image samples as a virtual sample, to expand the training set before running SRC [13]. Biggio et al. proposed to improve face verification via the sparse support face learned from a set of virtual faces [3]. However, most of current algorithms utilized the augmented training set only to learn more features, as shown in the first implementation (1. v-SR-DL) in Fig. 1, while the diversity between the virtual and the original has not been fully made use of.

In this paper, we propose a novel dual sparse learning method for image based face recognition, which first applies the horizontal flipping augmentation to generate virtual samples, and then integrates Sparse Representation-based Classification (SRC) with Locality-Constrained and Label Embedding Dictionary Learning (LCLE-DL) in the distance level, as shown in Fig. 1. Hereby, we refer SR as sparse representation on images for classification, while DL refers to dictionary learning for classification. Our contributions in this work are as follows. Firstly, we propose a dual sparse learning method to combine sparse representation and locality-constrained dictionary learning in the distance level, to obtain a more robust sparse model, which makes the usage of data augmentation necessary to generate a positive effort. This is a novel two-level of fusion method, including data and algorithms. Second, we implement three versions of algorithms, and consider the diversity, instead of size, of the samples as the most crucial condition (see the second and third implementations in Fig. 1). Third, we analyze the role of dual sparse learning in improving the sparsity. Our experiments, which are conducted on four benchmark facial databases, including CMU Faces [31], MUCT [30], Georgia Tech (GT) [10], LFW [19] and YouTubeFace [45], demonstrate that it outperforms state-of-the-art face recognition methods in many cases.

The structure of the following content in this paper is as follows. The related work is described in Sect. 2. In Sect. 3, we explain our proposed method. Sect. 4 demonstrates the experimental results obtained on some prevailing benchmark facial datasets. Sect. 5 gives an analysis on the improvements made by our method, and Sect. 6 concludes the paper.

2 Related work

Assume that there are C subjects or pattern classes with N training samples $a_1, a_2, \cdots, a_n$, and the test sample is y. Let matrix $A_i = \left[a_{i,1}, a_{i,2}, \cdots, a_{i,n_i}\right] \in I^{m\times n_i}$ denote n_i training

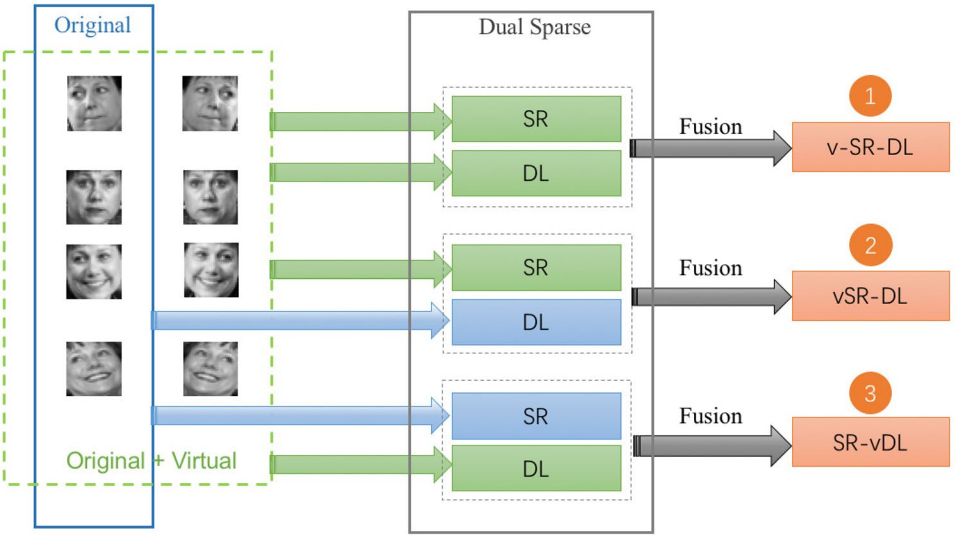

Fig. 1 Dual sparse representation via mixed synthetic samples in dictionary learning. When a set of facial images are inputted, the method will create a new set of virtual images, by horizontally flipping the original images, before forming an expanded training set via combining the original and the virtual images. Then, the original and the expanded image sets will be trained through different learning algorithms, i.e., Sparse Representation (SR) and Dictionary Learning (DL), to obtain two representations. Finally, the two representations are fused for the final robust classification. In this way, we have three optional implementations, as indicated by the circled numbers

samples from the ith class. By stacking all columns from the vector of a $w \times h$ gray-scale image, we can obtain the vector to identify this image: $a \in I^m (m = w \times h)$. Each column of A_i is then representing the training samples of the ith class.

2.1 l_1-norm based sparse representation

In SR, each test sample $a \in \Re^m$ can be represented as a linear combination of training samples [46]. Then, the sparse coefficient can be solved by an l_1 normalization,

$$(\hat{x}_i) = \arg\min_x \|x\|_1 \; s.t. \; \|y - Ax\|_2 \leq \varepsilon, \tag{1}$$

where $A = [A_1, A_2, \dots, A_C] \in \Re^{m\times n}$ is the l_2-norm of training columns, and $y \in \Re^m$ is a test sample. SR will obtain a dictionary consisting of multiple columns of data samples. If most of the entries in the coefficient vector are zeros, the representation is sparse. With this sparse coefficient, the residual between the test sample and each class can be obtained by the Euclidean distance

$$\mathbf{res}_i(y) = \|y - A_i\hat{x}_i\|_2. \tag{2}$$

Finally, the test sample can be classified to the class with the minimal distance. It is known that the sparsity plays an important role in classification [54]. The powerful feature extraction of SR makes it suitable to be applied to different types of classification tasks, e.g., facial diagnosis [61], retinal image analysis [60], and of course face recognition [56]. However, it is hard to improve the sparse representation if using only one model. Fortunately, as a distance-based classification method, SR can work with other classification methods to obtain a higher accuracy. Actually, many distance-based methods are capable of this task, such as collaborative representation [64], K-SVD [33], PCA, LDA [59], and KNN [63]. In our work, we will use the dictionary obtained by a recently proposed dictionary learning method, LCLE-DL [25], to improve SR for facial image classification.

2.2 Sparse features from dictionary learning

Dictionary learning, especially the supervised dictionary learning, shows a good performance in classification tasks due to the reconstruction error and label constraints [34]. Following the same schema in Sect. 2.1, the label matrix of the training samples A can be defined as $H = [h_1, h_2, \dots, h_N] \in \Re^{C\times N} (h_i = [0, \dots, 1, \dots, 0]^T \in \Re^C)$, and only the jth entry of h_i is non-zero, which indicates that the training sample a_i comes from the jth class. The dictionary, denoted by $D = [d_1, d_2, \dots, d_K] \in \Re^{n\times K}$, is obtained from the training samples A, where K is the number of atoms. Assume each class has the same number (f) of atoms, then $K = f \times C$. Hence, we can use the dictionary to define the samples as

$$A = d_1(\hat{x}_1)^T + \cdots + d_i(\hat{x}_i)^T + \cdots + d_j(\hat{x}_j)^T + \cdots + d_K(\hat{x}_K)^T, \tag{3}$$

Among different implementations of dictionary learning, LCLE-DL takes the locality and label information of atoms into account together in the learning process [25]. The error reconstruction is based on both locality and labels. The novel dual reconstruction terms allow the locality reconstruction and label reconstruction to be fitted at the same time. For this reason, LCLE-DL outperforms many conventional DL algorithms, e.g., the recursive least squares dictionary learning algorithm (RLS-DLA) [39], occlusion dictionary learning [32], the label consistent recursive least squares dictionary learning algorithm (LC-RLSDLA) [29], discriminative dictionary pair learning [9], etc. This is the reason why we choose LCLE-DL in our implementation. In particular, LCLE-DL utilizes the following objective function to obtain optimal sparse coefficients based on both the locality and label reconstruction.

$$\begin{aligned}\min_{(D,X,V,L)} \; & ||Y-DX||_2^2+\alpha Tr(X^TLX)+||Y-DV||_2^2 \\ & +\; \beta Tr(V^TUV)+\gamma||X-V||_2^2 s.t.\; ||d_i||^2 \\ & = 1, i=1,\dots,K,\end{aligned} \tag{4}$$

where $X \in \Re^{K\times N}$ and $V \in \Re^{K\times N}$ are the representation coefficients, $U \in \Re^{K\times K}$ is the scaled label matrix of dictionary D which will be revisited in more detail in the following sections, L is a graph Laplacian matrix, $||Y-DX||_2^2$ and $||Y-DV||_2^2$ represent the reconstruction error terms, and $||X-V||_2^2$ is a regularization term for the transformation between locality and label constraints. α, β and γ are the regularization parameters. The process in LCLE-DL has already been well explained in [25]. However, another problem is the features obtained in both SR and DL may still not be enough in some tough recognition situations.

2.3 Expanding the training set

When dealing with the problem of a small training set in face recognition, using virtual samples is a valid way to increase the feature set. For this purpose, a lot of techniques generating virtual samples have been proposed. Xu et al. used approximately symmetrical face images in SRC for face recognition [49]. Tang et al. expanded the training samples with virtual samples by randomly adding noise to the original samples, to obtain a more robust SRC [41]. Du et al. treated the multiplication of two image samples as a virtual sample, to expand the training set before running SRC [13]. The role of virtual samples is to enrich the features learned in the training set, even using the simplest form of approximately symmetrical images.

However, we believe that the diversity between virtual and original samples can also be utilized to improve the accuracy of recognition. Motivated by this, we create virtual samples by horizontally flipping, and then utilize the virtual samples in sparse representation or dictionary learning to learn dual sparse features. Our work mainly concentrates on proving the feasibility of dual sparse models by adding synthetic samples into the original training set. Although there are many sophisticated data augmentation techniques [4], like pair samples [21], the work in this paper mainly focuses on how to improve the sparse learning via virtual samples. Hence, this simplest data augmentation, horizontally flipping samples, is utilized to generate the virtual samples. Horizontally flipping the face images is an easy implementation and popular in face recognition. This is the reason why we apply it in our dual sparse learning.

