Hand Vein Recognition via Discriminative Convolutional Sparse Coding

Ali Nozaripour [1] * and Hadi Soltanizadeh [2]

***Abstract*-- Personal identification based on vein pattern is one of the latest biometric approaches that have attracted lots of attention. Besides, Convolutional Sparse Coding (CSC) is a popular model in the signal and image processing communities, resolving some limitations of the traditional patch-based sparse representations. As most existing CSC algorithms are suited for image restoration, we present a novel discriminative model based on CSC for dorsal hand vein recognition in this paper. The proposed method, discriminative local block coordinate descent (D-LoBCoD), is based on extending the LoBCoD algorithm by incorporating the classification error into the objective function that considers the performance of a linear classifier and the representational power of the filters simultaneously. Thus, for training, in each iteration, after updating the sparse coefficients and convolutional filters, we minimize the classification error by updating the classifier's parameters according to the label information. Finally, after training, the label of the query image will be determined by the trained classifier. One thousand two hundred dorsal hand vein images taken from 100 individuals are used to verify the validity of the proposed methods. The experimental results show that our method outperforms other competing methods. Further, we demonstrate that our proposed method is less dependent on the number of training samples because of capturing more representative information from the corresponding images.**

***Index Terms*-- Convolutional Sparse Coding, Dorsal hand vein pattern, Image classification, Region of interest, Sparse representation.**

I. INTRODUCTION

Personal identification based on subcutaneous vein patterns is in the spotlight since the respective devices can take contactless images of the body and also for the veins are localized inside the body, this method is highly safe and resistant to fraud [1, 2]. One of the most important concerns for an identification system is selecting a similar and uniform area in all images called the Region Of Interest (ROI), which greatly impacts the final result [3]. In this work, to eliminate restrictions on hand position while taking the image, a robust method against hand spin named floating ROI (FROI) is used, which is proposed in our previous work [4]. Another problem with the identification systems is the type of used algorithm for comparing and image classification. Recently, sparse representation algorithms have grown and have succeeded in some applications such as image classification [2, 5-10], image compression and restoration [11, 12], compressed sensing [13], denoising [14], and deep learning [15]. An overview of dictionary learning can be found in [16].

Since in sparse representation, each data sample is represented as a linear combination of a few atoms (dictionary columns), it cannot capture shifted local patterns common in image samples. Furthermore, learning the dictionary in high-dimensional signals is not easy [17]. Many algorithms suggest training a local model on fully overlapping patches taken from a sample to cope with these problems. This patch-based technique is similar to manually convolving the dictionary with the sample [18]. But, patch-based approaches are known to be sub-optimal because they ignore the relations between neighboring patches [19]. In addition, as each sample element (e.g., an image pixel) is in several overlapping patches, the separately learned representations may not be consistent. Moreover, the resultant representation is highly redundant [20].

The Convolutional sparse coding (CSC) model poses an alternative approach to meet these challenges. This model assumes that the signal can be represented by the sum of a few filters convolved with the corresponding sparse feature maps. In this model, by learning a shift-invariant dictionary composed of many filters, local patterns are extracted at translated positions of the samples by convolution. So, we no longer need to use overlapping patches [18]. The CSC model has been the subject of extensive research in the past several years. It has performed better than sparse coding in many images processing applications such as image separation [21], super-resolution [22], inpainting [23], biomedical applications [24], image fusion [25], audio processing [26], and source separation [27].

Several approaches have been proposed to solve the CSC problem in spatial and frequency domains [23, 28-30]. But all these methods operate in batch mode, so they are high-cost in terms of space and time on large data sets. Recently, some CSC methods have used the online learning approach. Thus, the data samples need not be stored after being processed, significantly reducing the algorithms' time and space complexities [18, 31, 32]. On the other hand, most of the existing CSC methods are unsupervised, and in some works of image classification like [33, 34], the CSC model is only used as a feature extractor; therefore, other classifiers are needed for classification. Also, in

1- A. Nozaripour is with the Department of Electrical and Computer Engineering Hakim University, Sabzevar, Iran.
2- H. Soltanizadeh is with the Faculty of Electrical and Computer Engineering, Semnan University, Semnan, Iran.
Corresponding author: khayamboy@yahoo.com

[35, 36], the authors combined the CSC framework with the traditional Sparse Representation-based Classification (SRC) method. They proposed a convolutional sparse coding classification (CSCC) model for image classification, which introduces the label during the training step.

