Deep Cascade Model based Face Recognition: When Deep-layered Learning Meets Small Data

Lei Zhang, *Senior Member, IEEE*, Ji Liu, Bob Zhang, *Member, IEEE*, David Zhang, *Fellow, IEEE*, Ce Zhu, *Fellow, IEEE*

***Abstract*—Sparse representation based classification (SRC), nuclear-norm matrix regression (NMR), and deep learning (DL) have achieved a great success in face recognition (FR). However, there still exist some intrinsic limitations among them. SRC and NMR based coding methods belong to one-step model, such that the latent discriminative information of the coding error vector cannot be fully exploited. DL, as a multi-step model, can learn powerful representation, but relies on large-scale data and computation resources for numerous parameters training with complicated back-propagation. Straightforward training of deep neural networks from scratch on small-scale data is almost infeasible. Therefore, in order to develop efficient algorithms that are specifically adapted for *small-scale* data, we propose to derive the *deep* models of SRC and NMR. Specifically, in this paper, we propose an end-to-end deep cascade model (DCM) based on SRC and NMR with hierarchical learning, nonlinear transformation and multi-layer structure for corrupted face recognition. The contributions include four aspects. First, an end-to-end deep cascade model for small-scale data without back-propagation is proposed. Second, a multi-level pyramid structure is integrated for local feature representation. Third, for introducing nonlinear transformation in layer-wise learning, softmax vector coding of the errors with class discrimination is proposed. Fourth, the existing representation methods can be easily integrated into our DCM framework. Experiments on a number of small-scale benchmark FR datasets demonstrate the superiority of the proposed model over state-of-the-art counterparts. Additionally, a perspective that deep-layered learning does not have to be *convolutional neural network* with *back-propagation* optimization is consolidated. The demo code is available in https://github.com/liuji93/DCM**

***Index Terms*—Deep cascade model, softmax vector, representation learning, face recognition, corruption.**

I. Introduction

FACE recognition has been recognized as one of the most popular and challenging topic in computer vision and pattern recognition. In the past decade, various face recognition methods have been developed by world-wide researchers. Among them, sparse coding, nuclear-norm matrix regression analysis and deep learning have yielded significant performance and become mainstream methods of an era.

Naseem *et al.* [1] proposed a linear regression classifier (LRC) and the key idea is to linearly represent a query face by using a gallery set. Further, Wright *et al.* [2] formulated a novel sparse representation based classification (SRC) framework, by imposing l_1-norm regularization term on the LRC model to avoid over-fitting. In SRC, a testing image is linearly coded by the training set with l_1-norm constraint on the coding coefficients for pursuit of sparsity. Since then, a number of methods [1], [3]–[11] have been proposed with sparse l_p-norm modeling. Zhang *et al.* [12], [13] argued that the collaborative representation scheme plays a more important role than l_1-norm based sparsity constraint, and then proposed a collaborative representation classifier (CRC) based on l_2-norm constraint. Competitive results on face recognition were also achieved without pursuit of sparsity. With Bayesian learning theory, the rationality behind l_2-norm or l_1-norm regularization follows a conditionally independent probabilistic prior assumption that the noise (coding error) obeys Gaussian or Laplacian distribution. The probabilistic prior works if the data is uncorrupted. Nevertheless, if there were some illumination variation, occlusion, or disguise, in which the corrupted region is pixel-correlated, the prior assumption becomes invalid. Therefore, a number of algorithms have been developed to model the images with corruptions and outliers [14]–[23].

Besides the sparse coding, low-rank coding is also proved to be useful for modeling intra-pixel highly-correlated corruptions. Yang *et al.* [24] proposed a two-dimensional image matrix model, *i.e.* nuclear-norm matrix regression (NMR) for corrupted face recognition. Further, Xie *et al.* [25] proposed a weighted nuclear-norm based robust matrix regression (RMR) model and the corrupted FR performance is promoted.

Recently, deep neural networks have achieved a great success in computer vision [26], [27] and face recognition [28], [29]. The reason is that deep learning allows nonlinear computational models to learn fantastically complex, subtle and abstract features by connecting multiple hierarchical layers.

Conventional representation models such as sparse representation and nuclear-norm matrix regression can capture discriminative features by efficient *one-step* prior modeling on the error (noise) for uncorrupted FR. However, the implied discriminant information in the coding error vector cannot be exploited via a one-step strategy, such that the performance is seriously degraded especially for corrupted face recognition.

This work was supported by the National Science Fund of China under Grants (61771079), Chongqing Youth Talent Program, and the Fundamental Research Funds of Chongqing (No. cstc2017zdcy-zdzxX0002). (*Corresponding author: Lei Zhang*)

Lei Zhang and Ji Liu are with the School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China (E-mail: leizhang@cqu.edu.cn, jiliu@cqu.edu.cn).

Bob Zhang is with the Department of Computer and Information Science at the University of Macau, Macau (E-mail: bobzhang@umac.mo).

David Zhang is with School of Science and Engineering, Chinese University of Hong Kong (Shenzhen), Shenzhen, China. (E-mail: csdzhang@comp.polyu.edu.hk).

Ce Zhu is with School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China. (E-mail: eczhu@uestc.edu.cn).

Additionally, although deep learning has become dominant in various applications, DL still relies on large-scale labeled data for massive parameters training, powerful computational resources for programming and expensive time cost due to the very cumbersome gradient back-propagation optimization and labeling process. Additionally, the biological nature and interpretability of DL (*i.e.* deep neural network) are still not completely clear. More importantly, straightforward training of DL from scratch on small-scale data is almost impossible. Therefore, the above issues motivate us to propose a deep framework that is specifically adapted for small-scale data. With this goal, in this paper, we target at integrating the *deep* concept of DL into the *coding* framework of conventional representation models such as SRC and NMR. Consequently, a specialized end-to-end deep cascade model (DCM) for small-scale data is developed. In DCM, we are committed to inherit the excellent traits in DL such as hierarchical learning, nonlinear feature transformation and multi-layer connections, and achieve discriminative feature abstraction with SRC and NMR as the basic operation unit in DCM. Besides, we would also like to show that deep-layered learning does not have to be convolutional neural network architecture with cumbersome back-propagation optimization, and the models with hierarchical learning, nonlinear transformation and multi-layer connection characteristics can also be called *deep-layered* methods. This will contribute to the emergence of more deep-layered algorithms for small-scale data (*e.g.* hundreds of samples). The recent deep forest method (gcForest) [30] also shows that multi-layer cascade of random forest could achieve competitive representation learning ability even without using convolutional neural network and back-propagation optimization, especially for small-scale data. The mind of deep-layered learning with end-to-end layer-wise training and without neural convolution is further consolidated in our DCM.

Specifically, as shown in Fig. 1, the proposed model includes two modules: multi-level image coding and multi-layer softmax vector coding. In multi-level image coding, three-level representation on the raw images are implemented in parallel. In softmax vector coding, a multi-layer cascade representation with layer-wise hierarchical learning is implemented. For both modules, a basic operator, i.e. *getting new feature (GNF)*, which refers to either SRC or NMR, is integrated. To the best of our knowledge, this is the first work formulating sparse coding as a deep framework trained from scratch on small-scale data. The main contributions of this paper include:

- An end-to-end deep cascade structure with hierarchical learning and multi-layer representation for high-level discriminative feature abstraction is proposed, which interprets *deep* learning as a novel hierarchical coding perspective with coding error retraining rather than conventional neural *convolution* with back-propagation. Different from conventional deep learning, the proposed deep model is designed for small-scale data.
- To explore whether the facial subregions can help improve classification performance, we utilize a three-level pyramid structure in the image coding part. In the three-level spatial pyramid structure, each image is divided into 4 and 16 subregions. For uncorrupted data, the local information can be fully explored; for corrupted data, the subregions without corruption can provide supplementary features that facilitate recognition. For discrimination, the representation errors of the whole images in the 1^{st} level together with the subregions in the 2^{nd} and 3^{rd} level are transformed into softmax vectors. To further explore the effectiveness of the softmax vectors, we design a cascade model where the softmax vectors are concatenated layer by layer in the softmax vector coding part.
- In sparse representation classifier, the testing image is categorized as the class with minimum reconstruction error. In other words, the representation error vector of all classes shows significant class discrimination. Therefore, in this paper, we propose to use the representation error vector of all classes to represent an image.
- Most of existing sparse and low-rank coding based methods can be easily integrated into the proposed multi-level image coding part. In this paper, we have integrated the SRC and NMR to obtain representation errors.

The rest of this paper is organized as follows. In Section II, we review the related work. Section III presents the proposed DCM framework. The experimental results are shown in Section IV. The discussion of the proposed DCM is presented in Section V. The analysis of algorithms is presented in Section VI. Finally, Section VII concludes this paper.

II. Related Work

In recent years, a number of representation based models have been proposed to deal with face recognition with occlusion and illumination changes. For both uncorrupted and corrupted data, a half-quadratic based method (HQ) [17] is applicable to perform both error correction and error detection. The additive function and multiplicative function based on l_1-norm sparsity constraint are defined in HQ framework, for handling corrupted and uncorrupted data, respectively. Li *et al.* explored the error structure incurred by occlusion from two aspects: the error morphology and the error distribution. They argued that the shape of the occlusion is also an important feature and therefore formulated a structural sparse error coding for face recognition with occlusion (SSEC) [20], where the error of the non-occluded part and the occluded part were measured differently. Although SSEC considered the non-occluded part and occluded part in an image, it still suppose that the occlusion exists in images.

Recently, robust regression methods based on low-rank constraint [25], [31]–[41] have been developed for face recognition problem. Typically, Luo *et al.* [42] argued that most existing one-dimensional pixel-based error models (*e.g.* SRC [2], RSC [14], RLRC [19], etc.) for dealing with face recognition problem with corruption are unreasonable for two reasons. On one hand, those one-dimensional pixel-based error models assume that pixel-wise errors are independent and identically distributed (i.i.d.). However, the pixel-wise errors are highly correlated in real-world images. On the other hand, those one-dimensional pixel-based error models use a vector to represent an image, which neglects the structure information of the