3 The proposed method

The core contribution of dual sparse learning is the combination of two sparse models, l_1-based SR and SVD-based DL. The conceptualization of our proposed method can be briefly described in the following objective function

$$\arg\min_x(\|y-A'x\|_2+\lambda\|y-DX\|_2^2), \tag{5}$$

where A' is the augmented training samples, and D is the dictionary learned from the original training samples A. The SR model, denoted by the first term, produces sparse representation of the training samples, while the DL model firstly constructs a dictionary from the training samples. The SR and DL models are learned simultaneously to form the proposed dual sparse learning. We believe that these two sparse models can mutually improve each other with the help of data augmentation. The detailed implementation is going to be explained in the following subsections.

3.1 Dual sparse learning

Dual sparse learning, as the name implies, consists of two independent sparse feature learning processes. The first one is the conventional l_1-norm based sparse representation, as depicted in Sect. 2.1. In the implementations of vSR-DL and v-SR-DL, the l_1-based SR utilizes the expanded training set A', which contains the virtual samples, while DL is still performed on the original training samples, as shown in Fig. 1. The sparse coefficient can be solved as follows

$$(\hat{x}_1) = \arg\min_x\|x\|_1 \; s.t. \; \|y-A'x\|_2 \le \varepsilon, \tag{6}$$

where ε is a small error tolerance to balance the coding error of y and the solved sparsity of x. This linear problem can be solved by many l_1 regularization algorithms. In our implementations, we choose to use the Orthogonal Matching Pursuit (OMP) algorithm [12]. The reason for this choice is twofold. On the one hand, the OMP algorithm is faster and easier to implement, which only requires $O(m^2)$ measurements and reliably recovers a sparse signal [36]. On the other hand, the conventional LCLE-DL method also utilizes the OMP to solve a sparse representation coefficient with the obtained dictionary D [25]. Using the same algorithm is good to maintain consistency and reusability in the implementation. What is more, our experiments demonstrate that using OMP we can produce a higher recognition accuracy than other regularization algorithms.

From this the residual between the test sample y and the ith class becomes

$$\mathbf{res}_{sr,i}(y) = \left\|y - A'_i\hat{x}_i\right\|_2. \tag{7}$$

This is the first part of sparse learning. The second sparse learning is performed according to the LCLE-DL algorithm. The procedure to obtain one more sparse residuals based on LCLE-DL consists of three steps. The literature [25] gave the optimization of the objective function Eq. (4) in Sect. 2. In order to decrease the computational complexity, the objective function Eq. (4) can be solved iteratively.

In LCLE-DL, dictionary initialization is the first important step. The K-SVD algorithm is used to learn sub-dictionary D_i and coding coefficient X_i for the ith class training samples A_i, to compute the scaled label matrix U and initialization graph Laplacian matrix L for Eq. (4). Before doing this, it is necessary to initialize a dictionary $D^0 = [D_1, D_2, \dots, D_k]$ and a coefficient matrix $X^0 = [X_1, X_2, \dots, X_k]$. Then, a label matrix $B = [b_1, \dots, b_K]^T \in \Re^{K\times C}$ of dictionary D is obtained by using the label matrix H of training samples, and consequently the labels are embedded to the atoms. Now, we can construct a weighted label matrix G

$$G = B(B^TB)^{-1/2} \in \Re^{K\times C}. \tag{8}$$

Afterwards, we obtain the scaled label matrix $U = GG^T$ with a block-diagonal structure. The last initialization is to prepare the graph Laplacian matrix L, which enforces the locality constraint of atoms. A nearest neighbor graph M of dictionary D is constructed as

$$M_{i,j} = \begin{cases} \exp\left(-\frac{||d_i-d_j||_2}{\sigma}\right) & \text{if } d_j \in kNN(d_i) \\ 0 & \text{else } x < 0 \end{cases}, \tag{9}$$

where σ is a parameter, and $kNN(d_i)$ denotes the k-nearest neighbors of atom d_i. In this way, $M_{i,j}$ reflects the distance between atoms d_i and d_j. When d_i is connected with d_j, we can infer that the two atoms have a close distance. For the sake of better describing the locality information of the atoms, we introduce a graph Laplacian matrix L by using the nearest neighbor graph M as follows

$$L = T - M, \tag{10}$$

where $T = diag(t_1, \dots, t_k)$, and $t_i = \sum_{j=1}^{K} M_{i,j}$. With the above initializations, we can then proceed with the following steps of the LCLE-DL algorithm.

The next step is learning the representation coefficients V and X. Ignoring the constant terms in the objective function Eq. (4), we can optimize the formulation and obtain the solutions of V and X, according to [25]. For V, dropping the constant term simplifies the objective function to

$$\min_V ||A - DV||_2^2 + \beta Tr(V^TUV) + \gamma||X - V||_2^2. \tag{11}$$

Thus, we can obtain the optimal V as follows

$$V = (D^TD + \beta U + \gamma I)^{-1}(D^TA + \gamma X). \tag{12}$$

In a similar way, another coefficient matrix X can be solved as follows

$$\min_X ||A - DV||_2^2 + \alpha Tr(X^TLX) + \gamma||X - V||_2^2, \tag{13}$$

then

$$X = (D^TD + \alpha L + \gamma I)^{-1}(D^TA + \gamma V). \tag{14}$$

In the final step we learn the dictionary D and graph Laplacian matrix L. Given that all the other variables in the objective function are already solved above, Eq. (4) becomes

$$\min_D ||A - DX||_2^2 + ||A - DV||_2^2 s.t. \; ||d_i||_2 = 1, \; i = 1, \dots, K. \tag{15}$$

This is a least square problem with quadratic constraints, and can be solved by the Lagrange dual function according to [25], which also gives the detailed derivation as well. The resultant optimization of D is obtained as

$$D = A(X^T + V^T)(XX^T + VV^T)^{-1}. \tag{16}$$

So far, we briefly finished the dictionary learning process. Above deduction is focus on the process to obtain D and X, and the more detailed derivation process can be found in [25]. Similarly, in the implementation of SR-vDL and v-SR-DL, the virtual training set A' will be used instead of the original A. Therefore, two coefficients V' and X', as well as the dictionary D', are all obtained by using the expanded training set A' in Eqs. (11)–(16). In the original LCLE-DL implementation, the dictionary D and only one coefficient matrix X, not V, are used to performed classification [25]. The label vector res_i of each test sample $\hat{y}_i$ can be obtained by

$$\mathbf{res}_{dl,i} = HX^T(XX^T + I)^{-1} \cdot omp(D'\hat{x}_i, D'D, t), \tag{17}$$

where H is the embedded label matrix, $\hat{x}_i$ is the sparse coefficient vector of the test sample solved by the Orthogonal Matching Pursuit algorithm [36], and t is the sparsity to ensure convergence. LCLE-DL classifies the test sample to a class with the maximal value in this residual. It is noted that we do not modify the representation of LCLE-DL, before solving the dictionary D and the coefficient X. However, we make a novel classification, by integrating l_1-based sparse representation, to improve the final result. This is our main contribution.

3.2 Fusion of two feature sets

In order to combine two sparse feature sets, we integrate them on the level of distance. Some previous studies have proposed fusion methods that perform the integration on the coefficient level [54]. Actually, there are three possible levels for the fusion of multiple representations, including pixel level, feature/coefficient level and distance level [48]. However, weighted fusion on the distance level is an easier and faster way [52]. Therefore, we combine two residuals from two sparse models to perform our final recognition. The identification step can be denoted by

$$\mathbf{res}_i = res_{src,i} + \lambda \cdot res_{dl,i}, \tag{18}$$

where λ is the fusion factor to balance two sparse models. Here two sparse feature sets are fused into one sparse model. It is noted that the classification schemas of SR and DL are opposites of each other, minimum for SR and maximum for DL. Our experiments also demonstrated that it should be assigned a negative value in most time.

By this time, the test sample y can be classified to a class with a minimal fusion residual

$$\mathbf{identity}(y) = \arg\min_i\{res_i\}. \tag{19}$$

3.3 Multiple implementations

Based on the usage of virtual samples, the proposed dual sparse learning has three different implementations. First of all, the virtual samples are used in the training process only. There are two training steps in dual sparse learning, one for sparse representation, and the other for dictionary learning. Therefore, we have three options to use the virtual samples: (1) in SR only (vSR-DL), (2) in DL only (SR-vDL), and (3) in both SR and DL (v-SR-DL). The role of virtual samples in these implementations is twofold. In the first two implementations, vSR-DL and SR-vDL, the virtual samples are added into one model, rather than both models. This will enlarge the data diversity used in the two models. We believe that the enlarged diversity of data will help to generate a better classification. The situation in the last implementation is different, where both models are trained on an augmented training set. The role of virtual samples is expanding the training set, instead of the data diversity. One can select an implementation based on the characteristics of the specific data. If the data lacks diversity, the better choice would be the first two, otherwise the last one is better.

It is noted that the test samples in our method are kept unchanged, without adding any virtual samples. This is due to the fact that it is not reasonable to perform modifications on the test set. The virtual samples will be appended into the training set, without affecting the recognition problem. This ensures that our method is reasonable.

3.4 Summary

The procedure of our proposed method, as well as three different implementations, is summarized in Alg. 1. The input set contains the parameters from the dictionary learning algorithms, such as α, β, γ. The usage of the OMP algorithm ensures the sparsity of both a linear solution and dictionary learning. The fusion factor λ is the only new parameter introduced by our method, which can be learned in a pre-evaluation process, to balance the contributions from the two models. The test sample y will be coded using a new dictionary, with the help of a constructed dictionary of SR and a learned dictionary of LCLE-DL. The minimal distance solved in the last step decides the target class defining the test sample. With the dual sparse representation using virtual samples, the recognition accuracy is likely to be enhanced.