However, as far as we know, most current CSC-based algorithms often rely on the Alternating Direction Method of Multipliers (ADMM) solvers that operate in the Fourier domain for extracting the signal representation of the model and training its corresponding filters. Thus, they lose the connection to the patch-based processing paradigm, as widely practiced in many signals and image processing applications.

To tackle this, Papyan et al. [21] have proposed Slice Based Dictionary Learning (SBDL), which adopts a local point of view and trains the filters in terms of only local computations in the signal domain. But their method still relies on the ADMM and requires N auxiliary variables that cause increasing the memory requirements. Recently, Zisselman et al. [17] proposed a novel CSC model called Local Block Coordinate Descent (LoBCoD) for image inpainting and image fusion. In this method, the CSC filter learning and the global pursuit solving are done with local computations in the original domain. Unlike other CSC methods, the LoBCoD algorithm operates without auxiliary variables and extra parameters for tuning in the pursuit stage.

This paper proposes a new approach to extend the LoBCoD algorithm for discriminative dictionary learning on vein patterns. To do this, we formulate an optimization problem that involves the classification's objective function and the representation error. In our proposed method, we add an extra term to the LoBCoD algorithm for considering the classifier performance. In each iteration, after updating the sparse coefficients and convolutional filters, we minimize the classification error by updating the classifier's parameters according to the training samples' label information. Thus, by considering the reconstruction error and the classifier performance simultaneously, we can define a problem for dictionary learning with both discriminative and representative power. We call this algorithm Discriminative Local Block Coordinate Descent (D-LoBCoD). To demonstrate the performance of the proposed D-LoBCoD method, we conduct experiments on the hand vein image data set in comparison with the state-of-the-art classification methods. The experimental results show that our method performs better than the other classification methods.

The remaining part of this paper is organized as follows. Section 2 provides an overview of the CSC model and discusses the LoBCoD method as a base of our proposed method. In section 3, we introduce the proposed D-LoBCoD method. Experimental results are reported in section 4, and the conclusions are given in the last section.

II. Convolutional Sparse Coding

A. Overview

The CSC model assumes that a signal $Y \in \mathbb{R}^M$ is approximated by the sum of a set of filters convolved with the corresponding sparse codes. In other words, the signal $Y \in \mathbb{R}^M$ can be decomposed as $Y = \sum_{i=1}^{n} d_i * Z_i$, where $d_i \in \mathbb{R}^m$ are filters that are convolved with their corresponding feature maps (as an alternative sparse representation) $Z_i \in \mathbb{R}^M$, which in general $m \ll M$. So, in the CSC model, the following minimization is performed over both the filters and the feature maps:

$$\min_{d_i, Z_i} \frac{1}{2} \|Y - \sum_{i=1}^{n} d_i * Z_i\|_2^2 + \lambda \sum_{i=1}^{n} \|Z_i\|_1. \quad (1)$$

Given the filters, this problem becomes the CSC pursuit task for finding $\{Z_i\}_{i=1}^{n}$. As shown in Fig. 1, we can write Y = $\mathbf{D}Z$ as a matrix form, where $\mathbf{D} \in \mathbb{R}^{M \times M_n}$ is a banded convolutional dictionary made from shifted versions of a local dictionary $\mathbf{D_L} \in \mathbb{R}^{m \times n}$. The local dictionary $\mathbf{D_L}$ includes the filters $\{d_i\}_{i=1}^{n}$ as its atoms, and the global sparse vector Z is obtained by interlacing the $\{Z_i\}_{i=1}^{n}$.

Therefore, the convolutional dictionary learning problem 1 can be rewritten as

$$\min_{\mathbf{D}, Z} \frac{1}{2} \|Y - \mathbf{D}Z\|_2^2 + \lambda \|Z\|_1. \quad (2)$$