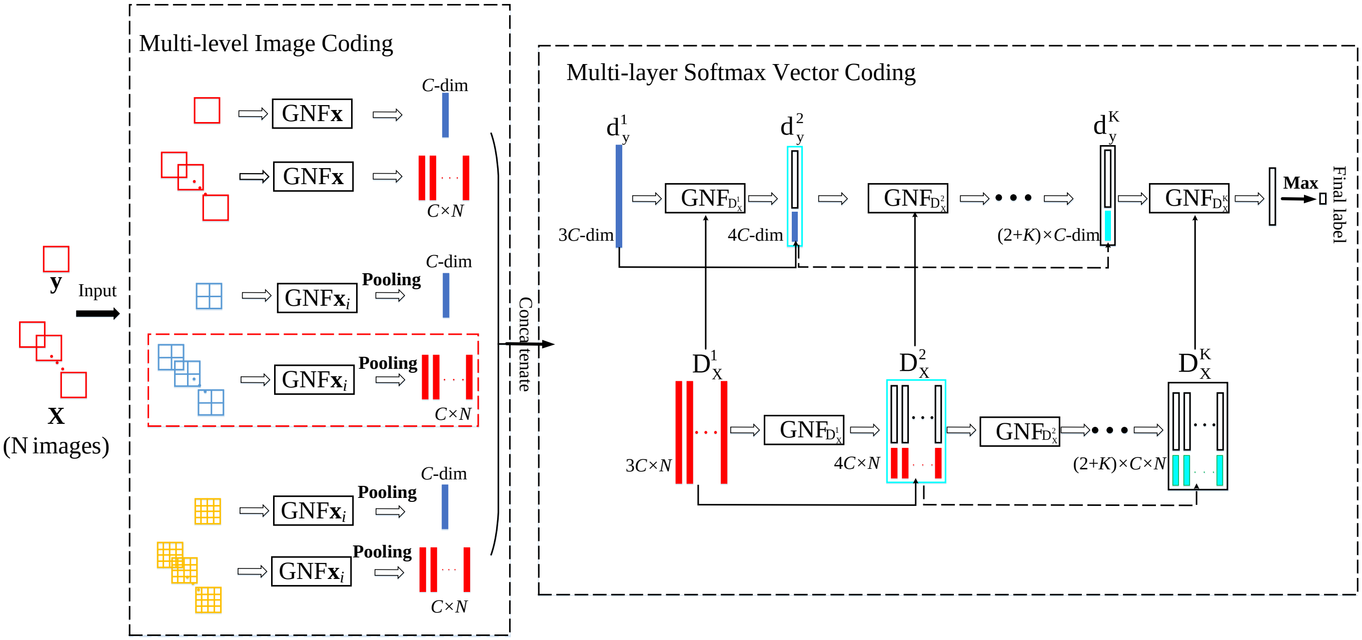


Fig. 1: The proposed DCM framework, which illustrates how the deep cascade model works when classifying a query image $\mathbf{y}$ under all training images $\mathbf{X}$. Specifically, two parts, multi-level image coding and softmax vector coding are included. Getting New Feature (GNF) is designed to compute the class-discriminative softmax vectors based on coding errors. High-level representation features, softmax vector, are computed with pyramid structure based on GNF function in the multi-level image coding part. The softmax vector coding aims to mining more discriminative features based on layer-wise representation. Note that the detailed operation in red dashed box is specially described in Fig. 3

error image (*e.g.*, the rank of error image). Based on the above motivations, Yang *et al.* proposed a two-dimensional image matrix based error model, *i.e.* nuclear norm matrix regression (NMR) [24]. The NMR model is also suitable for uncorrupted data because the elements of the residual image almost approach zeros and endows a low-rank structure.

Recently, a multi-grained cascade forest (gcForest) is proposed by Zhou *et al.* [30], in which some basic decision tree algorithms are adopted in the learning mechanism for class vectors computation. These class vectors associates with the original input are treated as the input of next level. Similarly, in this paper, we also adopt a cascade learning framework based on linear representation method for high-level discriminative feature learning with softmax function. In our previous work, a sparse softmax vector coding (SSVD) method that uses sparse coding for multi-layer feature representation was proposed in [43]. Different from the previous work, this paper introduces sparse coding and nuclear norm matrix regression models in image coding part. We can empirically choose the appropriate baseline representational learning methods in the image coding part to achieve better recognition performance.

III. Representation based Deep Cascade Model

In this section, we present the formulation of the proposed DCM framework for robust face recognition. First, we present the function, i.e., getting new feature (GNF) part. Further, for easy following the principle of DCM, we describe DCM in two parts: multi-level image coding part and multi-layer softmax vector coding part. The whole process of the proposed DCM framework is shown in Fig. 1.

A. The Basic Getting New Feature Unit: GNF

In our DCM model, we formulate a function of getting new feature (GNF), which, in the image coding part, transforms the whole image as well as its subregions into softmax vectors activated by the softmax function on the representation error. In this paper, two different representation (coding) models such as sparse representation and nuclear norm matrix regression are used in the GNF function to obtain softmax vectors. It is worth noting that more suitable coding methods that is beneficial to recognition can be freely selected and integrated into GNF in the proposed DCM model.

1) Getting New Feature based on Sparse Representation: Suppose that we have C classes of subjects, $\mathbf{d}$ represents a query sample and $\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_C]$ represents the dictionary (a group of basis). In terms of sparse representation based classifier (SRC) [2] and dictionary learning [44], the representation model can be transformed into the following minimization problem:

$$\min_{\boldsymbol{\alpha}} \|\mathbf{d} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \tag{1}$$

where λ is a scalar constant, $\|\cdot\|_2$ and $\|\cdot\|_1$ represent the l_2-norm and l_1-norm, respectively. After solving the coding coefficients $\boldsymbol{\alpha}$, the representation error of each class can be computed as follows:

$$r_c = \|\mathbf{d} - \mathbf{D}_c \boldsymbol{\alpha}_c\|_2^2 \tag{2}$$

where $\mathbf{D}_c$ is the sample set with respect to class c, and $\boldsymbol{\alpha}_c$ is the coefficient vector associated with class c. Then, by using the softmax function, the proposed softmax vector $\mathbf{S_v} \in \mathbb{R}^C$ can be computed as follows:

$$\mathbf{S_v} = [\frac{e^{-r_1}}{\sum_{c=1}^{C} e^{-r_c}}, \frac{e^{-r_2}}{\sum_{c=1}^{C} e^{-r_c}}, \cdots, \frac{e^{-r_C}}{\sum_{c=1}^{C} e^{-r_c}}]^{\mathrm{T}} \tag{3}$$

where C denotes the number of classes. It is obvious that if the testing sample $\mathbf{d}$ belongs to class i $(\leq C)$, S_v^i should be bigger than other atoms in the softmax vector $\mathbf{S_v}$, which shows class discrimination. For clarity, the above process of obtaining the softmax vector $\mathbf{S_v}$ is defined as Getting New Feature (GNF) conditioned on dictionary $\mathbf{D}$ (i.e., GNF_D). For convenience, we define the the whole procedure of computing the softmax vector $\mathbf{S_v}$ for a given query sample $\mathbf{y}$ as:

$$\mathbf{S_v} = GNF_D(\mathbf{y}, \boldsymbol{\alpha}, C) \tag{4}$$

2) Getting New Feature based on Nuclear-norm Matrix Regression: Suppose that we have C classes of subjects, $\mathbf{b}$ represents a query sample and $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_C]$ represents the dictionary. In terms of the nuclear-norm based matrix regression (NMR) [24], the representation model is formulated as the following problem:

$$\min_{\boldsymbol{\alpha}} \|\mathbf{b} - \mathbf{A}(\boldsymbol{\alpha})\|_* + \frac{\lambda}{2}\|\boldsymbol{\alpha}\|_2 \tag{5}$$

where λ is a scalar constant. $\|\cdot\|_*$ denotes the nuclear-norm computed as the summation of the singular values of a matrix, which is the convex approximation of the rank function. $\mathbf{A}(\boldsymbol{\alpha}) = \alpha_1\mathbf{A}_1 + \alpha_2\mathbf{A}_2 + \cdots + \alpha_N\mathbf{A}_N$ represents linear combination of all training images represented in matrix form. Therefore, by minimizing the nuclear-norm of the representation error, the low-rank property is guaranteed. After solving the coding coefficient $\boldsymbol{\alpha}$, the representation error of each class can be computed as follows:

$$r_c = \|\mathbf{b} - \mathbf{A}_c(\boldsymbol{\alpha}_c)\|_* \tag{6}$$

where $\mathbf{A}_c$ is the dictionary with respect to class c, and $\boldsymbol{\alpha}_c$ is the coefficient vector associated with class c. Then, the softmax vector $\mathbf{S_v}$ can be computed as follows:

$$\mathbf{S_v} = [\frac{e^{-r_1}}{\sum_{c=1}^{C} e^{-r_c}}, \frac{e^{-r_2}}{\sum_{c=1}^{C} e^{-r_c}}, \cdots, \frac{e^{-r_C}}{\sum_{c=1}^{C} e^{-r_c}}]^{\mathrm{T}} \tag{7}$$

If a query sample $\mathbf{b}$ belongs to class i $(\leq C)$, then S_v^i should be bigger than other atoms in the softmax vector $\mathbf{S_v}$, which implies class discrimination. The above process of computing the softmax vector $\mathbf{S_v}$ is defined as Getting New Feature conditioned on dictionary $\mathbf{A}$ (i.e., GNF_A). Then, for a given query sample $\mathbf{y}$, its softmax vector $\mathbf{S_v}$ can be computed as:

$$\mathbf{S_v} = GNF_A(\mathbf{y}, \boldsymbol{\alpha}, C) \tag{8}$$

B. Multi-level Image Coding

Without loss of generality, we let $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ represent the training images (gallery set), where $\mathbf{x}_n \in \mathbb{R}^{p\times q}$. We let $\mathbf{y} \in \mathbb{R}^{p\times q}$ represent the testing image (query sample), where p and q represent image size. The number of classes is C and the number of training images is N. For each image, a three-level spatial pyramid is established for representation and softmax vector generation. We take one face image in Extended Yale B database as an instance to illustrate the three-level spatial pyramid, which is shown in Fig. 2.

We let $\mathbf{H}_X^{l_i}$ represent the training set which consists of the i-th subregion in level l of all training images $\mathbf{x}_n$ $(n = 1, ..., N)$ and $\mathbf{H}_y^{l_i}$ represent the i-th subregion of the testing image in level l, where $l = [0, 1, 2]$ and $i = [1, \cdots, 4^l]$. Specially, when $l = 0$, $\mathbf{H}_X^{l_i}$ is the original training image set (i.e., $\mathbf{X}$) and $\mathbf{H}_y^{l_i}$ is the raw testing image (i.e., $\mathbf{y}$). As shown in Fig. 1, there are three parallel channels (3-level pyramid) designed to transform each input image into a softmax vector $\mathbf{S_v}$ in the image coding part. Note that, due to that multiple subregions exist in the 2^{nd} and 3^{rd} channels, we therefore introduce the max-pooling and average-pooling operations such that each image can only be represented by one softmax vector. In the following, we present the specific models and algorithms for computing the softmax vectors of the testing image $\mathbf{y}$ by using sparse representation and nuclear-norm matrix regression, respectively.

1) Sparse Representation based Multi-level Image Coding: Following SRC model, we formulate the sparse coding problem of the i-th subregion of a query in level l as:

$$\min_{\mathbf{w}_y^{l_i}} \|\mathbf{H}_y^{l_i} - \mathbf{H}_X^{l_i}\mathbf{w}_y^{l_i}\|_2^2 + \lambda\|\mathbf{w}_y^{l_i}\|_1 \tag{9}$$

where λ is the regularization parameter. In recent years, different solving algorithms have been proposed for sparse representation. In particular, the alternating direction method of multipliers (ADMM) proposed in 1970s [45] has drawn a lot of attention. Yang and Zhang [44] integrated the proximal methods into ADMM when solving l_1-norm minimization problems. In this paper, we also use ADMM method to solve the sparse representation problem.