Algorithm 1: Dual sparse learning using virtual samples in dictionary learning.

```
Input: the original training samples $A$, a test sample $y$, $\varepsilon$, $\alpha$,
        $\beta$, $\gamma$, $i$ (iterations), $ii$ (initial iterations), $th$ (sparsity
        threshold), $knn$, atoms, and a fusion factor $\lambda$.
Output: a class nearest to the test sample.
Step 1: Mirroring the original samples to generate the virtual
  samples $A'$.
Step 2: Normalizing the columns of both $A$ and $A'$ to have
  unit $l_2$-norm.
Step 3: for $i = 1, 2, \cdots, C$ do
    Solving sparse solution $X$ via OMP [12].
    (According to Eq. 1.)
    Computing the residual between $y_i$ and the $i^{th}$ class.
    (By using Eq. 2.)
end
Step 4: Performing LCLE-DL, to obtain a dictionary $D$, as
  well as a coefficient $X$ and $V$. (In Eq. 12, 14 and 16.)
Step 5: for $i = 1, 2, \cdots, C$ do
    Computing the residual between $y_i$ and the $i^{th}$ class.
    (By using Eq. 17.)
end
Step 6: Combing two residuals by using a fusion factor $\lambda$. (In
  Eq. 18.)
Step 7: Classifying or identifying the test sample $y$ into a
  class with minimal distance (By Eq. 19).
```

4 Experimental results

We conducted a series of experiments on four facial databases to evaluate the performance of the proposed algorithms, including vSR-DL, SR-vDL and v-SR-DL. First, recognition experiments based on the face images are conducted on CMU Faces [31] and MUCT [30] face databases. Previous studies have already proved that the accuracy of face recognition can be effectively raised by expanding the training set. However, our work mainly concentrates on proving the feasibility of dual sparse models by adding synthetic samples into the original training set. Although there are many sophisticated data augmentation techniques [4], like pair samples [21], the work in this paper mainly focuses on how to improve the sparse learning via virtual samples. Hence, one of the simplest data augmentation is utilized to generate the virtual samples. Horizontally flipping the face images is an easy implementation and popular in face recognition. This is the reason why we apply it in the experiments. We evaluated the accuracy produced by using a different number of training samples per class, to confirm the performance of our proposed algorithms.

Second, we tested the effects of using different number of atoms in dictionary learning. Previous works have shown that using larger number of atoms help produce higher accuracy. However, we prove that the proposed method shows a different trend on both Georgia Tech (GT) [10], and LFW [19] face databases. By fusing two sets of sparse features, even using a small number of atoms can produce a high recognition accuracy. The detailed experiments results are demonstrated as follows. Besides of the aforementioned smaller datasets, we also utilized another large-scale face dataset YouTubeFace [45] to evaluate the proposed methods, so as to confirm the performance in large-scale tasks.

In this way, we evaluated our method on variant datasets, so as to simulate the complicated conditions of the real-world applications. In particular, some of them were collected under uncertain illumination, while others were captured from video clips (a.k.a. YouTubeFace) with random backgrounds. Only the classifiers with adequate robustness can cope with the tough real-world conditions. The robustness of our method can therefore be proven if it is able to produce very good classification results. The detailed settings of all these datasets are described as follows.

4.1 Configuration and preprocessing

On the CMU Faces database, all 640 black and white face images were taken with varying pose (straight, left, right, up), expression (neutral, happy, sad, angry), eyes (wearing sunglasses or not), and size [31]. This is a very small dataset, and it is useful to evaluate the performance of the method on small datasets. We shrunk the images to a size of 30 × 32 pixels to speed up the experiments. No preprocessing or feature extraction was done on the samples.

The MUCT face database [30] contains 3755 faces with 76 manual landmarks. First of all, we cropped the images based on the landmarks to locate the faces. There are 276 objects in total in the dataset. The samples were captured from five cameras, but the samples are imbalanced. Some of them have only two images from each camera, while the others have three samples. We conducted different experiments to cover the balanced and imbalanced cases, to benchmark the recognition and compare with current state-of-the-art methods. Furthermore, we resized all images to 32 × 24 pixels and used only the gray-scale data of the samples.

Similar to MUCT, the Georgia Tech (GT) face database [10] is also a colorful face dataset, where we converted the RGB images to gray scale in order to run our tests. There are only 750 face images from 50 individuals. The original face images were all at the resolution of 640 × 480 pixels, so we resized them to 30 × 40 pixels to reduce the computing complexity.

LFW [18, 24] consists of more than 13,000 images. According to [44], we selected a subset of the LFW face database consisting of 1215 images of 86 persons, and around 11–20 images for each person. All the images had already been converted into gray images, and also manually cropped and resized to 32 × 32 pixels. When extracting deep features via FaceNet [37], we picked only 10 classes with at least ten samples, being consistent with the implementation of FaceNet.

The last database YouTubeFace [45] is a large-scale database of face videos. It is useful and designed for unconstrained face recognition in videos. There are 3425 clips of 1595 different subjects. In the experiment, we chose 1830 classes with at least 100 samples, and randomly selected 100 samples per subject. Therefore, the experiment was run on 183,000 total samples. Examples of images from the above five face databases are displayed in Fig. 2.

4.2 Recognition accuracy

The experiments were conducted with three specific goals. The first goal was to test the recognition accuracy of the proposed algorithms compared to other state-of-the-art algorithms. The second one was to measure the role of virtual samples in improving the sparse representation, and the last one was to evaluate the results of using different atom numbers. The parameters of the four benchmark databases are listed in Table 1. All of them are obtained through cross-validation experiments. Figures 3, 4, 5, 6 and 7 demonstrate the detailed results obtained on the four benchmark databases.

Compared with state-of-the-art methods like SRC [46] (using OMP [12]), CRC [64], ProCRC [6], INNC [51], DSRL2 [50], SCRC [55], K-SVD [1], LC-KSVD [22], LCLE-DL [25], KNN and PCA, the proposed algorithms (in bold curves with markings) produced a higher accuracy in most cases. In the experiments, we iteratively treat the first n samples in each class as the training samples, and use the remaining samples as the test samples. First of all, the proposed algorithms obtain a much higher recognition

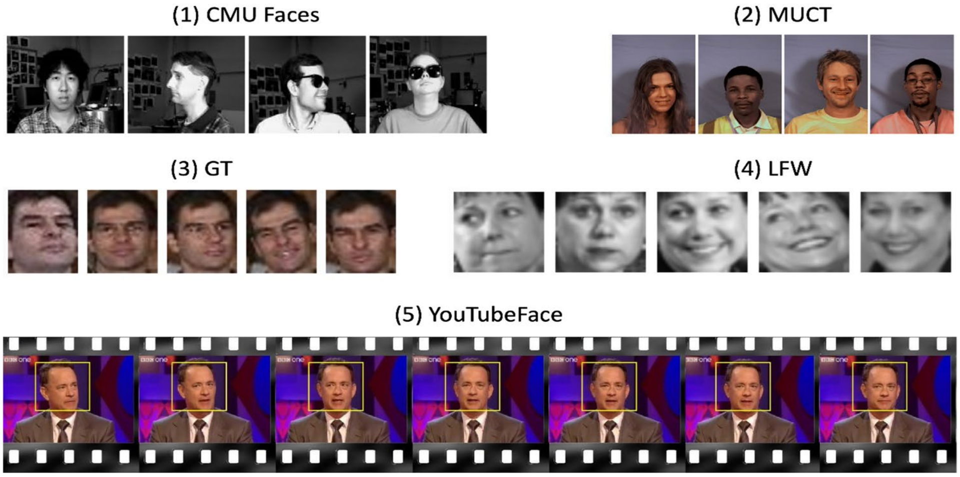

Fig. 2 Image samples from the four face databases

Table 1 The optimal parameters for the four benchmark databases

Database	Implementations	Fusion factor λ	Dictionary learning: $th, i, ii, knn, \alpha, \beta, \gamma$
CMUFaces	vSR-DL	−0.5	40, 10, 1, 1, 0.01, 0.01, 0.01
CMUFaces	SR-vDL	−5	40, 10, 1, 1, 0.01, 0.1, 0.01
CMUFaces	v-SR-DL	−0.5	30, 10, 1, 1, 0.01, 0.01, 0.1
MUCT	vSR-DL	−0.5	30, 10, 1, 1, 0, 0.1
MUCT	SR-vDL	−0.5	40, 10, 1, 1, 0.1, 0.1, 0.1
MUCT	v-SR-DL	−0.5	30, 10, 1, 1, 0.1, 0.01, 0.1
GT	vSR-DL	−0.5	30, 10, 1, 1, 0.01, 0.01, 0.1
GT	SR-vDL	−5	40, 15, 1, 1, 0.01, 0.01, 0.01
GT	v-SR-DL	−0.5	40, 10, 2, 1, 0.1, 0.01, 0.1
LFW	vSR-DL	−5	40, 10, 1, 1, 0.01, 0.1, 0.01
LFW	SR-vDL	−5	40, 15, 1, 1, 0.01, 0.01, 0.01
LFW	v-SR-DL	−5	30, 15, 1, 3, 0.01, 0.1, 0.1
YouTubeFace	vSR-DL	−5	40, 10, 1, 1, 0.1, 0, 0.01
YouTubeFace	SR-vDL	−5	40, 15, 1, 1, 0, 0.01, 0
YouTubeFace	v-SR-DL	−5	30, 15, 1, 1, 0, 0.01, 0

accuracy than classic KNN and PCA in all cases. For most of the sparse methods (in dotted lines), including SRC, CRC, SCRC, DSRL2 and INNC, the proposed algorithms also performed better in most cases on the four databases. We evaluated the performance on both imbalanced and balanced using the MUCT dataset, as shown in Figs. 4 and 5. In the imbalanced case, some subjects have only 10 samples in total, while others have 15 samples. We used the first 1–8 samples from each class for training. In the balanced case, only 199 subjects with 15 samples are picked and used in the experiments. The training samples contain 1–10 images from each subject. The results demonstrate that the proposed methods are better than all other methods. On the GT and LFW databases, both CRC and ProCRC methods are inferior as well when using less than 6 training samples, as shown in Fig. 7a. Therefore, we can see that the dual sparse learning method produced a better performance for face recognition.

From Figs. 3, 4, 5, 6 and 7 (in both subplots), we can also see the comparison among our three implementations. It is noted that the parameters chosen in the three algorithms are different, given alongside of the captions. Using virtual samples in both sparse models is not a good choice according to the results. On the one hand, v-SR-DL is not as good as vSR-DL or SR-vDL in most cases for the four databases. On the other hand, using more training samples, when adding the virtual samples, increases the training time. However, the performance of vSR-DL and SR-vDL, which used the virtual samples in one of the sparse models, is well-matched. For instance, on the GT face database, SR-vDL (the blue line) outperformed vSR-DL (the green line) when using 5–7 training samples, while in other cases vSR-DL had a higher recognition accuracy. However, using a smaller training set does help reduce the training time. Thus, it is better to use synthetic samples in one sparse model when performing the dual sparse learning.