According to Fig. 1, the global sparse vector Z can be broken into M non-overlapping n-dimensional local vectors z_i, which are called needles. So, one can express the global signal Y as$Y = \sum_{i=1}^{M} \mathbf{P}_i^T \mathbf{D}_L z_i$, where $\mathbf{P}_i \in \mathbb{R}^{m \times M}$ is the operator that extracts the ith n-dimensional patch from Y. In other words, $\mathbf{P}_i^T \in \mathbb{R}^{M \times m}$ is the operator that puts $\mathbf{D}_L z_i$ in the ith location and fills the rest of the entries with zeros. On the other hand, a patch $\mathbf{P}_i Y = \mathbf{P}_i \mathbf{D} Z$ taken from the signal Y equals $\mathbf{\Omega}\gamma_i$, where $\mathbf{\Omega} \in \mathbb{R}^{m \times (2m-1)n}$ is a stripe dictionary containing $\mathbf{D}_L$ in its center and $\gamma_i \in \mathbb{R}^{(2m-1)n}$ is the stripe vector containing the local vector z_i in its center [17]. In other words, a stripe vector γ_i is the sparse vector that codes all the content in $\mathbf{P}_i Y$, whiles a needle z_i only codes part of the information within it. Papyan et al. [21] showed that if all the stripes γ_i are sparse, the solution of the convolutional sparse pursuit problem is unique and can be recovered by the Orthogonal Matching Pursuit (OMP) [37] or Basis Pursuit (BP) [38].

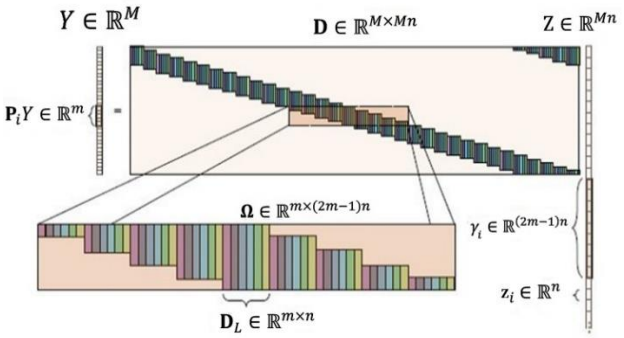

Fig. 1. The CSC model and its components [17]

B. Local Block Coordinate Descent [17]

Using the previous definitions and the separability of the l_1-norm, Zisselman et al. [17] expressed the global CSC problem (2) in terms of the local sparse vectors z_i and the local dictionary $\mathbf{D}_L$ by

$$\min_{\mathbf{D}_L, \{z_i\}} \frac{1}{2} \|Y - \sum_{i=1}^{M} \mathbf{P}_i^T \mathbf{D}_L z_i\|_2^2 + \lambda \sum_{i=1}^{n} \|z_i\|_1. \quad (3)$$

To solve the above problem, rather than optimizing all the needles together, they consider each needle z_i as a block of coordinates, taken from the global vector Z and optimize concerning each such block singly and sequentially. So, when $\mathbf{D}_L$ is known, the update rule of each needle can be written as

$$\min_{z_i} \frac{1}{2} \left\| Y - \sum_{\substack{i=1 \\ i \neq j}}^{M} \mathbf{P}_j^T \mathbf{D}_L z_j - \mathbf{P}_i^T \mathbf{D}_L z_i \right\|_2^2 + \lambda \|z_i\|_1. \quad (4)$$

By defining $R_i = Y - \sum_{\substack{i=1 \\ i \neq j}}^{M} \mathbf{P}_j^T \mathbf{D}_L z_j$ as the residual without the contribution of the needle z_i, and since the minimization involves global variables such as R_i, as it has shown in [17], one can decompose into the following local problem: the above equation can be rewritten as

$$\min_{z_i} \frac{1}{2} \|\mathbf{P}_i R_i - \mathbf{D}_L z_i\|_2^2 + \lambda \|z_i\|_1. \quad (5)$$

The reason for that is, based on the observation, the update rule of the needle z_i is effected only by pixels belonging to the corresponding patch $\mathbf{P}_i R_i$. The block coordinate descent algorithm's main idea is that every step minimizes the entire penalty w.r.t. a given block of coordinates, while the others are in their most updated values. So, every local pursuit stage (5) proceeds by updating $\hat{Y}$ and $R = Y - \hat{Y}$, as the reconstructed signal and the global residual, respectively. It is a preprocessing stage for the subsequent stage that updates the next needle based on the most updated values of the prior needles. Moreover, in this method, the needles that have no overlap in the image can be updated in parallel. So, one can use the efficient batch implementations of the LARS algorithm. To formalize these, the layer L_i is defined as the set of needles that have no induced overlap in the image. The algorithm sweeps through these layers and updates their respective needles in parallel, followed by updating the $\hat{Y}$ and R. Note that in this way, the number of the inner iterations only depends on the patch size instead of the number of needles (M). So, for $\sqrt{m} \times \sqrt{m}$ patches, the number of layers is m.