Generally, based on ADMM, we introduce an auxiliary variable $\mathbf{z}$, there is $\mathbf{z}_y^{l_i} = \mathbf{w}_y^{l_i}$. Then, the augmented Lagrangian function of problem (9) can be formulated as

$$\begin{aligned} L_\mu(\mathbf{w}_y^{l_i}, \mathbf{z}_y^{l_i}, \boldsymbol{\Lambda}_y^{l_i}) = \min_{\mathbf{w}_y^{l_i}, \mathbf{z}_y^{l_i}, \boldsymbol{\Lambda}_y^{l_i}} \|\mathbf{H}_y^{l_i} - \mathbf{H}_X^{l_i}\mathbf{w}_y^{l_i}\|_2^2 + \\ \lambda\|\mathbf{z}_y^{l_i}\|_1 + < \boldsymbol{\Lambda}_y^{l_i}, \mathbf{w}_y^{l_i} - \mathbf{z}_y^{l_i} > + \frac{\mu}{2}\|\mathbf{w}_y^{l_i} - \mathbf{z}_y^{l_i}\|_2^2 \end{aligned} \tag{10}$$

where $< \mathbf{P}, \mathbf{Q} >= tr(\mathbf{P}^{\mathbf{T}}\mathbf{Q})$, $\boldsymbol{\Lambda}_y^{l_i}$ is a Lagrange multiplier and μ is a scalar constant. The augmented Lagrangian function is minimized alone one coordinate direction at each iteration. Specifically, ADMM consists of the following iterations.
(i) Given $\mathbf{z}_y^{l_i} = \mathbf{z}_y^{l_i(t)}, \boldsymbol{\Lambda}_y^{l_i} = \boldsymbol{\Lambda}_y^{l_i(t)}$, updating $\mathbf{w}_y^{l_i}$ by

$$\mathbf{w}_y^{l_i(t+1)} = \arg\min_{\mathbf{w}_y^{l_i}} L_\mu(\mathbf{w}_y^{l_i}, \mathbf{z}_y^{l_i}, \boldsymbol{\Lambda}_y^{l_i}) \tag{11}$$

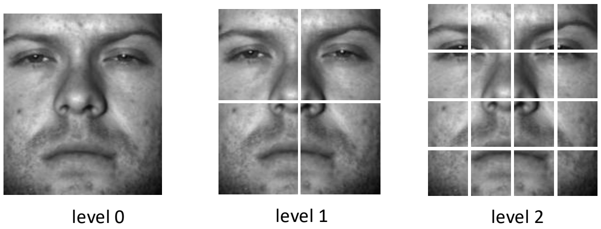


Fig. 2: Toy example of constructing a three-level pyramid. Level 0 is the original image. In level 1, the original image is equally divided into 4 subregions. In level 2, the original image is equally divided into 16 subregions.

Algorithm 1 The solving algorithm for problem (9)

```
Input: Training samples $\mathbf{H}_X^{l_i}$ and testing samples $\mathbf{H}_y^{l_i}$ with
    $l_2$-normalization, class number $C$, parameters $\lambda_1 = 10^{-4}, \mu_1 = 10^{-1}$;
Output: $\mathbf{w}_y^{l_i}$, $\mathbf{S_v}_y^{l_i}$
Initialize: $\mathbf{w}_y^{l_i(0)} = \mathbf{z}_y^{l_i(0)} = \mathbf{\Lambda}_y^{l_i(0)} = \mathbf{0}$
repeat
    Update $\mathbf{w}_y^{l_i}$: $\mathbf{w}_y^{l_i(t+1)} = (\mathbf{H}_X^{l_i\,\mathrm{T}}\mathbf{H}_X^{l_i}+\mu\mathbf{I})^{-1}(\mathbf{H}_X^{l_i\,\mathrm{T}}\mathbf{H}_y^{l_i} - \mathbf{\Lambda}_y^{l_i(t)} + \mu\mathbf{z}_y^{l_i(t)})$ using Eq.(15);
    Update $\mathbf{z}_y^{l_i}$: $\mathbf{z}_y^{l_i(t+1)} = shrinkage_{\frac{\lambda}{\mu}}(\mathbf{w}_y^{l_i(t+1)} + \frac{\mathbf{\Lambda}_y^{l_i(t)}}{\mu})$ using Eq.(17);
    Update $\mathbf{\Lambda}_y^{l_i}$: $\mathbf{\Lambda}_y^{l_i(t+1)} = \mathbf{\Lambda}_y^{l_i(t)} + \mu(\mathbf{w}_y^{l_i(t+1)} - \mathbf{z}_y^{l_i(t+1)})$
until convergence
$\mathbf{S_v}_y^{l_i} = GNF_{\mathbf{H}_X^{l_i}}(\mathbf{H}_y^{l_i}, \mathbf{w}_y^{l_i}, C)$
```

(ii) Given $\mathbf{w}_y^{l_i} = \mathbf{w}_y^{l_i(t+1)}, \mathbf{\Lambda}_y^{l_i} = \mathbf{\Lambda}_y^{l_i(t)}$, updating $\mathbf{z}_y^{l_i}$ by

$$\mathbf{z}_y^{l_i(t+1)} = arg \min_{\mathbf{z}_y^{l_i}} L_\mu(\mathbf{w}_y^{l_i}, \mathbf{z}_y^{l_i}, \mathbf{\Lambda}_y^{l_i}) \tag{12}$$

(iii) Given $\mathbf{w}_y^{l_i} = \mathbf{w}_y^{l_i(t+1)}, \mathbf{z}_y^{l_i} = \mathbf{z}_y^{l_i(t+1)}$, updating $\mathbf{\Lambda}_y^{l_i}$ by

$$\mathbf{\Lambda}_y^{l_i(t+1)} = \mathbf{\Lambda}_y^{l_i(t)} + \mu(\mathbf{w}_y^{l_i(t+1)} - \mathbf{z}_y^{l_i(t+1)}) \tag{13}$$

The key steps are to solve the optimization problems in Eqs. (11) and (12). Based on the augmented Lagrangian function in Eq. (10), Eq. (11) can be expressed as

$$\begin{aligned}\mathbf{w}_y^{l_i(t+1)} &= \arg\min_{\mathbf{w}_y^{l_i}}(\|\mathbf{H}_y^{l_i} - \mathbf{H}_X^{l_i}\mathbf{w}_y^{l_i}\|_2^2 + < \mathbf{\Lambda}_y^{l_i}, \\ &\mathbf{w}_y^{l_i} - \mathbf{z}_y^{l_i} > + \frac{\mu}{2}\|\mathbf{w}_y^{l_i} - \mathbf{z}_y^{l_i}\|_2^2)\end{aligned} \tag{14}$$

Since Eq. (14) is a standard least-square regression problem, the closed-form solution can be obtained as follows

$$\mathbf{w}_y^{l_i(t+1)} = (\mathbf{H}_X^{l_i\,\mathrm{T}}\mathbf{H}_X^{l_i} + \mu\mathbf{I})^{-1}(\mathbf{H}_X^{l_i\,\mathrm{T}}\mathbf{H}_y^{l_i} - \mathbf{\Lambda}_y^{l_i(t)} + \mu\mathbf{z}_y^{l_i(t)}) \tag{15}$$

where $\mathbf{I}$ is an identity matrix. Based on the augmented Lagrangian function in Eq. (10), Eq. (12) is rewritten as

$$\begin{aligned}\mathbf{z}_y^{l_i(t+1)} &= \arg\min_{\mathbf{z}_y^{l_i}}(\lambda\|\mathbf{z}_y^{l_i}\|_1 + < \mathbf{\Lambda}_y^{l_i}, \mathbf{w}_y^{l_i} - \mathbf{z}_y^{l_i} > \\ &+ \frac{\mu}{2}\|\mathbf{w}_y^{l_i} - \mathbf{z}_y^{l_i}\|_2^2)\end{aligned} \tag{16}$$

Because l_1-norm problem is convex but non-differentiable at zero point, the shrinkage technique [44] is used to solve this problem. Therefore, the optimal solution can be presented as

$$\mathbf{z}_y^{l_i(t+1)} = shrinkage_{\frac{\lambda}{\mu}}(\mathbf{w}_y^{l_i(t+1)} + \frac{\mathbf{\Lambda}_y^{l_i(t)}}{\mu}) \tag{17}$$

After solving the representation coefficients $\mathbf{w}_y^{l_i}$, the Getting New Feature (GNF) function can be used to obtain the multi-level softmax vectors $\mathbf{S_v}_y^{l_i} \in \mathbb{R}^{C\times 1}$.

- In the 1^{st} channel ($l = 0$), we can get one softmax vector $\mathbf{S_v}_y^{0(1)}$ which is used to replace the testing image $\mathbf{H}_y^{0_1}$.
- In the 2^{nd} channel ($l = 1$), by using Eq. (4) and or (8), we can obtain four softmax vectors $\mathbf{S_v}_y^{1_i}(i = 1, \cdots, 4^1)$, which can be transformed into one softmax vector $\mathbf{S_v}_y^1 \in \mathbb{R}^{C\times 1}$ by using max-pooling function, there is
$$\mathbf{S_v}_y^1 = \max\{\mathbf{S_v}_y^{1(1)}, \cdots, \mathbf{S_v}_y^{1(4)}\} \tag{18}$$
or average pooling function, then there is
$$\mathbf{S_v}_y^1 = \frac{1}{4}\sum_{i=1}^{4}\mathbf{S_v}_y^{1_i} \tag{19}$$
- Similarly, in the 3^{rd} channel ($l = 2$), we can obtain four softmax vectors $\mathbf{S_v}_y^{1_i}(i = 1, \cdots, 4^2)$ which can be transformed into one softmax vector $\mathbf{S_v}_y^2 \in \mathbb{R}^{C\times 1}$ by using max pooling function, there is
$$\mathbf{S_v}_y^2 = \max\{\mathbf{S_v}_y^{2(1)}, \cdots, \mathbf{S_v}_y^{2(16)}\} \tag{20}$$
or average pooling function, then there is
$$\mathbf{S_v}_y^2 = \frac{1}{16}\sum_{i=1}^{16}\mathbf{S_v}_y^{2_i} \tag{21}$$

In terms of ADMM algorithm, the objective function will converge when a certain optimality condition and stopping criteria are satisfied. In this paper, a maximum number of iterations is set instead. The detailed procedure for solving problem (9) is summarized in Algorithm 1.

Generally, with these similar operations, for each training image $\mathbf{x}_n$, we can also obtain its softmax vectors $\mathbf{S_v}_{x_n}^0$ in 1^{st} channel (level 1), $\mathbf{S_v}_{x_n}^1$ in 2^{nd} channel (level 2), and $\mathbf{S_v}_{x_n}^2$ in 3^{rd} channel (level 3). By putting the softmax vectors of all training images together in three channels, we can obtain three groups of softmax vector set $\mathbf{S_v}_X^0 \in \mathbb{R}^{C\times N}$, $\mathbf{S_v}_X^1 \in \mathbb{R}^{C\times N}$, and $\mathbf{S_v}_X^2 \in \mathbb{R}^{C\times N}$. Visually, the generation process of the softmax vectors for each level based on max/average pooling function and GNF function are illustrated in Fig. 3, which describes the generation process of softmax vectors in level 1.