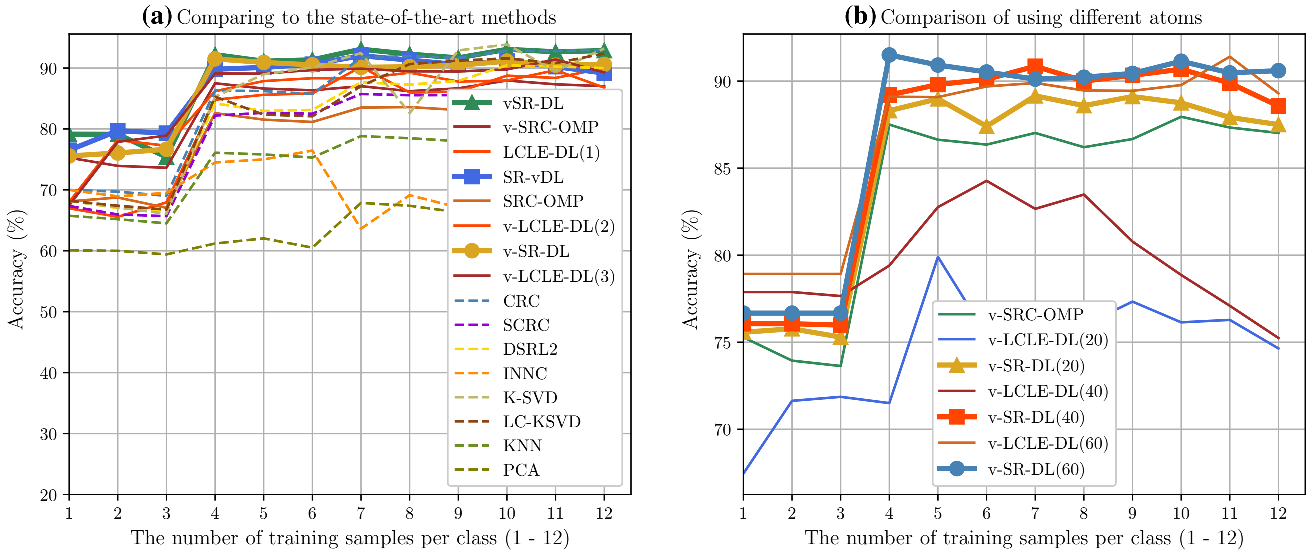

Fig. 3 Recognition accuracy obtained on the CMU Faces database

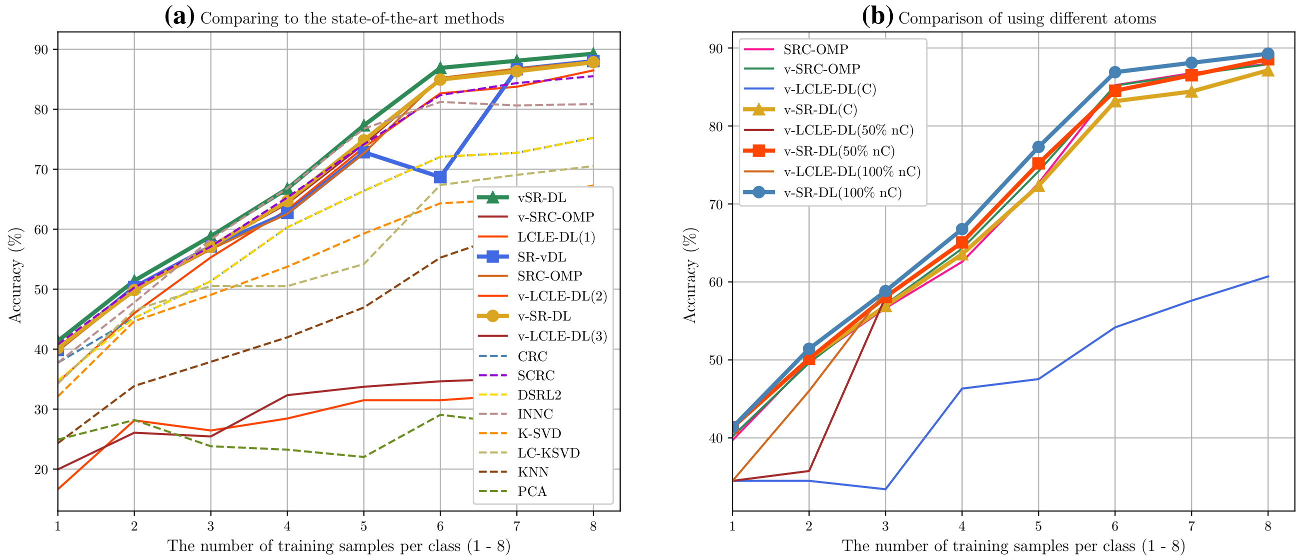

Fig. 4 Recognition accuracy of different methods on the imbalanced MUCT face database

The last measurement we want to determine is how the number of atoms affects the proposed dual sparse learning. In the second subplots in Figs. 3, 4, 5, 6 and 7, we demonstrated the results of v-SR-DL when using different atom numbers in dictionary learning. We obtained two completely opposite trends. On the CMU Faces and MUCT databases, we found out that using a larger number of atoms produces a higher recognition, as shown in Figs. 3b, 4 and 5b (n is the number of training samples, and C is the number of classes). However, the accuracy obtained by using fewer atoms is higher on the GT and LFW databases, as shown in Figs. 6b and 7b. The fewer atoms are utilized in dictionary learning, the faster the learning process performs. Therefore, it should be a sensible choice to configure a small number of atoms when encountering data similar with the GT and LFW databases.

Besides these state-of-the-art sparse representation-based methods, we evaluated some other methods that targeted the MUCT dataset, including sparse representation and Illumination Dictionary (SR-Ill-DL) [8], multi-soft biometric identification (MSBI) [2] and lattice-based feature fusion (LFF) [28]. All of them reported very promising recognition results on the MUCT datasets, but its configuration and preprocessing may be different from our proposed method. For

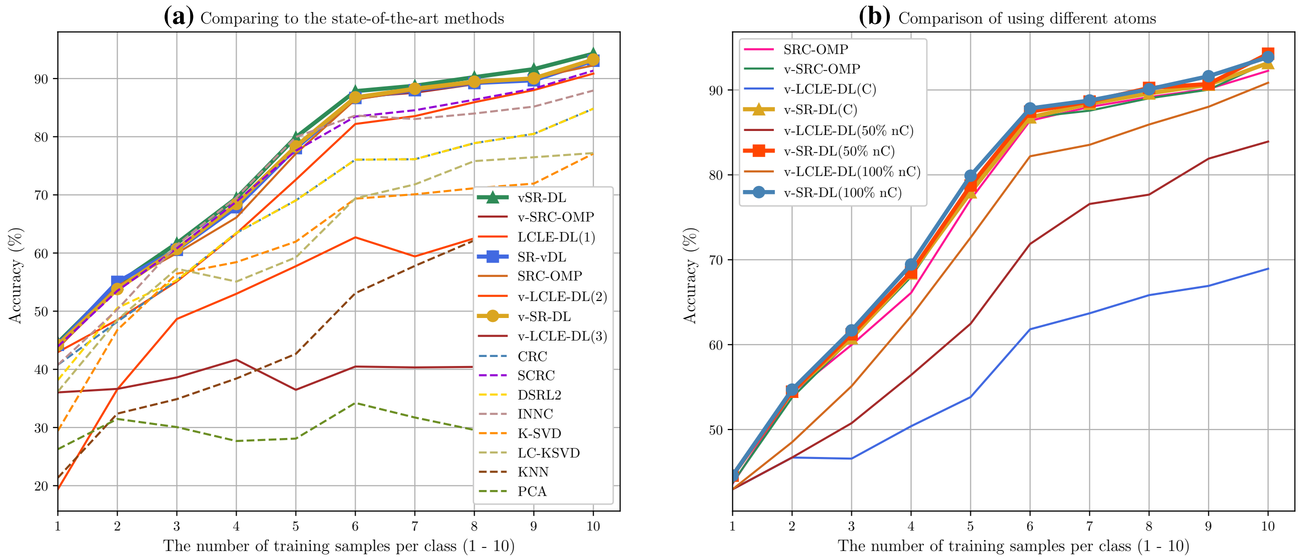

Fig. 5 Recognition accuracy by different methods on the balanced MUCT face database

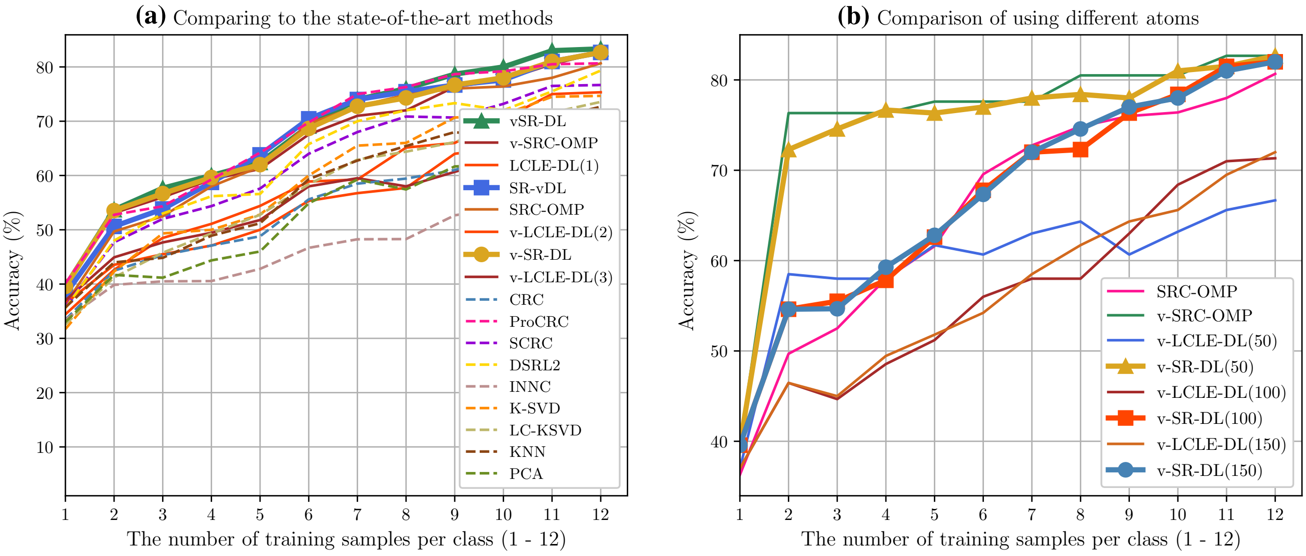

Fig. 6 Recognition accuracy of different methods on the GT face database

example, SR-Ill-DL used only 103 classes from the frontal images in the dataset [8], each of which contained 3 samples. The best results were obtained when using two samples from each class for training. The highest accuracy of SR-Ill-DL on MUCT was 84.95%. MSBI picked 199 subjects with three images from all 276 subjects, and utilized the dataset as a balanced dataset. The best result was 95.50%, which was produced by randomly selecting 50% of the samples as the training set. LFF obtained an accuracy of 94.23% using all the samples in the datasets. It is noted that all of these three methods require feature extraction before classification. Our proposed method not only supports recognition directly on the images, but also on the image features, e.g., features from deep CNN models.