For the CSC dictionary learning or learning the CSC filters, the usual strategy is to alternate between sparse-coding and dictionary update steps for a predetermined number of iterations. The purpose of the dictionary update step is to find the minimum of the quadratic term of (3) subject to normalization of the dictionary columns:

$$\min_{\mathbf{D}_L} \frac{1}{2} \|Y - \sum_{i=1}^{M} \mathbf{P}_i^T \mathbf{D}_L z_i\|_2^2 \quad s.t. \ \{\|d_i\|_2 = 1\}_{i=1}^{n}. \quad (6)$$

The authors in [17] proposed an online dictionary updating method in a stochastic manner. In this approach, rather than concluding the entire pursuit stage and then advancing in the direction of the global gradient, they take a small step size η and update the dictionary after finding the sparse representation of only a small group of needles. Note that after every dictionary update, the filters should be normalized. More details can be found in [17].

III. Proposed Method

This section will give our proposed Discriminative-LoBCoD (D-LoBCoD) method. As we said, (3) has been found to work well in applications such as image inpainting and image fusion. However, since the objective function in (3) considers only the reconstruction error and the sparsity of the feature maps, the learned dictionary is not optimized for a classification task. In other words, the learned dictionary may not have the best discriminative power despite its representational power.

Since the dictionary D has no correspondence with the training image labels, we used a classifier to make the dictionary optimal for classification. The classifier $H = \mathbf{W}Z + \alpha$ can be replaced by the following problem:

$$\min_{\mathbf{W},\alpha} \|H - \mathbf{W}Z - \alpha\|_2^2 + \beta \|\mathbf{W}\|_2^2, \quad (7)$$

where $H = [h_1, h_1, \dots, h_C]$ are the labels of the training samples, $\mathbf{W}$ is the parameter of the classifier, Z is the global sparse vector, α is the coefficient, and C is the number of all dataset classes. When the samples are symmetric, and the input is centralized, α can be considered zero [39]. We have a discriminative problem by adding the above problem into (2) as the following problem:

$$\min_{\mathbf{D},Z,\mathbf{W}} \frac{1}{2} \|Y - \mathbf{D}Z\|_2^2 + \lambda \|H - \mathbf{W}Z\|_2^2 + \gamma \|Z\|_1 + \beta \|\mathbf{W}\|_2^2, \quad (8)$$

Where λ, γ, and β are scalers controlling the relative contribution of the corresponding terms. As the matrix $\mathbf{W}$ is always normalized column-wise in our method, we can further drop the regularization penalty term $\|\mathbf{W}\|_2^2$, and so the final formulation of the problem can be written as:

$$\min_{\mathbf{D},Z,\mathbf{W}} \frac{1}{2} \|Y - \mathbf{D}Z\|_2^2 + \lambda \|H - \mathbf{W}Z\|_2^2 + \gamma \|Z\|_1. \quad (9)$$

When $\mathbf{D}$, $\mathbf{W}$, and Z are fixed alternatively, optimization of $\mathbf{D}$, $\mathbf{W}$, and Z can be conducted. In the next subsection, we explain the optimization procedure for our proposed method.

C. Optimization

The previous global CSC problem (9) can be expressed in terms of the local sparse vectors z_k and the local dictionary $\mathbf{D}_L$ and the local matrix $\mathbf{W}_L$ by

$$\min_{\mathbf{D}_L,\mathbf{W}_L,\{z_k\}} \frac{1}{2} \left\| Y - \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{D}_L z_k \right\|_2^2 + \lambda \left\| H - \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{D}_L z_k \right\|_2^2 + \gamma \|z_k\|_1. \quad (10)$$