2) *Nuclear-norm Matrix Regression based Multi-level Image Coding:* According to NMR [24] model, we formulate the image coding problem of a query sample $\mathbf{y}$ as follows:

$$\min_{\mathbf{f}_y^{l_i}}\|\mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{H}_y^{l_i}\|_* + \frac{\kappa_1}{2}\|\mathbf{f}_y^{l_i}\|_2^2 \tag{22}$$

where κ_1 is trade-off parameter. As presented in the NMR [24], ADMM method is used to solve this nuclear norm optimization problems as well as [46]–[48]. In the following section, we present the detailed solving algorithm.

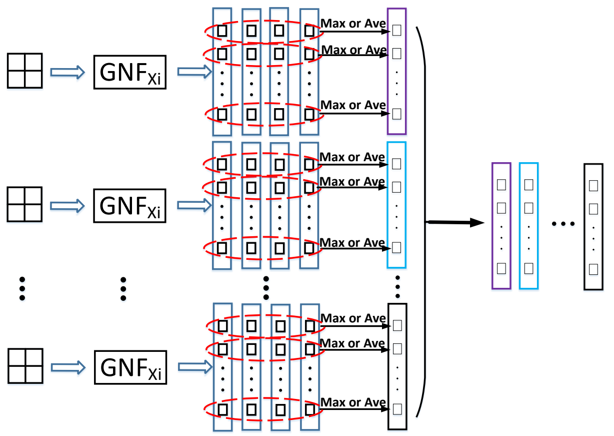


Fig. 3: The generation process of softmax vectors in level 1. X_i denotes the dictionary of the i-th subregion. For each image, a $C \times 4$ softmax matrix is generated after GNF_{X_i}. Then, by using max or average pooling operator, a $C \times 1$ softmax vector (new feature) is formulated for each image. Finally, the softmax vector for each image is grouped into a feature matrix.

In this work, we transform the Eq. (22) into a constrained optimization problem as follows.

$$\min_{\mathbf{f}_y^{l_i}} \|\mathbf{E}_y^{l_i}\|_* + \frac{\kappa_1}{2}\|\mathbf{f}_y^{l_i}\|_2^2 \quad s.t. \quad \mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{H}_y^{l_i} = \mathbf{E}_y^{l_i} \tag{23}$$

Afterwards, the augmented Lagrangian function is defined as

$$\begin{aligned} L_{\nu_1}(\mathbf{E}_y^{l_i}, \mathbf{f}_y^{l_i}, \mathbf{P}_y^{l_i}) &= \|\mathbf{E}_y^{l_i}\|_* + \frac{\kappa_1}{2}\|\mathbf{f}_y^{l_i}\|_2^2 + < \mathbf{P}_y^{l_i}, \\ \mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{E}_y^{l_i} - \mathbf{H}_y^{l_i} &> + \frac{\nu_1}{2}\|\mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{E}_y^{l_i} - \mathbf{H}_y^{l_i}\|_F^2 \end{aligned} \tag{24}$$

where $\nu_1 > 0$ is a scalar constant and $\mathbf{P}_y^{l_i}$ is the Lagrange multipliers. ADMM consists of the following iterations.
(i) Given $\mathbf{E}_y^{l_i} = {\mathbf{E}_y^{l_i}}^{(t)}$ and $\mathbf{P}_y^{l_i} = {\mathbf{P}_y^{l_i}}^{(t)}$, updating $\mathbf{f}_y^{l_i}$ by

$${\mathbf{f}_y^{l_i}}^{(t+1)} = \arg\min_{\mathbf{f}_y^{l_i}} L_{\nu_1}(\mathbf{E}_y^{l_i}, \mathbf{f}_y^{l_i}, \mathbf{P}_y^{l_i}) \tag{25}$$

(ii) Given $\mathbf{f}_y^{l_i} = {\mathbf{f}_y^{l_i}}^{(t+1)}$ and $\mathbf{P}_y^{l_i} = {\mathbf{P}_y^{l_i}}^{(t)}$, updating $\mathbf{E}_y^{l_i}$ by

$${\mathbf{E}_y^{l_i}}^{(t+1)} = \arg\min_{\mathbf{E}_y^{l_i}} L_{\nu_1}(\mathbf{E}_y^{l_i}, \mathbf{f}_y^{l_i}, \mathbf{P}_y^{l_i}) \tag{26}$$

(iii) Given $\mathbf{f}_y^{l_i} = {\mathbf{f}_y^{l_i}}^{(t+1)}$ and $\mathbf{E}_y^{l_i} = {\mathbf{E}_y^{l_i}}^{(t+1)}$, updating $\mathbf{P}_y^{l_i}$ by

$${\mathbf{P}_y^{l_i}}^{(t+1)} = {\mathbf{P}_y^{l_i}}^{(t)} + \nu_1(\mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{E}_y^{l_i} - \mathbf{H}_y^{l_i}) \tag{27}$$

In the following, the update equations for $\mathbf{f}_y^{l_i}$ and $\mathbf{E}_y^{l_i}$ are described. Similar to the sparse representation, the coding coefficient $\mathbf{f}_y$ has a closed-form solution derived as

$${\mathbf{f}_y^{l_i}}^{(t+1)} = ({\mathbf{G}_X^{l_i}}^{\mathrm{T}}\mathbf{G}_X^{l_i} + \frac{\kappa_1}{\nu_1}\mathbf{I})^{-1}{\mathbf{G}_X^{l_i}}^{\mathrm{T}}\mathbf{G}_y^{l_i} \tag{28}$$

where $\mathbf{G}_X^{l_i} = [Vector(\mathbf{H}_{x_1}^{l_i}), \cdots, Vector(\mathbf{H}_{x_N}^{l_i})]$ and $\mathbf{G}_y^{l_i} = Vector(\mathbf{H}_y^{l_i} + \mathbf{E}_y^{l_i} - \frac{1}{\nu_1}\mathbf{P}_y^{l_i})$. $Vector(\cdot)$ is an operator that reshapes the image matrix into a vector.

Algorithm 2 The solving algorithm for problem (23)

```
Input: The training samples $\mathbf{H}_X^{l_i}$ and testing samples $\mathbf{H}_y^{l_i}$,
    class number $C$, the trade-off parameters $\kappa_1 = 1, \nu_1 = 1$,
    $\mathbf{M}_X^{l_i} = ({\mathbf{G}_X^{l_i}}^{\mathrm{T}}\mathbf{G}_X^{l_i} + \frac{\kappa_1}{\nu_1}\mathbf{I})^{-1}{\mathbf{G}_X^{l_i}}^{\mathrm{T}}$ where
    $\mathbf{G}_X^{l_i} = [Vector(\mathbf{H}_{x_1}^{l_i}), \cdots, Vector(\mathbf{H}_{x_n}^{l_i})]$
Output: $\mathbf{f}_y^{l_i}$, ${\mathbf{S}_{\mathbf{v}}}_y^{l_i}$
Initialize: $t = 0$, ${\mathbf{E}_y^{l_i}}^{(t)} = -\mathbf{H}_y^{l_i}$, ${\mathbf{P}_y^{l_i}}^{(t)} = \mathbf{0}$
repeat
    Update $\mathbf{f}_y^{l_i}$: Let $\mathbf{G}_y^{l_i} = Vector(\mathbf{H}_y^{l_i} + {\mathbf{E}_y^{l_i}}^{(t)} - \frac{1}{\nu_1}{\mathbf{P}_y^{l_i}}^{(t)})$,
        then ${\mathbf{f}_y^{l_i}}^{(t+1)} = \mathbf{M}_X^{l_i}\mathbf{G}_y^{l_i}$;
    Update $\mathbf{E}_y^{l_i}$: ${\mathbf{E}_y^{l_i}}^{(t+1)} = D_{\frac{1}{\nu_1}}(\mathbf{H}_X^{l_i}({\mathbf{f}_y^{l_i}}^{(t+1)}) - \mathbf{H}_y^{l_i} +$
        $\frac{1}{\nu_1}\mathbf{P}_y^{l_i})^{(t)}$ using Eq. (30);
    Update $\mathbf{P}_y^{l_i}$: ${\mathbf{P}_y^{l_i}}^{(t+1)} = {\mathbf{P}_y^{l_i}}^{(t)} + \nu_1(\mathbf{H}_X^{l_i}({\mathbf{f}_y^{l_i}}^{(t+1)}) -$
        ${\mathbf{E}_y^{l_i}}^{(t+1)} - \mathbf{H}_y^{l_i})$;
until convergence
${\mathbf{S}_{\mathbf{v}}}_y^{l_i} = GNF_{\mathbf{H}_X^{l_i}}(\mathbf{H}_y^{l_i}, \mathbf{f}_y^{l_i}, C)$
```

We can rewrite the Lagrangian function (24) as the follows.

$$\begin{aligned} {\mathbf{E}_y^{l_i}}^{(t+1)} &= \arg\min_{\mathbf{E}_y^{l_i}}(\frac{1}{\nu_1}\|\mathbf{E}_y^{l_i}\|_* + \frac{1}{2} \\ &\|\mathbf{E}_y^{l_i} - (\mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{H}_y^{l_i} + \frac{1}{\nu_1}\mathbf{P}_y^{l_i})\|_F^2) \end{aligned} \tag{29}$$

The above problem (29) involves nuclear-norm minimization. According to singular value thresholding (SVT) algorithm [49], the solution of $E_y^{l_i}$ is computed as follows:

$$\mathbf{E}_y^{l_i} = D_{\frac{1}{\nu_1}}(\mathbf{H}_X^{l_i}(\mathbf{f}_y^{l_i}) - \mathbf{H}_y^{l_i} + \frac{1}{\nu_1}\mathbf{P}_y^{l_i}) \tag{30}$$

where $D(\mathbf{Q})$ is the singular value shrinkage operator, which is defined as follows

$$D(\mathbf{Q}) = \mathbf{U}_{p_1\times r}diag(\{\max(0, \sigma_j - \epsilon)\}_{1\leq j\leq r})\mathbf{V}_{q_1\times r}^{\mathrm{T}} \tag{31}$$

where r is the rank of matrix $\mathbf{Q} \in \mathbb{R}^{p_1\times q_1}$ and $\sigma_1, \cdots, \sigma_r$ are the positive singular values of $\mathbf{Q}$. The singular value decomposition of $\mathbf{Q}$ is derived as

$$\mathbf{Q} = \mathbf{U}_{p_1\times r}\Sigma\mathbf{V}_{q_1\times r}^{\mathrm{T}} \tag{32}$$

where $\Sigma = diag(\sigma_1, \cdots, \sigma_r)$ is a diagonal matrix.

The detailed solving algorithm based on ADMM for problem (23) is summarized in Algorithm 2.