Table 4 shows the experimental configuration, as well as the recognition results. Two sets of experiments were conducted to compare with current state-of-the-art methods on MUCT. The first one treated MUCT as a balanced dataset, using only 199 subjects with three images. The training set contains two images from each class, and the remaining images are used as the test samples. The best results are obtained when appending virtual samples to the training

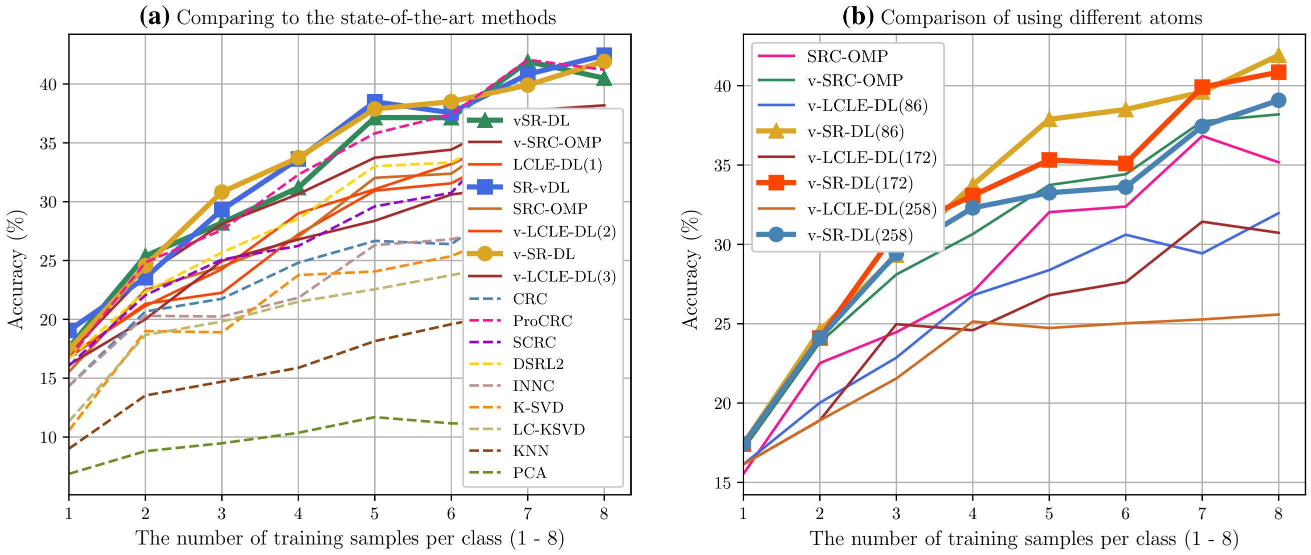

Fig. 7 Recognition accuracy obtained on the LFW face database

set in sparse representation (v-SR-DL and vSR-DL). The recognition accuracy on images can go up to 94.27%, while it reaches 96.98% on deep features extracted by FaceNet [37], higher than all current state-of-the-art methods. The second set of experiments were run on all images from all 276 subjects. Since the minimal number of samples in each class from all five cameras is $2 \times 5 = 10$, we pick eight samples from each class to perform the training process. Therefore, the test set is about 41.2% ($= 1 - 8 \times 276/3755$) of the whole samples, and the accuracy ranges from 88.43 to 95.28%. The code and data for this set of experiments have been released on our public GitHub repository (https://github.com/zengsn/muct-cropped).

4.3 Improvement analysis

In order to visually observe the improvements by the proposed methods, Table 2 shows the most promising improvements obtained by the proposed methods (the three columns before the last one, and the bold values are the best), with respect to the conventional SRC method (the second column). First of all, the improvements are stable on all four databases. The lowest improvement occurs on the largest MUCT database, where the average improvement is only 1.06%. On the small GT database, one false positive case occurs (when using 6 training samples), but the improvements are slightly higher (8.12%). As for the well aligned LFW database, the improvement is the most promising, which reaches up to 17.46%. Consequently, learning dual sparse features by using synthetic samples is a feasible way to improve sparse representation and perform robust face recognition.

The usage of the virtual samples had dual implications. On the one hand, virtual samples are solely used to expand the training set in most cases, but not always improve the sparse representation. There are several false positive cases on the MUCT and GT databases, where using augmentation does not perform well (marked with ↓). However, even in these cases, performing dual sparse learning by fusing two models produces true positive improvements. Thus, learning dual sparse features is necessary to stabilize the representation.

On the other hand, the best position to use the virtual samples is uncertain, but using it in only one sparse model obtains the most promising recognition most of the time. In particular, using virtual samples in sparse presentation only (vSR-DL) produces the highest accuracy on both the CMU Faces and GT databases, while SR-vDL achieves better results on the LFW database. Only the MUCT database is most suitable for using v-SR-DL. Considering that a larger training set needs more training time, it would be a better choice to learn only one sparse model from the virtual samples. Furthermore, when using the virtual samples in only one model, the recognition will utilize the diversity between virtual and original samples, which has proved to be helpful to the final result.

4.4 State-of-the-art recognition using image features

Our goal is to design an algorithm that is capable of robust face image recognition tasks via sparse representation and dictionary learning. So far, no feature extraction was applied in our experiments before performing recognition. Deep learning methods, like FaceNet [37] and ResNet [16], have

Table 2 Highest accuracy and improvements on all benchmark databases (%)

DB (Train)	SRC	v-SRC	Atoms	DL	v-DL	vSR-DL	SR-vDL	v-SR-DL	Improvement (%)
CMU Faces (10)	83.5	88.0	60	88.8	89.8	**93.1**	91.1	91.1	3.58
MUCT-276 (8)	88.0	87.9↓	2208	86.5	33.2↓	**89.3**	88.0	87.8	1.06
MUCT-199 (10)	92.3	93.3	1990	90.9	55.5↓	**94.3**	93.1	93.3	1.53
GT (12)	80.7	82.7	50	75.3	68.0↓	**83.3**	82.7	82.7	8.12
LFW (8)	35.2	38.2	86	31.8	36.6	40.5	**42.5**	41.9	17.46
YouTubeFace (80%)	85.9	87.0	3849	81.2	67.4	**88.6**	86.0	88.0	3.13

Table 3 Recognition results (%) on deep features

Data	Train	vSR-DL			SR-vDL			v-SR-DL		
		Images	FaceNet	ResNet	Images	FaceNet	ResNet	Images	FaceNet	ResNet
CMUFaces	12	**92.86**	67.02	90.24	89.17	70.83	89.64	90.60	67.86	89.64
GT	12	83.33	94.00	84.00	82.67	**95.33**	86.67	82.67	94.67	86.00
LFW	8	40.50	**99.74**	28.60	42.45	99.67	27.35	41.92	99.71	28.24
MUCT	8	88.56	95.02	–	81.00	74.01	–	88.43	**95.28**	–
MUCT	10	94.27	**96.98**	–	93.27	46.14	–	93.87	**96.98**	–
YouTubeFace	50%	88.60	**98.06**	–	85.97	96.41	–	88.00	97.22	–

Bold values are the best

produced a very high accuracy in face recognition tasks. However, they are hardly suited for such small datasets. In order to perform recognition with state-of-the-art results, deep features extracted by pre-trained deep neural networks can be fed into the fine-tuned classifiers. In our experiments, deep features are extracted from the pre-trained FaceNet[1] and ResNet[2] models on the same datasets, before loaded into our proposed methods. The code loads the pre-trained models and recovers the hyper-parameters to extract the deep features from the images. The recognition results are listed in Table 3.

The column *Images* shows the recognition results obtained directly from the images. By comparison with the results from deep features, we can see that recognition using deep features was very good. The proposed methods achieved a promising result on the original images when using the deep features. As shown in Table 3, the proposed methods generated recognition results from 99.67 to 99.71% for LFW on deep features from FaceNet, which were slightly higher than the result (99.2%) by the locally trained FaceNet model [37]. On the MUCT dataset, the results were promising and outperforms current state-of-the-art methods, as described in Table 4 of Sect. 4.2. This confirmed that the proposed methods are workable on deep features. On the YouTubeFace dataset, the designers [45] had already provided a set of benchmark results, as shown in Table 5. We have obtained better results, up to 88.6% using images and 98.1% using deep features.

4.5 Recognition speed

Fusing multiple models needs to perform most of the computation belonging to all models, which consumes more time. For this reason, we also evaluate the recognition speed of the proposed method. The experiments were conducted on two multi-cores servers. One is configured with Intel(R) Xeon(R) CPU E5-2660 @ 2.20GHz (8 Cores 16 Threads), and the other is with Intel(R) Xeon(R) CPU E5-1620 v2 @ 3.70GHz (4 Cores 8 Threads). The code was run with MATLAB R2017a. We recorded the total time of the recognition for all test samples (6912 in total) on the LFW database, using 1–8 training samples per class iteratively and 86 atoms in the dictionary learning. It is noted that all experiments were run with one single thread without any parallel programming. And the recognition speeds were calculated according to the total test samples and elapsed time, as shown in Table 6.