We can optimize $\mathbf{D}_L$, $\mathbf{W}_L$, and z_k alternatively via updating $\mathbf{D}_L$ and $\mathbf{W}_L$ by fixing z_k, and updating z_k by fixing $\mathbf{D}_L$ and $\mathbf{W}_L$. First, suppose that $\mathbf{D}_L$ and $\mathbf{W}_L$ are fixed. Like the LoBCoD method, rather than optimizing all the needles together, we can treat each needle z_k as a block of coordinates, taken from the global vector Z, and optimize each such block separately and sequentially. Thus, the update rule of each needle for the jth sample of the ith class can be written as

$$\min_{z_{i,j}^k} \frac{1}{2} \left\| (y_{i,j} - \sum_{\substack{s=1 \\ s \neq k}}^{M} \mathbf{P}_s^T \mathbf{D}_L z_{i,j}^s) - \mathbf{P}_k^T \mathbf{D}_L z_{i,j}^k \right\|_2^2 + \lambda \left\| (h_{i,j} - \sum_{\substack{s=1 \\ s \neq k}}^{M} \mathbf{P}_s^T \mathbf{W}_L z_{i,j}^s) - \mathbf{P}_k^T \mathbf{W}_L z_{i,j}^k \right\|_2^2 + \gamma \left\| z_{i,j}^k \right\|_1. \quad (11)$$

where $y_{i,j}$ is the jth sample of class i, and also $h_{i,j}$ is a label

matrix with the same size of $y_{i,j}$ which all entries are zero except *i*th column. In other words, the class number of the training sample indicates the position of the non-zero column in $h_{i,j}$. For example, if the training sample is the fourth sample from the second class (i.e., $y_{2,4}$), just the second column entries in $h_{2,4}$ are non-zero, and all other entries in $h_{2,4}$ will be zero. Like before, By defining $R_k = (y_{i,j} - \sum_{\substack{s=1 \\ s\neq k}}^{M} \mathbf{P}_s^T \mathbf{D}_L z_{i,j}^s)$ and $F_k = (h_{i,j} - \sum_{\substack{s=1 \\ s\neq k}}^{M} \mathbf{P}_s^T \mathbf{D}_L z_{i,j}^s)$ as the residuals without the contribution of the needle $z_{i,j}^k$, we can write (11) as

$$\min_{z_{i,j}^k} \tfrac{1}{2}\left\|\mathbf{P}_k R_k - \mathbf{D}_L z_{i,j}^k\right\|_2^2 + \lambda\left\|\mathbf{P}_k F_k - \mathbf{W}_L z_{i,j}^k\right\|_2^2 + \gamma\left\|z_{i,j}^k\right\|_1. \quad (12)$$

Thus, we can update every needle $z_{i,j}^k$ belongs to the *j*th sample of the *i*th class by the LoBCoD method. Now, we discuss updating $\mathbf{D}_L$ and $\mathbf{W}_L$ when $z_{i,j}^k$ is fixed. The goal of the dictionary update step is to find the minimum of the term of (10) subject to the constraint of normalized dictionary columns:

$$\min_{\mathbf{D}_L} \tfrac{1}{2}\left\|y_{i,j} - \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{D}_L z_{i,j}^k\right\|_2^2 \quad s.t.\ \{\|d_i\|_2 = 1\}_{i=1}^n. \quad (13)$$

One can do so in a batch manner that requires access to the entire data set at every iteration. We can use the projected steepest descent to solve (13) with this method. To do this, we perform the steepest descent with a small step size and project the solution to the constraint set after each iteration until convergence. To that end, the gradient of the quadratic term in (13) w.r.t. $\mathbf{D}_L$ is:

$$\nabla_{\mathbf{D}_L} = -\sum_{k=1}^{M} \mathbf{P}_k \left(y_{i,j} - \hat{y}_{i,j}\right) z_{i,j}^k. \quad (14)$$

Thus, we can update the local dictionary $\mathbf{D}_L$ by advancing in the direction of the above gradient while normalizing the columns of the resulting $\mathbf{D}_L$ in each iteration until convergence. But that is very slow because each dictionary update can be performed only after finishing the entire sparse coding stage. Therefore, we have used the Stochastic-LoBCoD approach [17] to update the local dictionary $\mathbf{D}_L$ matrix.

In this approach, we can update the local dictionary in a stochastic manner. To explain that, instead of concluding the whole pursuit stage and then advancing in the direction of the global gradient, we can take a small step size η and update the dictionary after finding the sparse representation of only a small group of needles [17].

According to this, in every iteration, a group of needles is updated, which is referred to as a layer L_i. Therefore, the algorithm is converged faster than the batch method. Note that the filters (atoms) should be normalized after every dictionary update. Thus, we can write the following formulation for updating the local dictionary:

$$\mathbf{D}_L = \mathbb{P}\left[\mathbf{D}_L - \eta \nabla_{\mathbf{D}_L}\right], \quad (15)$$

where $\mathbb{P}[.]$ is the operator that normalizes the dictionary atoms.