After solving the representation coefficients $\mathbf{f}_y^{l_i}$, GNF is used to obtain the softmax vectors ${\mathbf{S}_{\mathbf{v}}}_y^{l_i} \in \mathbb{R}^{C\times 1}$. In the 1^{st} channel ($l = 0$), we can get one softmax vector ${\mathbf{S}_{\mathbf{v}}}_y^{0^{(1)}}$ which is used to replace the testing image feature $\mathbf{H}_y^{0^{(1)}}$. In the 2^{nd} channel ($l = 1$), we can obtain four softmax vectors ${\mathbf{S}_{\mathbf{v}}}_y^{1^i} (i = 1, \cdots, 4^1)$, which are then transformed into one softmax vector ${\mathbf{S}_{\mathbf{v}}}_y^1 \in \mathbb{R}^{C\times 1}$ by using the max-pooling function,

$${\mathbf{S}_{\mathbf{v}}}_y^1 = \max\{{\mathbf{S}_{\mathbf{v}}}_y^{1^{(1)}}, \cdots, {\mathbf{S}_{\mathbf{v}}}_y^{1^{(4)}}\} \tag{33}$$

or the average pooling function,

$${\mathbf{S}_{\mathbf{v}}}_y^1 = \frac{1}{4}\sum_{i=1}^{4}{\mathbf{S}_{\mathbf{v}}}_y^{1_i} \tag{34}$$

Similarly, in the 3^{rd} channel ($l = 2$), we can obtain four softmax vectors ${\mathbf{S_v}}_y^{1_i} (i = 1, \cdots, 4^2)$, which are then transformed into one softmax vector ${\mathbf{S_v}}_y^2 \in \mathbb{R}^{C \times 1}$ by using the max-pooling function, there is

$$ {\mathbf{S_v}}_y^2 = \max\{{\mathbf{S_v}}_y^{2(1)}, \cdots, {\mathbf{S_v}}_y^{2(16)}\} \tag{35} $$

or the average pooling function,

$$ {\mathbf{S_v}}_y^2 = \frac{1}{16} \sum_{i=1}^{16} {\mathbf{S_v}}_y^{2_i} \tag{36} $$

According to the softmax vector generation process, for each training image $\mathbf{x}_n$, we can also obtain its softmax vector $\mathbf{r}_{x_n}^0$ in the 1^{st} channel, $\mathbf{r}_{x_n}^1$ in the 2^{nd} channel, and $\mathbf{r}_{x_n}^2$ in the 3^{rd} channel, respectively. By putting each softmax vector of each training image together, we will obtain three groups of softmax vector sets ${\mathbf{S_v}}_X^0 \in \mathbb{R}^{C \times N}$, ${\mathbf{S_v}}_X^1 \in \mathbb{R}^{C \times N}$, and ${\mathbf{S_v}}_X^2 \in \mathbb{R}^{C \times N}$. Further, after getting the softmax vectors (${\mathbf{S_v}}_y^0$, ${\mathbf{S_v}}_y^1$, and ${\mathbf{S_v}}_y^2$) of the query sample $\mathbf{y}$ and the softmax vector sets (${\mathbf{S_v}}_X^0$, ${\mathbf{S_v}}_X^1$, and ${\mathbf{S_v}}_X^2$) of the training set $\mathbf{X}$ with max/average pooling by using sparse representation or nuclear-norm matrix regression, the layer-wise Softmax Vector Coding (SVC) is then implemented in the following section.

C. Multi-layer Softmax Vector Coding

In this section, the detailed multi-layer SVC in the proposed deep cascade model is presented. As described in the softmax vector, a sparse probability value vector can be computed. Therefore, sparse coding is modeled for multi-layer SVC in DCM framework.

Specifically, after image coding as described above, three softmax vectors (${\mathbf{S_v}}_y^0$, ${\mathbf{S_v}}_y^1$, and ${\mathbf{S_v}}_y^2$) of a query sample and three softmax vector sets (${\mathbf{S_v}}_X^0$, ${\mathbf{S_v}}_X^1$, and ${\mathbf{S_v}}_X^2$) of the training set are obtained. Here, we concatenate the three softmax vectors of the query $\mathbf{y}$ into one single feature vector $\mathbf{d}_y^1 = [({\mathbf{S_v}}_y^0)^{\mathrm{T}}, ({\mathbf{S_v}}_y^1)^{\mathrm{T}}, ({\mathbf{S_v}}_y^2)^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{3C \times 1}$ and concatenate the three softmax vector sets of the training set into one single feature softmax vector set as $\mathbf{D}_X^1 = [({\mathbf{S_v}}_X^0)^{\mathrm{T}}, ({\mathbf{S_v}}_X^1)^{\mathrm{T}}, ({\mathbf{S_v}}_X^2)^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{3C \times N}$. Then, $\mathbf{d}_y^1$ and $\mathbf{D}_X^1$ are recognized as the input of the 1^{st} layer in our DCM.

Then, $\mathbf{d}_y^1$ is fed into the $GNF_{D_X^1}$ procedure to compute the softmax vector $\mathbf{s}_y^1$, that is then concatenated with $\mathbf{d}_y^1$ to construct the input sample $\mathbf{d}_y^2 = [(\mathbf{d}_y^1)^{\mathrm{T}}, (\mathbf{s}_y^1)^{\mathrm{T}}]^{\mathrm{T}}$ of the 2^{nd} layer. Similarly, each column in $\mathbf{D}_X^1$ is fed into the $GNF_{D_X^1}$ procedure to compute the softmax vector set $\mathbf{S}_X^1$ that is concatenated with $\mathbf{D}_X^1$ to construct the input dictionary $\mathbf{D}_X^2 = [(\mathbf{D}_X^1)^{\mathrm{T}}, (\mathbf{S}_X^1)^{\mathrm{T}}]^{\mathrm{T}}$ of the 2^{nd} layer. In the same way, the feature vector $\mathbf{d}_y^k$ of the query $\mathbf{y}$ and the training dictionary $\mathbf{D}_X^k$ of $\mathbf{X}$ in the k^{th} ($k = 1, \cdots, K$) layer can be computed. Finally, $\mathbf{d}_y^{(K-1)}$ and $\mathbf{D}_X^{(K-1)}$ are fed into the $GNF_{D_X^{(K-1)}}$ to get the final softmax vector $\mathbf{d}_y^K$ of query $\mathbf{y}$ and the final softmax vector set $\mathbf{D}_X^K$ of the training set $\mathbf{X}$. In recognition of $\mathbf{y}$, the label is determined as the class with respect to the maximum value in the final softmax vector, therefore there is

$$ label(\mathbf{y}) = \arg\max_i \mathbf{s}_y^K \tag{37} $$

The detailed procedure of the multi-layer softmax vector coding part is summarized in Algorithm 3.

Algorithm 3 The solving algorithm for Deep Cascade Model

```
Input: $\mathbf{d}_y^1$, $\mathbf{D}_X^1$, the trade-off parameters $\lambda_2$ and $\mu_2$ in sparse
    coding, and the layer number $K$ in DCM.
Output: The predicted label of the query image $\mathbf{y}$.
Initialize: $k = 1$.
repeat
    Compute $\mathbf{s}_y^k = GNF_{D_X^k}(\mathbf{d}_y^k)$ using Eq. (4);
    Compute $\mathbf{S}_X^k = [GNF_{D_X^k}(\mathbf{D}_{x_1}^k), \cdots, GNF_{D_X^k}(\mathbf{D}_{x_N}^k)]$
    using Eq. (4);
    Update $\mathbf{d}_y^{k+1} = [(\mathbf{d}_y^k)^{\mathrm{T}}, (\mathbf{s}_y^k)^{\mathrm{T}}]^{\mathrm{T}}$;
    Update $\mathbf{D}_X^{k+1} = [(\mathbf{D}_X^k)^{\mathrm{T}}, (\mathbf{s}_X^k)^{\mathrm{T}}]^{\mathrm{T}}$;
    $k = k + 1$;
until $k > K$
Find the index of the maximum value in $\mathbf{s}_y^{(K)}$ using Eq.
(37), which shows the label of $\mathbf{y}$
```

D. Remarks on Why DCM Works: An Example

We use a simple example to explain why the proposed method with softmax vectors and pooling operator can amend the misclassified samples.

Suppose a face recognition task of 4 subjects. Given a query image of class 1, for misclassification, a softmax vector $\mathbf{S_v} = [0.25, 0.40, 0.15, 0.20]^{\mathrm{T}}$ will be obtained after sparse coding, frow which one can know that the query image is misclassified as class 2. However, DCM aims to amend the misclassified result. Specifically, in the 2^{nd} channel, as for its subregions, the softmax vectors can be obtained by using DCM(S). Two cases can be considered. (1) There exists one subregion which shows better discrimination than other subregions and the whole image, because it is possible that other subregions are occluded or corrupted. To this end, the softmax vectors of the 4 subregions are supposed to be $\mathbf{S_{v1}} = [0.60, 0.20, 0.10, 0.10]^{\mathrm{T}}$, $\mathbf{S_{v2}} = [0.30, 0.45, 0.10, 0.15]^{\mathrm{T}}$, $\mathbf{S_{v3}} = [0.25, 0.50, 0.10, 0.15]^{\mathrm{T}}$, and $\mathbf{S_{v4}} = [0.30, 0.35, 0.25, 0.10]^{\mathrm{T}}$, respectively. From the softmax vectors, we see that misclassification is encountered based on three subregions. However, by using max-pooling operator, we will obtain the final softmax vector $\mathbf{S_v} = [0.60, 0.50, 0.25, 0.15]^{\mathrm{T}}$, from which we can see that the query image is correctly classified as class 1, instead of class 2. (2) The above extreme case is actually unusual, however, it is more likely that most subregions are discriminative due to that the occlusions and corruption would not always appear. Therefore, the average pooling works under this condition. We let $\mathbf{S_{v1}} = [0.35, 0.25, 0.15, 0.25]^{\mathrm{T}}$, $\mathbf{S_{v2}} = [0.40, 0.20, 0.30, 0.10]^{\mathrm{T}}$, $\mathbf{S_{v3}} = [0.20, 0.50, 0.10, 0.20]^{\mathrm{T}}$, and $\mathbf{S_{v4}} = [0.45, 0.15, 0.20, 0.20]^{\mathrm{T}}$ represent the softmax vectors of the 4 subregions, respectively. We see that the misclassification is encountered for the 3^{rd} softmax vector. By using the average pooling, the final softmax vector $\mathbf{S_v} = [0.35, 0.28, 0.19, 0.19]^{\mathrm{T}}$ can be obtained, which can also amend the misclassified image.