Due to the fact that dictionary learning methods do not use an iterative solver like SRC, the computation time of dictionary learning used to be significantly less than the time of sparse representation. In our experiments, SRC was implemented using parallel programming, while LELC-DL was runing on single thread. Therefore, they

[1] The code to extract deep features using FaceNet via TensorFlow - https://github.com/zengsn/facenet.

[2] The code to extract deep features using ResNet via TensorFlow - https://github.com/zengsn/TF_FeatureExtraction.

Table 4 Comparing with state-of-the-art methods for the MUCT database

Method	Classes	Train (%)	Test (%)	Accuracy (%)	Parameters: $\lambda, th, i, ii, knn, \alpha, \beta, \gamma$
SR-Ill-DLL [8]	103	67	33	84.95	Only the frontal images
MSBI [2]	199	50	50	95.50	Pyramid level L = 3, H = 8, features = 680
LFF [28]	276	60	40	94.23	Fusion of 96 LICA features
v-SR-DL (Image)	199	67	33	93.87	– 0.5, 40, 15, 2, 1, 0.01, 0.1, 0.1, images
v-SR-DL (FaceNet)	199	67	33	**96.98**	– 0.5, 30, 10, 1, 3, 0, 0.1, 0, features = 128
vSR-DL (Image)	199	67	33	**94.27**	– 0.5, 40, 15, 2, 1, 0.01, 0.01, 0.1, images
vSR-DL (FaceNet)	199	67	33	**96.98**	– 0.5, 30, 10, 1, 1, 0.1, 0.1, 0.1, features = 128

Bold values are the best

Table 5 Comparing with state-of-the-art methods for the YouTubeFace database

Method	Classes	Train (%)	Test (%)	Accuracy (%)	Parameters: $\lambda, th, i, ii, knn, \alpha, \beta, \gamma$
CSLBP in [45]	1595	50	50	78.9	Center-Symmetric LBP (CSLBP)
FBLBP in [45]	1595	50	50	80.1	Four-Patch LBP
LBP in [45]	1595	50	50	82.6	Local Binary Patterns (LBP)
v-SR-DL (Image)	1283	50	50	88.0	– 0.5, 30, 15, 1, 1, 0,0.01, 0, images
v-SR-DL (FaceNet)	1283	50	50	97.2	– 0.5, 40, 10, 1, 1, 0.1,0.01, 0.1, features = 128
SR-vDL (Image)	1283	50	50	86.0	– 0.5, 40, 15, 1, 1, 0,0.01, 0, images
SR-vDL (FaceNet)	1283	50	50	96.4	– 0.5, 40, 10, 1, 1, 0.1,0.01, 0.1, features = 128
vSR-DL (Image)	1283	50	50	**88.6**	– 0.5, 40, 10, 1, 1, 0,0.1, 0, images
vSR-DL (FaceNet)	1283	50	50	**98.1**	– 0.5, 40, 15, 1, 1, 0.01,0.1, 0.1, features = 128

Bold values are the best

Table 6 Recognition time (seconds) and speed (seconds/image) on the LFW database

Method	SRC (OMP)		LCLE-DL		ProCRC		vSR-DL		SR-vDL		v-SR-DL	
	Time	Speed	Time	Speed	Time	Speed	Time	Speed	Time	Speed	Time	Speed
E5-2660	25.85	0.004	27.37	0.004	1022.69	0.148	91.57	0.013	62.28	0.009	121.90	0.018
E5-1620 v2	31.958	0.005	21.20	0.003	1806.005	0.261	83.08	0.012	49.68	0.007	94.02	0.014

consumed almost the same time, as shown in Table 6. The ProCRC and our proposed methods were all implemented using parallel programming as well. The dual sparse learning method did not add any additional recognition time at all. Expanding the training set with virtual samples doubled the learning time (0.007+ seconds per sample).

Although ProCRC had a very high accuracy on CMU Faces and MUCT databases (as shown in Figs. 3, 4 and 5), it needed much more time to perform recognition than the proposed method. When running in a single thread without parallel programming, ProCRC consumed over ten times the recognition time (0.148+ seconds per sample) to the proposed algorithms. Therefore, the proposed method should be a better choice when considering balancing the accuracy and speed.

5 Discussion

The proposed method consists of three main steps, creating virtual samples by mirroring, obtaining l_1-norm based sparse representation and performing dictionary learning. Also, the training set with virtual samples can be utilized in SR and/or DL. Therefore, we designed three different implementations, vSR-DL, SR-vDL and v-SR-DL. In order to analyze the fusion, we plot the sparse coefficients obtained in these methods, as shown in Figs. 8, 9 and 10. All the data was captured when using eight training samples per class on the LFW database. From the visualization of the coefficients, we have the following deductions. In the figures, each row displays the sparse coefficient (x

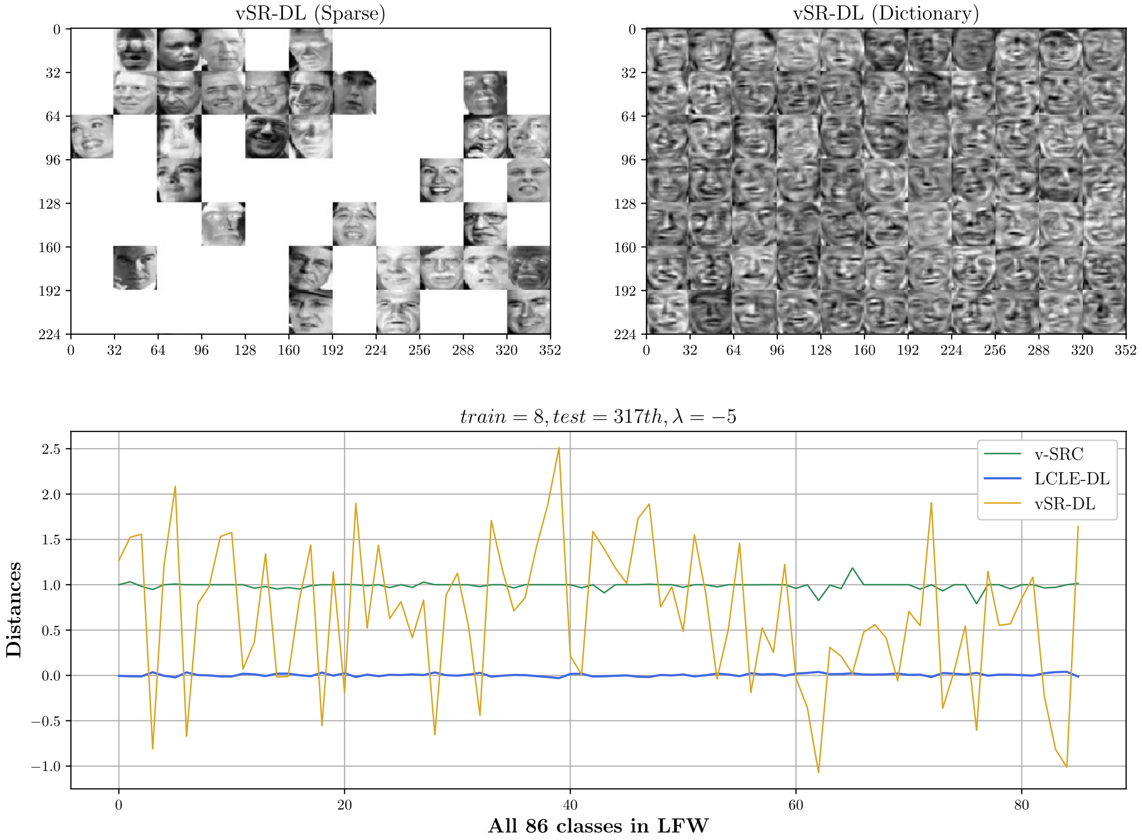

Fig. 8 Sparse coefficients, dictionary patches and distances by vSR-DL

in Eq. 1), the dictionary (D in Eq. 4) in patches and the distance for classification, corresponding to the three proposed algorithms, vSR-DL, SR-vDL and v-SR-DL, respectively. For the sparse coefficients and dictionary, we spread them on a number of 32×32 patches, to demonstrate their difference visually, as shown in the first and second columns. Two black patches in each figure have no data. The last column of figures demonstrates the distances between the test sample and all classes (here it is 86 classes in LFW), obtained by (v)SR, (v)LCLE-DL and the proposed fusion methods, vSR-DL, SR-vDL and v-SR-DL.

Coefficient sparsity plays an important role in robust sparse representation. What sparse means is that only part of the entries in the coefficient are not empty. Enhanced sparsity means there are fewer training samples needed to represent the test sample. Reducing the choices is good to obtain a higher accuracy of recognition.

Each of the first figure in Figs. 8, 9 and 10 is the visualization of the sparse coefficients obtained in the proposed methods. The coefficients are visualized to a series of tiles with the same size of image samples (32×32). The numbers of tiles without values are 36, 41, and 17, respectively. Therefore, two figures using virtual samples (by vSR-DL and v-SR-DL, in Figs. 8 and 10) have "sparser" patches than the middle one (SR-vDL), which does not use any virtual samples in sparse representation.

Feature dictionary helps to produce a discriminative recognition. Dictionary learning uses a linear combination of atoms to represent or approximate the coefficient. The task of dictionary learning is finding a dictionary to make the approximations of the training set to produce sparse coefficients as good as possible. The best result is allowing only a small number of non-zero coefficients for each approximation.

Every second figure in Figs. 8, 9 and 10 displays the features recovered from the dictionary atoms. We can see that using virtual samples in different models caused some distinction in the features. We can see that the ones by v-SR-DL (Fig. 10) have a slightly smaller amplitude of vibration than the other two, vSR-DL and SR-vDL. Though it is hard to tell which feature set is better than others, the dictionary learning models using virtual samples (SR-vDL and v-SR-DL) are slightly more discriminative than the rest (vSR-DL).