As we said, like local dictionary updates, we have used the Stochastic-LoBCoD method for updating the $\mathbf{W}_L$ matrix. Thus, we have the following formulation to find the minimum of the term of (10) subject to the constraint normalized matrix columns:

$$\min_{\mathbf{W}_L} \tfrac{1}{2}\left\|h_{i,j} - \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{W}_L z_{i,j}^k\right\|_2^2 \quad s.t.\ \{\|w_i\|_2 = 1\}_{i=1}^n. \quad (16)$$

As we said before, in the above equation, $h_{i,j}$ is a label matrix with the same size as the training sample matrix $y_{i,j}$ all of the entries are zero except the *i*th column, where the position of the non-zero column indicates the training sample class. For example, if the training sample belongs to class 1, all entries of $h_{i,j}$ (in fact $h_{1,j}$) are zero except column 1. Thus, $\mathbf{W}_L$ will be updated in the direction of classification error minimization and $\left\|h_{i,j} - \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{W}_L z_{i,j}^k\right\|_2^2$ is the classification error. Please note that since the update process of $\mathbf{D}_L$ and $\mathbf{W}_L$ are independent, we can perform that simultaneously in our algorithm. The final algorithm that incorporates needles, $\mathbf{D}_L$, and $\mathbf{W}_L$ update is summarized in Algorithm 1.

D. Classification

After obtaining the local learned dictionary $\mathbf{D}_L$ and the corresponding classifier $\mathbf{W}_L$, we want to specify the query sample class. To do this, first, we find the sparse coefficients for a query sample y_q by (12). We name it Z_q. After that, we obtain the class number of the query sample y_q by

$$Cn = \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{W}_L z_q^k, \quad (17)$$

where z_q^k are needles of Z_q. In other words, $Z_q = [z_q^1, z_q^2, \dots, z_q^M]$ consists of *M* needles for reconstructing query sample y_q. The class number of the query sample y_q is the index corresponding to the largest column of Cn. The related algorithm is summarized in Algorithm 3.

IV. EXPERIMENTS

In this section, we compare our D-LoBCoD method with state-of-the-art methods such as conventional SRC [5], Mutual SRC (MSRC) [2], weighted discriminative collaborative competitive representation (WDCCR) [8], label consistent KSVD (LC-KSVD) [6], Fisher Discrimination Dictionary Learning (FDDL) [7], Discriminative fisher embedding dictionary learning (DFEDL) [9], Low-rank Shared Dictionary Learning (LRSDL) [10]; and convolutional sparse coding classification (CSCC) [35]. We conducted our evaluations on a dorsal hand vein database [40], provided by Prof. Sankur from the Bogazici University, Nozaripour/Sankur, personal communication. The database includes 1200 left-hand infrared images of 100 subjects. These images were captured in four situations, 1) normal, 2) after squeezing an elastic ball repetitively, 3) after carrying a 3kg bag, and 4) after holding an ice pack on the back of the hand for one minute. Fig. 2 shows some sample images of this database.

Algorithm 1: The discriminative-LoBCoD algorithm (learning)

Input: training samples Y, label matrices H, initial dictionary $\mathbf{D}_L$, initial matrix $\mathbf{W}_L$, initial needles $\{z_{i,j}^{k,0}\}_{k=1}^{M}$, parameters λ, γ, and η

Output: needles $\{z_{i,j}^{k}\}_{k=1}^{M}$ for all training samples, the dictionary $\mathbf{D}_L$, classifier matrix $\mathbf{W}_L$

Initialization: $R_{i,j} = y_{i,j}, F_{i,j} = h_{i,j}, \hat{y}_{i,j} = \hat{h}_{i,j} = 0, s = 0$

for iteration=1 **to** T **do**
  s=s+1
  for i=1 **to** C **do**
    for j=1 **to** n_i **do**
      for t=1 **to** m, **do**
        Computation of the residual:
        $R_{i,j}^{t} = R_{i,j} + \sum_{k\in L_t} \mathbf{P}_k^T \mathbf{D}_L \, z_{i,j}^{s-1}$
        $F_{i,j}^{t} = F_{i,j} + \sum_{k\in L_t} \mathbf{P}_k^T \mathbf{W}_L \, z_{i,j}^{s-1}$