IV. Experiments

In this section, the experimental results of our proposed DCM method on publicly benchmark databases, including Extended Yale B database [50], AR database [51], CMU

TABLE I: Recognition rates (%) on Extended Yale B database with different number of training samples per subject

Algorithm	15	20	25	30
CRC	91.39	94.26	95.91	97.04
SRC	91.72	93.71	95.56	96.37
CESR	77.92	83.42	85.68	88.51
RSC	95.01	97.04	97.81	98.40
HQA	93.39	93.99	90.19	92.41
HQM	91.14	94.15	95.29	96.46
NMR	93.50	96.29	97.57	98.54
RMR	93.56	94.08	92.15	92.72
FDDL	93.44	94.92	96.38	96.94
LRSDL	94.92	96.69	97.88	98.31
DCM(N)	93.17	95.97	97.38	98.38
DCM(S)	**98.87**	**99.51**	**99.63**	**99.79**

TABLE II: Recognition rates (%) on CMU PIE database with different number of training samples per subject

Algorithm	15	20	25	30
CRC	89.76	92.42	93.80	94.61
SRC	88.97	91.14	92.62	93.71
CESR	79.47	84.55	87.16	89.24
RSC	92.93	94.91	95.98	96.38
HQA	80.23	84.77	89.73	91.98
HQM	86.15	89.72	91.90	93.24
NMR	91.77	93.54	94.75	95.46
RMR	91.99	94.02	94.60	95.38
FDDL	90.44	92.12	91.00	93.87
LRSDL	92.12	94.40	92.34	95.21
DCM(N)	90.70	92.63	94.05	94.99
DCM(S)	**93.79**	**95.59**	**96.37**	**96.84**

TABLE III: Recognition rates (%) on AR database with different number of training samples per subject

Algorithm	8	11	14	17
CRC	94.96	97.01	98.06	98.53
SRC	95.49	97.50	98.45	98.87
CESR	60.53	68.88	79.35	81.98
RSC	94.78	96.93	98.08	99.11
HQA	80.57	67.60	91.37	95.60
HQM	73.07	81.31	86.30	90.63
NMR	94.18	97.01	98.02	98.47
RMR	**96.39**	**97.84**	98.49	99.00
FDDL	93.00	95.93	96.38	96.44
LRSDL	95.28	97.51	98.02	98.67
DCM(N)	91.96	95.02	96.82	97.92
DCM(S)	96.17	97.78	**98.66**	**99.11**

PIE database [52], FRGC database [53] are presented for performance verification. We compare the proposed method with state-of-the-art representation based learning methods for face recognition, such as CRC [12], SRC [2], CESR [16], RSC [14], half-quadratic with the additive form (HQA) [17], half-quadratic with the multiplicative form (HQM) [17], N-MR [24], RMR [25], FDDL [54], and LRSDL [55]. FDDL and LRSDL are dictionary based learning methods. Note that RMR refers to two sub-models, i.e., RMRL1 and RMRL2, in which the best one is reported. For fair comparison, these methods have been fully tuned to achieve the best results by choosing the optimal parameters. Specifically, the experiments are divided into two groups: uncorrupted data and corrupted data. Note that, for differentiating the representation model used in the multi-level image coding part, we use DCM(S) and DCM(N) to demonstrate the choice of sparse representation and nuclear-norm matrix regression in the GNF module, respectively. The best results with respect to max-pooling or average pooling in DCM are reported in this paper.

For DCM(S), λ_1 and μ_1 are the hyper parameters for query and gallery set in the image coding part. For DCM(N), κ_1 and ν_1 are the hyper parameters for query and gallery set in the image coding part. λ_2 and μ_2 are the hyper parameters for query and gallery set in the multi-layer softmax vector coding part. It is worth noting that all experiments are conducted on the raw face image pixels without extra hand-crafted feature extraction. In DCM(S), we reshape each image into a feature vector in the image coding part for computing the softmax vectors based on model (9). In DCM(N), the raw face image matrix is directly used to calculate the softmax vectors by following the NMR method based on model (22).

A. Experiments on Uncorrupted Data

We randomly split the each database into two parts: training set and testing set. Generally, 10 random splits are experimented for all compared methods and the average recognition rates are reported. The images in the four image databases (Extended Yale B, CMU PIE, AR and FRGC) are cropped and resized into 32×32. For our DCM(S), we set $\lambda_1 = 10^{-4}$, $\mu_1 = 10^{-1}$, $\lambda_2 = 10^{-4}$, and $\mu_2 = 1$. For our DCM(N), we set $\kappa_1 = 1$, $\nu_1 = 1$, $\lambda_2 = 10^{-4}$, and $\mu_2 = 1$. For CMU PIE database, we set $\mu_2 = 10^{-2}$.

1) Results on Extended Yale B Database: The Extended Yale B database contains 2414 frontal face images of 38 individuals, and each of them has around 64 near frontal images under different illuminations. We randomly select 15, 20, 25, 30 images per person for training, and the rest for testing. The average recognition rates of 10 random splits by using different methods based on this dataset are summarized in Table I, in which the best recognition rates are highlighted by bold numbers. It can be observed that our method, i.e. DCM(S), can achieve the best recognition rates. Typically, when the number of training samples is 15, the recognition rate of our method is almost 4% higher than the RSC that ranks the second among the compared methods. Besides, we also observe that our DCM can achieve competitive better recognition result when there are few training samples. It is worth noting that DCM(S) is much better than DCM(N), which demonstrates that the choice of representation model is task specific. For uncorrupted data, the sparse representation model outperforms nuclear-norm based representation in multi-level image coding. This is also verified by the NMR method which achieves slightly worse result than the RSC algorithm.

Fig. 4: Example images for CMU PIE database.

Fig. 5: Example images for FRGC databae.

2) Results on CMU PIE Database: The CMU PIE face database contains totally 41,368 face images from 68 subjects. The image sets under 5 near frontal poses (C05, C07, C09, C27 and C29) are used in our experiment. The example images of frontal pose under C05 are shown in Fig. 4. We randomly select 15, 20, 25, 30 images from each subject as training samples and the remaining images are used as test samples. The classification rates with different number of training samples by using different methods are summarized in Table II, from which we can clearly observe that our method DCM(S) outperforms the compared methods with different number of training images per subject.

3) Results on AR Database: The AR face database contains about 4,000 color face images from 126 subjects, which are shown in frontal faces with different facial expressions, illuminations and disguises, respectively. In this experiment, we select a subset consisting of 2600 images from 50 female and 50 male subjects. We randomly select 8, 11, 14, 17 images per subject as the training samples and the rest of images are directly used as test samples. The recognition rates with different number of training samples per subject by using different algorithms on this database are summarized in Table III. From the table, we can observe that our method DCM(S) achieves competitive recognition rates when we choose 8 or 11 images per person for training. Additionally, when 14 or 17 images per subject are used for training, the recognition performance is higher than other methods, which demonstrates the effectiveness of our DCM. Notably, DCM(S) always outperforms DCM(N) under uncorrupted data.

4) Results on FRGC Database: The Face Recognition Grand Challenge (FRGC) v2 database contains 12,776 training images, 16,028 controlled target images, and 8,014 uncontrolled query images. Several example face images are shown in Fig. 5. For this database, we choose a subset of FRGC database by following the same experimental setting as NMR [24]. This subset contains 220 persons and 20 images per person in different conditions such as large illumination variations, low resolution, and blurring are included. We use the first 10 images per person for training and the remaining 10 images are used for testing. Specifically, the recognition rates (%) for different methods are shown in Table IV, from which we can see that the proposed DCM(S) shows the best recognition performance in extreme conditions.

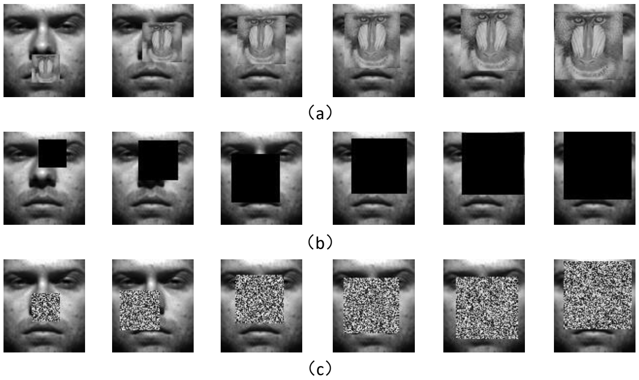


Fig. 6: Example images with different level of occlusions from 10% to 60% on Extended Yale B database. (a) *Baboon* block; (b) *Dark* block; (c) *Random* block

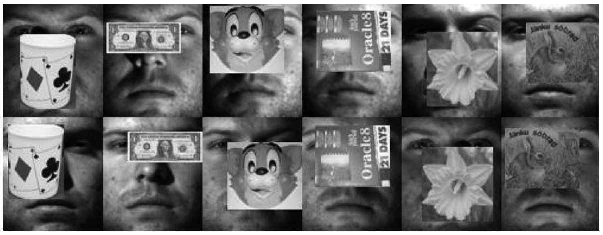


Fig. 7: Example images with different kinds of object occlusions on Extended Yale B database.

B. Experiments on Corrupted Data

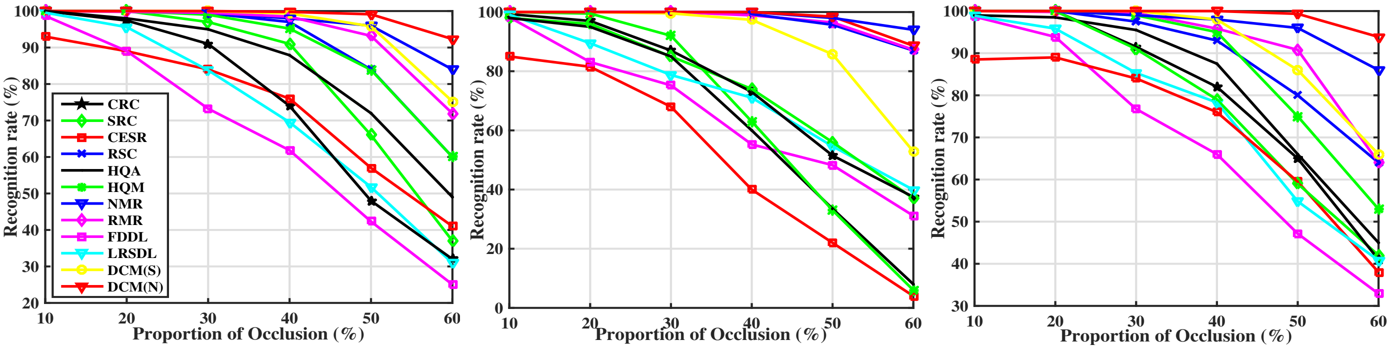
In this section, by following the same experimental setting as [24], the experiments on corrupted data are conducted to verify the effectiveness of the proposed DCM method for robust face recognition with different kinds of occlusions.

1) Experiments under added occlusions on Extended Yale B faces: For the Extended Yale B database, manually added occlusions are experimented and each image is resized to 96×84 pixels. In parameter setting, for DCM(S), we set $\lambda_1 = 10^{-4}$, $\mu_1 = 10^{-2}$, $\lambda_2 = 10^{-4}$, and $\mu_2 = 10^{-1}$. For DCM(N), we set $\kappa_1 = 1$, $\nu_1 = 1$, $\lambda_2 = 10^{-4}$, and $\mu_2 = 10^{-1}$. In this section, 4 experiments with 4 kinds of manually added occlusions shown in Fig. 6(a), Fig. 6(b), Fig. 6(c), and Fig. 7 are conducted, respectively.

Baboon Block Occlusion. In the 1^{st} experiment, by following the same experimental setting as [2], [24], we use the Subset 1 and 2 of Extended Yale B database for training and Subset 3 for testing. For occlusion, a randomly located square block in the test images is replaced by a *baboon* image

TABLE IV: Recognition results on FRGC database

Method	CRC	SRC	CESR	RSC	HQA	HQM	NMR	RMR	FDDL	LRSDL	DCM(S)	DCM(N)
Accuracy (%)	92.2	89.2	81.9	92.0	84.7	91.9	93.3	92.6	84.1	89.0	**93.6**	91.3



(a) (b) (c)

Fig. 8: Results on Extended Yale B with different levels of occlusions. (a) *Baboon*; (b) *Dark*; (c) *Random*.

with a varying block size, as shown in Fig. 6(a). The varying block size is determined by 6 levels of the occlusion from 10% to 60%. The recognition rates with 6 levels of *baboon* occlusion are shown in Fig. 8(a), from which we can clearly observe that our DCM(N) method outperforms other state-of-the-art methods. With low occlusion levels (e.g., 10, 20, and 30%), RMR, NMR, RSC, and HQM shows competitive performance with the proposed DCM model. Particularly, when the occlusion level is higher than 40%, our DCM(N) achieves much better and more stable recognition performance than NMR method. For other methods, the performance drops sharply when occlusion is larger. Therefore, the robustness of the proposed DCM in face recognition under large occlusions is well demonstrated. It is worth noting that the DCM(N) outperforms DCM(S) when it encounters occlusions, which shows the significant importance of low-rank property in modeling the representation error.