Fusing distances from SR and DL is a workable method for robust face recognition. Fusing two models on the distance level is easier and faster than directly on the feature level. By using a fusion factor, the distance from one

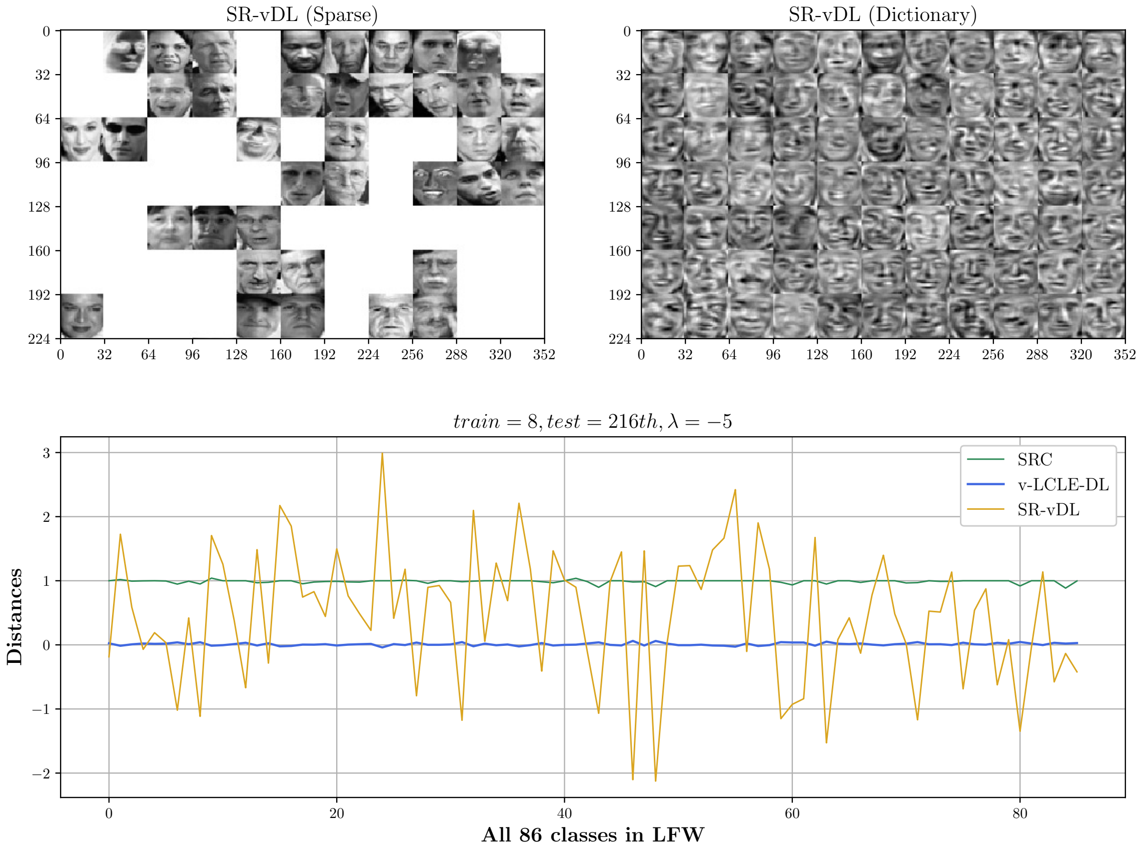

Fig. 9 Sparse coefficients, dictionary patches and distances by SR-vDL

model can be reinforced and has more influence to the final recognition.

The three curve graphs show the distances obtained by different methods, and we can see that the distance is magnified after the fusion. In all three algorithms, the fusion distances, for the 317, 216 and 114 test samples, all occupy a wider extent than two conventional distances. Expanded range of change is beneficial to filter a best class. It is noted that this kind of magnification is introduced by using the fusion factor λ. For example, all three methods on the LFW database used $\lambda = -5$. After being magnified and fused, the complementarity of the two distances are combined together to obtain a robust recognition.

6 Conclusions

Conventional l_1-norm based sparse representation and dictionary learning are two promising sparse models for face recognition. Data augmentation, even as simple as horizontal flipping, is good for the two models to generate a better sparse representation. The proposed dual sparse learning method integrates the two sparse features and utilizes the diversity from data augmentation, to implement a robust sparse model for face recognition. With three different implementations, vSR-DL, SR-vDL and v-SR-DL, the proposed method obtains a higher recognition accuracy on four face databases than conventional SRC and DL methods, as well as some other state-of-the-art sparse methods. The result demonstrates that not only the expanded size of samples by data augmentation is important, but the diversity between virtual and original samples is also beneficial for robust face recognition.

There are still some questions to be answered in this work. For example, using the virtual samples in only one model produces a better performance than in both models, but the root cause has not been clarified yet. We will work on this problem in the future. Also, we will try to determine more candidate sparse models suited for this similar multiple sparse learning. What is more, the methods currently can only handle closed-set recognition. It is interesting to explore how it performs for open-set recognition tasks in the future.

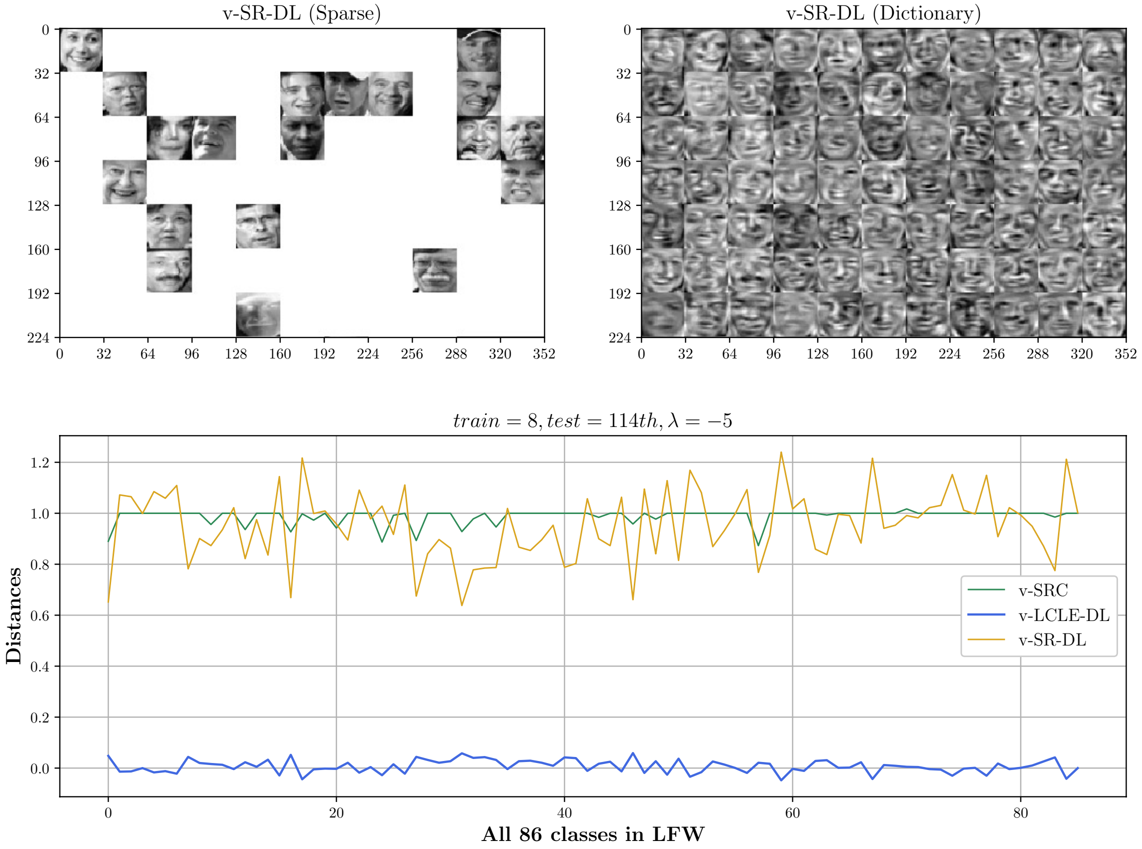

Fig. 10 Sparse coefficients, dictionary patches and distances by v-SR-DL

Acknowledgements This research was funded by the University of Macau (no. MYRG2018-00053-FST) and National Natural Science Foundation of China (NSFC) (no. 61976107), Indigenous Innovation's Capability Development Program of Huizhou University (no. hzu201907), and Science and Technology Program of Huizhou City (Grant no. 2016X0422037). This work was performed in part at the High Performance Computing Cluster (HPCC) which is supported by Information and Communication Technology Office (ICTO) of the University of Macau. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Aharon M, Elad M, Bruckstein AM (2006) K-svd: an algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans Signal Process 54(11):4311–4322
2. Arigbabu OA, Ahmad SMS, Adnan WAW, Yussof S (2015) Integration of multiple soft biometrics for human identification. Pattern Recognit Lett 68:278–287
3. Biggio B, Melis M, Fumera G, Roli F (2015) Sparse support faces. In: 2015 international conference on biometrics (ICB), IEEE, pp 208–213
4. Bloice MD, Stocker C, Holzinger A (2017) Augmentor: an image augmentation library for machine learning. arXiv preprint arXiv:1708.04680
5. Boult T, Cruz S, Dhamija A, Gunther M, Henrydoss J, Scheirer W (2019) Learning and the unknown: Surveying steps toward open world recognition. In: Proceedings of the AAAI conference on artificial intelligence, vol. 33, pp 9801–9807
6. Cai S, Zhang L, Zuo W, Feng X (2016) A probabilistic collaborative representation based approach for pattern classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2950–2959
7. Candemir S, Borovikov E, Santosh K, Antani S, Thoma G (2015) Rsilc: rotation-and scale-invariant, line-based color-aware descriptor. Image Vis Comput 42:1–12
8. Cao F, Hu H, Lu J, Zhao J, Zhou Z, Wu J (2016) Pose and illumination variable face recognition via sparse representation and illumination dictionary. Knowl Based Syst 107:117–128
9. Chen B, Li J, Ma B, Wei G (2018) Discriminative dictionary pair learning based on differentiable support vector function for visual recognition. Neurocomputing 272:306–313
10. Chen L, Man H, Nefian AV (2005) Face recognition based on multi-class mapping of fisher scores. Pattern Recognit 38(6):799–811