Dark Block Occlusion. In the 2^{nd} experiment, with the same training and testing data as experiment 1, a randomly located square block in each testing image is replaced with a *dark block* image whose elements are all 0 with a varying block size, as shown in Fig. 6(b). The varying block size is determined by 6 levels of occlusion from 10% to 60%. Obviously, the *dark block* occlusion is intra-correlated with zero pixels. The experimental results are shown in Fig. 8(b), from which we can observe that the proposed DCM method outperforms the compared methods when the occlusion levels are from 10 to 50 percent. When larger *dark block* occlusion is encountered, DCM(N) performs competitively similar with RMR, NMR and RSC methods.

Random Block Occlusion. In the 3^{rd} experiment, we also use the Subset 1 and 2 of Extended Yale B database for training and Subset 3 for testing. A randomly located square block in the test images is replaced with a *random block* image whose elements are random integral number from 0 to 255 with a varying block size, as shown in Fig. 6(c). The varying block size is also determined by 6 levels of occlusion which varies from 10% to 60%. It is clear that the *random block* occlusion is pixel independent. The experimental results are shown in Fig. 8(c), from which we can see that the recognition rates of our DCM(N) are always superior to the results of other state-of-the-art methods. The NMR ranks the second, which is better than DCM(S). Another finding is that DCM(N) always outperforms DCM(S) under corrupted data, which is contrary under uncorrupted data. The experiments on relevant and irrelevant noise show that the proposed DCM model has significantly better performance and robustness in large occlusion conditioned face recognition.

Multiple Object Occlusions. In the 4^{th} experiment, by using the same training and testing data as that in previous experiments, different kinds of objects, such as cup, dollar, cartoon mask, book, flower and puzzle are used as occlusion to cover a block in each test image, as shown in Fig. 7. The recognition accuracies are shown in Table V, from which we can observe that the proposed DCM(N) achieves state-of-the-art performance (97.8%) over other compared methods (96.1% for NMR). The experimental results demonstrate that our DCM method under nuclear-norm based representation model gives rise to better robustness than others when handling face recognition tasks under occlusions.

2) Experiments under real-world occlusions on AR faces: For the AR database, real world occlusions (e.g., sunglass vs. scarf) are experimented and each image is resized to 50×40 pixels. A subset of AR database that contains 120 individuals (65 men and 55 women) is exploited. For each person, the photos are taken in two sessions and 13 photos are contained per session. Example photos of one person in AR database are shown in Fig. 9. In parameter setting, for DCM(S), we set $\lambda_1 = 10^{-2}$, $\mu_1 = 1$, $\lambda_2 = 10^{-4}$, and $\mu_2 = 10^{-1}$. For DCM(N), we set $\kappa_1 = 1$, $\nu_1 = 1$, $\lambda_2 = 10^{-4}$, and $\mu_2 = 10^{-1}$. In this section, experiments with real world occlusions are conducted on AR database to verify the effectiveness and robustness of the proposed method. As described in Fig. 9, 8 frontal face images per person without occlusion, composed of the first 4 images from Session 1 and 2, are used as training images. The test images are divided into two groups: (1)

TABLE V: Recognition results under different kinds of occlusions on Extended Yale B database

Method	CRC	SRC	CESR	RSC	HQA	HQM	NMR	RMR	FDDL	LRSDL	DCM(S)	DCM(N)
Accuracy (%)	57.0	53.5	72.2	92.0	83.6	91.9	96.1	82.7	23.3	30.0	82.2	**97.8**

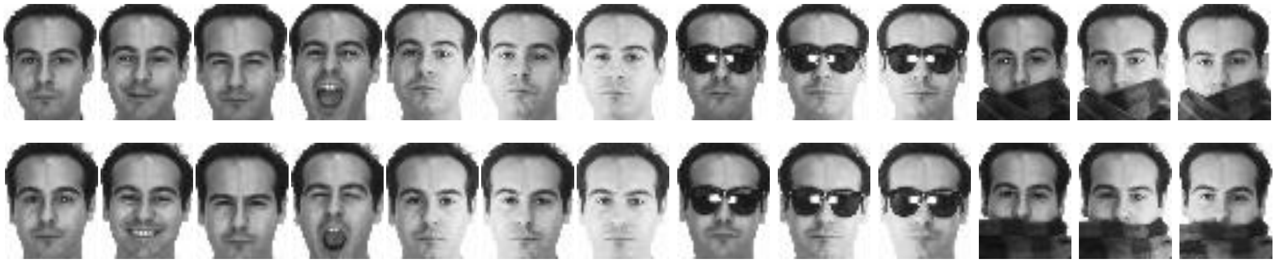
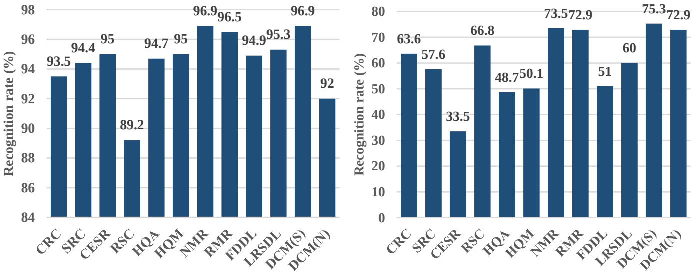


Fig. 9: Example images of the first person in AR face database. The face images in the first row are from Session 1 and the faces in the second row are from Session 2. The faces without occlusion are used for training. The faces with sunglass occlusion are used for test (Group 1) and the faces with scarve occlusion are used for test (Group 2).



(a) (b)

Fig. 10: Recognition performance with real-world occlusion on AR database. (a) sunglass occlusion; (b) scarf occlusion.

in Group 1, there are 6 images per person with $sunglass$ occlusion from both sessions; (2) in Group 2, there are 6 images per person with $scarf$ occlusion from both sessions. The recognition rates of the CRC, SRC, CESR, SRC, HQA, HQM, NMR, RMR, FDDL, LRSDL, and the proposed DCM including DCM(S) and DCM(N) are reported in Fig. 10. We can observe from Fig. 10 that the proposed DCM(S) achieves the best test performance for each group. For $sunglass$ occlusion, the proposed DCM(S) performs competitively well with state-of-the art NMR method. For larger $scarf$ occlusion, the proposed DCM(S) can achieve much higher recognition performance than other methods. The DCM(N) is slightly inferior to the NMR method. Through the test results with real world occlusion, the proposed DCM is demonstrated to be more effective and robust than other state-of-the-art methods in face recognition under larger occlusion.

V. Discussion

From the face recognition experiments on several benchmark databases under uncorrupted and corrupted conditions, the effectiveness of the proposed DCM(S) and DCM(N) methods is fully demonstrated. Some insightful and interesting perspectives on the proposed methods are observed.

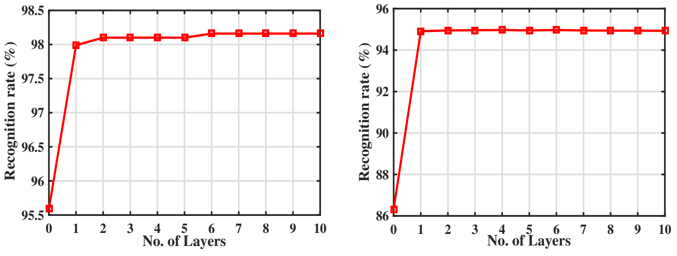
In the proposed DCM method, a three-level spatial pyramid structure is used for multi-level image coding with subregions. In the 1^{st} level, each face image is viewed as one region. In the 2^{nd} level, each face image is equally divided into 4 subregions. In the 3^{rd} level, each face image is equally divided into 16 subregions. Therefore, each image can be represented 1, 4 and 16 times, respectively, which is useful to face recognition under large occlusions. Additionally, the multi-layer sparse softmax vector coding is beneficial to improve the discriminative ability of the represented features. As a result, DCM(S) and DCM(N) outperform state-of-the-art representation models under uncorrupted data and corrupted data, respectively. For encoding the softmax vectors from the GNF function in each subregion, max-pooling or average pooling is introduced for feature vector generation.

From the face recognition performance under different level of occlusions, the proposed DCM can still achieve the state-of-the-art recognition performance and robustness. The examples still hold when occlusions exist in the test images. The subregions show local information of each face image. Similar to human being who can recognize a person based on parts of each face image, the local information can also be used to recognize a person with computer vision techniques. To achieve this goal, a three-level spatial pyramid structure and max/pooling operator are designed to transform the image and its subregion into softmax vectors by using sparse coding method. Most of sparse coding methods are designed under the statistical prior of Laplacian noise existed in the face images. However, in real-world application, it is unknown whether the corruptions exist. Therefore, the proposed DCM is formulated for robust face recognition of clean or dirty data.

It is also worth noting that DCM(S) is better than DCM(N) in recognition when there is no corruption in the test data. On the contrary, when there is corruption, DCM(N) is better than DCM(S). This demonstrates that the reconstruction errors are random and approximately obey Gaussian distribution for uncorrupted data. Therefore, it is better to model the errors by using sparse representation model than low-rank model. Instead, the representation error matrix of a corrupted query image generally shows a low-rank structure when encountered with manually added occlusions, under which nuclear-norm

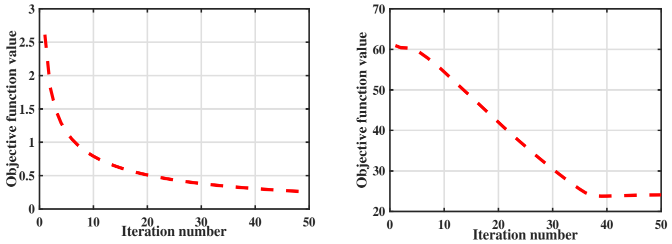
TABLE VI: Computation time (s) of all methods on Extended Yale B database with 15 training images per subject

Method	CRC	SRC	CESR	RSC	HQA	HQM	NMR	RMR	FDDL	LRSDL	DCM(S)	DCM(N)
Time (s)	**3.32**	172.77	163.03	6305.54	1527.70	1313.81	81.88	56.79	18.92	396.87	314.04	479.56



(a) (b)

Fig. 11: Performance variation with increasing number of layers on two databases. (a) Extended Yale B (b) CMU-PIE.



(a) (b)

Fig. 12: The convergence curves of Algorithm 1 (a) and Algorithm 2 (b) on Extended Yale B database.

modeling is therefore more suitable to quantize the corruption. Therefore, the performance promotion of the proposed DCM is rational, feasible, and interpretable.

VI. Analysis of Layers, Convergence, and Computation Time

A. Analysis of Layers

In this paper, the DCM is a multi-layer learning framework consisting of image coding and softmax vector coding. It is insightful to observe the performance variation with the increasing number of layers. Specifically, the performance curves based on benchmark databases under uncorrupted condition are presented in Fig. 11. In analysis, 15 images per subject are selected for training and the total number of layers is set as 10. From Fig. 11, we can observe that with the increasing number of layers, the recognition rates show a rising tendency on Extended Yale B dataset and keep unchanged when the layer number is higher than 2 on CMU-PIE dataset. Thus, *for different datasets, one can empirically set the layer number as $K = 2$*. Note that the recognition rate of $Layer = 0$ represents the test result based on the output feature of multi-level image coding without subsequent softmax vector coding. The effectiveness of the proposed deep cascade model (DCM) over general sparse representation models is demonstrated.

B. Analysis of Convergence

To illustrate the convergence of the solving algorithm for problem (9) and (23), the objective function values with 50 iterations on the Extended Yale B database by using the Algorithm 1 and Algorithm 2 are presented in Fig. 12(a) and Fig. 12(b), respectively. We can observe from Fig. 12 that the DCM model shows a stable convergence.

C. Analysis of Computation Time

Due to that the proposed DCM is a multi-level and multi-layer representation learning model, more computation time is needed in training phase. For better insight of the computational complexity, we present the training time of different methods in Table VI, from which it is rational that the DCM costs more time than others, because each image is divided into 21 (1+4+16) subregions in representation. However, it is still several times faster than RSC, HQA and HQM methods. By comparing with the very hot deep convolutional neural network, the proposed sparse representation based deep cascade model should be much more efficient due to the hierarchical learning layer-by-layer without back-propagation. All experiments are conducted on Matlab 2015, by using a computer with CPU E3-1231 v3 3.40GHz and 16G RAM.

VII. Conclusion

This paper presented a novel softmax vector coding based deep cascade model (DCM) for robust face recognition under large occlusions. There are three key merits in the proposed DCM model. First, we propose a multi-level image coding module which includes a three-level spatial pyramid structure (three channels) for processing each image. Second, we propose a getting new feature (GNF) operator, in which existing representation models can be easily integrated. To that end, in this paper, sparse representation and nuclear-norm based matrix regression are considered, which therefore formulates two methods, DCM(S) and DCM(N). Each level is recognized as a channel for transforming the image and its subregions into softmax vectors that carry class discrimination information by using GNF operator. Particularly, in the 2^{nd} and 3^{rd} channel, the off-the-shelf pooling functions (e.g., max vs. average) are introduced to encode the softmax vectors of multiple subregions into one single feature representation vector. Third, a hierarchical and multi-layer softmax vector coding module is designed for deep cascade representation, which is beneficial to the learning of discriminative facial identity features. Extensive experiments on several benchmark face recognition databases under uncorrupted and corrupted conditions are conducted. Experimental results demonstrate the superior performance of the proposed DCM framework over other state-of-the-art representation models. Remarkably, our proposed method shows significantly superior performance to counterparts under large corruptions.

References

[1] I. Naseem, R. Togneri, and M. Bennamoun, "Linear regression for face recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, p. 2106, 2010.

[2] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, p. 210, 2009.

[3] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[4] R. Saab, R. Chartrand, and O. Yilmaz, "Stable sparse approximations via nonconvex optimization," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2008, pp. 3885–3888.

[5] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Processing Letters*, vol. 14, no. 10, pp. 707–710, 2007.

[6] M. Schmidt, G. Fung, and R. Rosales, *Fast Optimization Methods for L1 Regularization: A Comparative Study and Two New Approaches*. Springer Berlin Heidelberg, 2007.

[7] L. Zhang, W. Zuo, and D. Zhang, "Lsdt: Latent sparse domain transfer learning for visual adaptation," *IEEE Trans. Image Processing*, vol. 25, no. 3, pp. 1177–1191, 2016.

[8] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint ? 2,1 -norms minimization," in *NIPS*, 2010, pp. 1813–1821.

[9] C. Hou, F. Nie, X. Li, D. Yi, and Y. Wu, "Joint embedding learning and sparse regression: a framework for unsupervised feature selection." *IEEE Trans. Cybernetics*, vol. 44, no. 6, p. 793, 2014.

[10] L. Zhang and D. Zhang, "Robust visual knowledge transfer via extreme learning machine based domain adaptation," *IEEE Trans. Image Processing*, vol. 25, no. 10, pp. 4959–4973, 2016.

[11] Y. Xu, D. Zhang, J. Yang, and J. Y. Yang, "A two-phase test sample sparse representation method for use with face recognition," *IEEE Trans. CSVT*, vol. 21, no. 9, pp. 1255–1262, 2011.

[12] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *ICCV*, 2012, pp. 471–478.

[13] S. Cai, L. Zhang, W. Zuo, and X. Feng, "A probabilistic collaborative representation based approach for pattern classification," in *CVPR*, 2016, pp. 2950–2959.

[14] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *CVPR*, 2011, pp. 625–632.

[15] M. Yang, T. Song, F. Liu, and L. Shen, "Structured regularized robust coding for face recognition," *IEEE Trans. Image Processing*, vol. 22, no. 5, pp. 1753–1766, 2013.

[16] R. He, W. S. Zheng, and B. G. Hu, "Maximum correntropy criterion for robust face recognition," *IEEE TPAMI*, vol. 33, no. 8, 2011.

[17] R. He, W. S. Zheng, T. Tan, and Z. Sun, "Half-quadratic-based iterative minimization for robust sparse representation," *IEEE Trans. PAMI*, vol. 36, no. 2, pp. 261–275, 2014.

[18] R. He, W. S. Zheng, B. G. Hu, and X. W. Kong, "A regularized correntropy framework for robust pattern recognition," *Neural Computation*, vol. 23, no. 8, pp. 2074–2100, 2011.

[19] I. Naseem, R. Togneri, and M. Bennamoun, "Robust regression for face recognition," in *ICPR*, 2010, pp. 1156–1159.

[20] X. X. Li, D. Q. Dai, X. F. Zhang, and C. X. Ren, "Structured sparse error coding for face recognition with occlusion," *IEEE Trans. Image Processing*, vol. 22, no. 5, p. 1889, 2013.

[21] K. Jia, T. H. Chan, and Y. Ma, "Robust and practical face recognition via structured sparsity," in *ECCV*, 2012, pp. 331–344.

[22] C. Lang, B. Cheng, S. Feng, and X. Yuan, "Supervised sparse patch coding towards misalignment-robust face recognition," *J. Vis. Comm. Image Representation*, vol. 24, no. 2, pp. 103–110, 2013.

[23] J. Lu, G. Wang, W. Deng, and P. Moulin, "Simultaneous feature and dictionary learning for image set based face recognition," in *ECCV*, 2014, pp. 265–280.

[24] J. Yang, L. Luo, J. Qian, Y. Tai, F. Zhang, and Y. Xu, "Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes," *IEEE Trans. PAMI*, vol. 39, no. 1, pp. 156–171, 2017.

[25] J. Xie, J. Yang, J. Qian, Y. Tai, and H. Zhang, "Robust nuclear norm-based matrix regression with applications to robust face recognition," *IEEE Trans. Image Processing*, vol. 26, no. 5, pp. 2286–2295, 2017.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2015, pp. 770–778.

[28] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.

[29] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *CVPR*, 2014, pp. 1701–1708.

[30] Z. Zhou and J. Feng, "Deep forest: Towards an alternative to deep neural networks," *CoRR*, vol. abs/1702.08835, 2017. [Online]. Available: http://arxiv.org/abs/1702.08835

[31] S. Wang, L. Zhang, and W. Zuo, "Class-specific reconstruction transfer learning via sparse low-rank constraint," in *ICCV*, Oct 2017.

[32] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation." *IEEE Trans. PAMI*, vol. 35, no. 1, pp. 171–184, 2013.

[33] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *ICCV*, 2012, pp. 1615–1622.

[34] C. Zhang, J. Liu, Q. Tian, and C. Xu, "Image classification by non-negative sparse coding, low-rank and sparse decomposition," in *CVPR*, 2014, pp. 1673–1680.

[35] J. Chen and Z. Yi, "Sparse representation for face recognition by discriminative low-rank matrix recovery," *Journal of Visual Communication & Image Representation*, vol. 25, no. 5, pp. 763–773, 2014.

[36] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *IEEE Trans. PAMI*, vol. 34, no. 11, pp. 2233–2246, 2012.

[37] D. Huang, R. Cabral, and F. D. L. Torre, "Robust regression," *IEEE Trans. PAMI*, vol. 38, no. 2, pp. 363–375, 2016.

[38] Y. C. F. Wang, C. P. Wei, and C. F. Chen, "Low-rank matrix recovery with structural incoherence for robust face recognition," in *CVPR*, 2012, pp. 2618–2625.

[39] S. F. Chang, D. T. Lee, D. Liu, and I. Jhuo, "Robust visual domain adaptation with low-rank reconstruction," in *CVPR*, 2012, pp. 2168–2175.

[40] Y. Xu, X. Fang, J. Wu, X. Li, and D. Zhang, "Discriminative transfer subspace learning via low-rank and sparse representation," *IEEE Trans. Image Processing*, vol. 25, no. 2, p. 850, 2016.

[41] M. Okutomi, S. Yan, S. Sugimoto, G. Liu, and Y. Zheng, "Practical low-rank matrix approximation under robust l1-norm," in *CVPR*, 2012, pp. 1410–1417.

[42] L. Luo, J. Yang, J. Qian, and J. Yang, "Nuclear norm regularized sparse coding," in *ICPR*, 2014, pp. 1834–1839.

[43] J. Liu and L. Zhang, "Sparse softmax vector coding based deep cascade model," in *CCCV*, 2017, pp. 603–614.

[44] J. Yang and Y. Zhang, "Alternating direction algorithms for ℓ_1-problems in compressive sensing," *Siam Journal on Scientific Computing*, vol. 33, no. 1, pp. 250–278, 2009.

[45] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," *Computers & Mathematics with Applications*, vol. 2, no. 1, pp. 17–40, 1976.

[46] M. Chen, Z. Lin, Y. Ma, and L. Wu, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *Arxiv*, 2010.

[47] A. Hansson, Z. Liu, and L. Vandenberghe, "Subspace system identification via weighted nuclear norm optimization," in *IEEE Conf. Decision and Control*, 2012, pp. 3439–3444.

[48] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations & Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[49] J.-F. Cai, Cand, E. J. s, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *Siam Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2008.

[50] A. S. Georghiades, P. N. Belhumeur, and D. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE TPAMI*, vol. 23, no. 6, pp. 643–660, 2001.

[51] A. M. Martinez, "The ar face database," *Technical Report*, vol. 24, 1998.

[52] T. Sim, S. Baker, and M. Bsat, "The cmu pose, illumination, and expression (pie) database," in *IEEE AFGR*, 2002, p. 53.

[53] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," in *CVPR*, 2005, pp. 947–954.

[54] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Sparse representation based fisher discrimination dictionary learning for image classification," *IJCV*, vol. 109, no. 3, pp. 209–232, 2014.

[55] T. H. Vu and V. Monga, "Fast low-rank shared dictionary learning for image classification," *IEEE Trans. Image Processing*, 2017.