11. Cho S, Cha K (1996) Evolution of neural network training set through addition of virtual samples. In: Proceedings of IEEE international conference on evolutionary computation, IEEE, pp 685–688
12. Donoho DL, Tsaig Y, Drori I, Starck JL (2012) Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. IEEE Trans Inf Theory 58(2):1094–1121
13. Du Y, Wang Y (2016) Generating virtual training samples for sparse representation of face images and face recognition. J Mod Opt 63(6):536–544
14. Fawzi A, Samulowitz H, Turaga D, Frossard P (2016) Adaptive data augmentation for image classification. In: IEEE international conference on image processing (ICIP), pp 3688–3692
15. Han B, He B, Sun T, Yan T, Ma M, Shen Y, Lendasse A (2016) Hsr: L1/2-regularized sparse representation for fast face recognition using hierarchical feature selection. Neural Comput Appl 27(2):305–320
16. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
17. He Z, Patel V (2017) Sparse representation-based open set recognition. IEEE Trans Pattern Anal Mach Intell 39(8):1690–1696
18. Huang GB, Learned-Miller E (2014) Labeled faces in the wild: updates and new reporting procedures. University of Massachusetts, Amherst, Technical Report UM-CS-2014-003, pp 1–5
19. Huang GB, Ramesh M, Berg T, Learned-Miller E (2007) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Technical Report 07–49, University of Massachusetts, Amherst
20. Hussain MF, Wang H, Santosh K (2018) Gray level face recognition using spatial features. In: International conference on recent trends in image processing and pattern recognition, Springer, pp 216–229
21. Inoue H (2018) Data augmentation by pairing samples for images classification. arXiv preprint arXiv:1801.02929
22. Jiang Z, Lin Z, Davis LS (2013) Label consistent k-svd: learning a discriminative dictionary for recognition. IEEE Trans Pattern Anal Mach Intell 35(11):2651–2664
23. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ (eds) Advances in neural information processing systems, vol 25. Curran Associates Inc, Lake Tahoe, pp 1097–1105
24. Learned-Miller E, Huang G, RoyChowdhury A, Li H, Hua G (2016) Labeled faces in the wild: a survey. In: Advances in face detection and facial image analysis. Springer, pp 189–248
25. Li Z, Lai Z, Xu Y, Yang J, Zhang D (2015) A locality-constrained and label embedding dictionary learning algorithm for image classification. IEEE Trans Neural Netw Learn Syst 28(2):278–293
26. Lu Z, Zhang L (2016) Face recognition algorithm based on discriminative dictionary learning and sparse representation. Neurocomputing 174:749–755
27. Lv JJ, Shao XH, Huang JS, Zhou XD, Zhou X (2017) Data augmentation for face recognition. Neurocomputing 230:184–196
28. Marqués I, Graña M (2013) Fusion of lattice independent and linear features improving face identification. Neurocomputing 114:80–85
29. Matiz S, Barner KE (2016) Label consistent recursive least squares dictionary learning for image classification. In: IEEE international conference on image processing (ICIP), pp 1888–1892
30. Milborrow S, Morkel J, Nicolls F (2010) The MUCT landmarked face database. http://www.milbo.org/muct. Accessed 24 Jan 2020
31. Mitchell T (1999) Cmu face images. https://archive.ics.uci.edu/ml/machine-learning-databases/faces-mld/faces.html. Accessed 7 May 2017
32. Ou W, You X, Tao D, Zhang P, Tang Y, Zhu Z (2014) Robust face recognition via occlusion dictionary learning. Pattern Recognit 47(4):1559–1572
33. Patel VM, Wu T, Biswas S, Phillips PJ, Chellappa R (2012) Dictionary-based face recognition under variable lighting and pose. IEEE Trans Inf Forensics Secur 7(3):954–965
34. Quan Y, Xu Y, Sun Y, Huang Y (2016) Supervised dictionary learning with multiple classifier integration. Pattern Recognit 55:247–260
35. Rubinstein R, Peleg T, Elad M (2013) Analysis k-svd: a dictionary-learning algorithm for the analysis sparse model. IEEE Trans Signal Process 61(3):661–677
36. Rubinstein R, Zibulevsky M, Elad M (2008) Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit. Cs Technion Report CS-2008-08, pp 1–14
37. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 815–823
38. Shu T, Zhang B, Tang, YY (2018) Sparse supervised representation-based classifier for uncontrolled and imbalanced classification. IEEE transactions on neural networks and learning systems 1 (Early Access), pp 1–10
39. Skretting K, Engan K (2010) Recursive least squares dictionary learning algorithm. IEEE Trans Signal Process 58(4):2121–2130
40. Slavkovikj V, Verstockt S, De Neve W, Van Hoecke S, Van de Walle R (2015) Hyperspectral image classification with convolutional neural networks. In: Proceedings of the 23rd ACM international conference on multimedia, ACM, pp 1159–1162
41. Tang D, Zhu N, Yu F, Chen W, Tang T (2014) A novel sparse representation method based on virtual samples for face recognition. Neural Comput Appl 24(3–4):513–519
42. Thian NPH, Marcel S, Bengio S (2003) Improving face authentication using virtual samples. In: 2003 IEEE international conference on acoustics, speech, and signal processing, 2003. Proceedings (ICASSP'03), IEEE, pp 3–233
43. Wang H, Hussain MF, Mukherjee H, Obaidullah SM, Hegadi RS, Roy K, Santosh K (2018) An empirical study: elm in face matching. In: International conference on recent trends in image processing and pattern recognition, Springer, pp 277–287
44. Wang SJ, Yang J, Sun MF, Peng XJ, Sun MM, Zhou CG (2012) Sparse tensor discriminant color space for face verification. IEEE Trans Neural Netw Learn Syst 23(6):876–888
45. Wolf L, Hassner T, Maoz I (2011) Face recognition in unconstrained videos with matched backgroundsimilarity. In: IEEE conference on computer vision and pattern recognition, pp 529–534
46. Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y (2009) Robust face recognition via sparse representation. IEEE Trans Pattern Anal Mach Intell 31(2):210–227
47. Xu Y, Li Z, Zhang B, Yang J, You J (2017) Sample diversity, representation effectiveness and robust dictionary learning for face recognition. Inf Sci 375(C):171–182
48. Xu Y, Zhang B, Zhong Z (2015) Multiple representations and sparse representation for image classification. Pattern Recognit Lett 68:9–14
49. Xu Y, Zhang Z, Lu G, Yang J (2016) Approximately symmetrical face images for image preprocessing in face recognition and sparse representation based classification. Pattern Recognit 54:68–82
50. Xu Y, Zhong Z, Yang J, You J, Zhang D (2016) A new discriminative sparse representation method for robust face recognition via l2 regularization. IEEE Trans Neural Netw Learn Syst PP(99):1–10

51. Xu Y, Zhu Q, Chen Y, Pan JS et al (2012) An improvement to the nearest neighbor classifier and face recognition experiments. Int J Innov Comput Inf Control 8(12):1349–4198
52. Yong X, Lu Y (2015) Adaptive weighted fusion: a novel fusion approach for image classification. Neurocomputing 168:566–574
53. Zeng S, Gou J, Deng L (2017) An antinoise sparse representation method for robust face recognition via joint l1 and l2 regularization. Expert Syst Appl 82(1):1–9
54. Zeng S, Gou J, Yang X (2018) Improving sparsity of coefficients for robust sparse and collaborative representation-based image classification. Neural Comput Appl 30(10):2965–2978
55. Zeng S, Yang X, Gou J (2017) Multiplication fusion of sparse and collaborative representation for robust face recognition. Multimed Tools Appl 76(20):20889–20907
56. Zeng S, Yang X, Gou J (2017) Using kernel sparse representation to perform coarse-to-fine recognition of face images. Optik 140:528–535
57. Zeng S, Zhang B, Du Y (2017) Joint distances by sparse representation and locality-constrained dictionary learning for robust leaf recognition. Comput Electron Agric 142:563–571
58. Zeng S, Zhang B, Lan Y, Gou J (2019) Robust collaborative representation-based classification via regularization of truncated total least squares. Neural Comput Appl 31(10):5689–5697
59. Zhang B, Ji S, Li L, Zhang S, Yang W (2016) Sparsity analysis versus sparse representation classifier. Neurocomputing 171:387–393
60. Zhang B, Karray F, Li Q, Zhang L (2012) Sparse representation classifier for microaneurysm detection and retinal blood vessel extraction. Inf Sci 200:78–90
61. Zhang B, Vijaya Kumar B, Zhang D (2014) Noninvasive diabetes mellitus detection using facial block color with a sparse representation classifier. IEEE Trans Biomed Eng 61(4):1027–1033
62. Zhang C, Zhou P, Li C, Liu L (2015) A convolutional neural network for leaves recognition using data augmentation. In: IEEE international conference on computer and information technology; ubiquitous computing and communications; dependable, autonomic and secure computing; pervasive intelligence and computing (CIT/IUCC/DASC/PICOM), Liverpool, pp 2143–2150
63. Zhang H, Wang F, Chen Y, Zhang W, Wang K, Liu J (2016) Sample pair based sparse representation classification for face recognition. Expert Syst Appl 45:352–358
64. Zhang L, Yang M, Feng X (2011) Sparse representation or collaborative representation: Which helps face recognition? In: IEEE international conference on computer vision (ICCV), Barcelona, pp 471–478
65. Zhang Q, Li B (2010) Discriminative k-svd for dictionary learning in face recognition. In: IEEE conference on computer vision and pattern recognition (CVPR), California, pp 2691–2698
66. Zhang Y, Zeng S, Zeng W, Gou J (2018) Gnn-crc: discriminative collaborative representation-based classification via gabor wavelet transformation and nearest neighbor. J Shanghai Jiaotong Univ (Sci) 23(5):657–665
67. Zhang Z, Xu Y, Yang J, Li X, Zhang D (2017) A survey of sparse representation: algorithms and applications. IEEE Access 3:490–530
68. Zhu P, Zhu W, Wang W, Zuo W, Hu Q (2017) Non-convex regularized self-representation for unsupervised feature selection. Image Vis Comput 60:22–29

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.