        Sparse pursuit: $\forall k \in L_t$(in parallel):
        $$z_{i,j}^{k,s} = \underset{z_{i,j}^{k}}{\operatorname{argmin}} \frac{1}{2}\left\|\mathbf{P}_k R_{i,j}^{t} - \mathbf{D}_L z_{i,j}^{k}\right\|_2^2 + \lambda\left\|\mathbf{P}_k F_{i,j}^{t} - \mathbf{W}_L z_{i,j}^{k}\right\|_2^2 + \gamma\left\|z_{i,j}^{k}\right\|_1$$

        Computation of the reconstructed signal and reconstructed label matrix:
        $\hat{y}_{i,j} = \hat{y}_{i,j} + \sum_{k\in L_t} \mathbf{P}_k^T \mathbf{D}_L \, (z_{i,j}^{k,s} - z_{i,j}^{k,s-1})$
        $\hat{h}_{i,j} = \hat{h}_{i,j} + \sum_{k\in L_t} \mathbf{P}_k^T \mathbf{W}_L \, (z_{i,j}^{k,s} - z_{i,j}^{k,s-1})$

        Computation of the residual signal and residual label matrix:
        $R_{i,j} = y_{i,j} - \hat{y}_{i,j}$
        $F_{i,j} = h_{i,j} - \hat{h}_{i,j}$

        Computation of the gradient w.r.t $\mathbf{D}_L$ and $\mathbf{W}_L$:
        $\nabla_{\mathbf{D}_L} = \sum_{k\in L_t} \mathbf{P}_k R_{i,j} \, (z_{i,j}^{k,s})^T$
        $\nabla_{\mathbf{W}_L} = \sum_{k\in L_t} \mathbf{P}_k F_{i,j} \, (z_{i,j}^{k,s})^T$

        Dictionary and classifier matrix update and normalization:
        $\mathbf{D}_L = \mathbb{P}[\mathbf{D}_L - \eta \nabla_{\mathbf{D}_L}]$
        $\mathbf{W}_L = \mathbb{P}[\mathbf{W}_L - \eta \nabla_{\mathbf{W}_L}]$
      end for
    end for
  end for
end for

Algorithm 2: The D-LoBCoD algorithm (classification)

Input: the trained dictionary $\mathbf{D}_L$, the trained classifier matrix $\mathbf{W}_L$, initial needles $\{z_q^{k,0}\}_{k=1}^{M}$, query sample y_p, and the parameter γ

Output: the class of the query sample y_p

Initialization: $R_q = y_q, \ \hat{y}_q = 0, s = 0$

for iteration=1 **to** T **do**
  s=s+1
  for t=1 **to** m **do**
    Computation of the residual:
    $R_q^{t} = R_q + \sum_{k\in L_t} \mathbf{P}_k^T \mathbf{D}_L \, z_q^{s-1}$

    Sparse pursuit: $\forall k \in L_t$(in parallel):
    $$z_q^{k,s} = \underset{z_q^{k}}{\operatorname{argmin}} \frac{1}{2}\left\|\mathbf{P}_k R_q^{t} - \mathbf{D}_L z_q^{k}\right\|_2^2 + \gamma\left\|z_q^{k}\right\|_1$$

    Computation of the reconstructed signal and reconstructed label matrix:
    $\hat{y}_q = \hat{y}_q + \sum_{k\in L_t} \mathbf{P}_k^T \mathbf{D}_L \, (z_q^{k,s} - z_q^{k,s-1})$

    Computation of the residual signal:
    $R_q = y_q - \hat{y}_q$
  end for
end for
$Cn = \sum_{k=1}^{M} \mathbf{P}_k^T \mathbf{W}_L z_q^{k}$

A. ROI and Vein Extraction

As mentioned before, selecting a similar ROI in all images is one of the most important concerns for an identification system. In this work, for limiting the effects of relocation and hand spin while imaging acquisition, a robust method against wrist spin named Floating ROI (FROI), is used, which is proposed in [4]. In this method, the lengths and angles of sides in the ROI quadrilateral will be changeable based on the amount of wrist spin. This is because when we have a wrist spin, one side of the hand will be compressed, and the other will be extended. Therefore, the ROI length on the compressed side should be less than on the extended side. Also, the angles of two sides, toward the upper side, should be changed relative to the amount of the hand spin. Thus, a similar area with the maximum vein pattern will be obtained by changing the two upper angles of FROI and the lateral side's length [4, 41]. Fig. 3 shows the FROI extracted from a hand in three states of wrist spin.

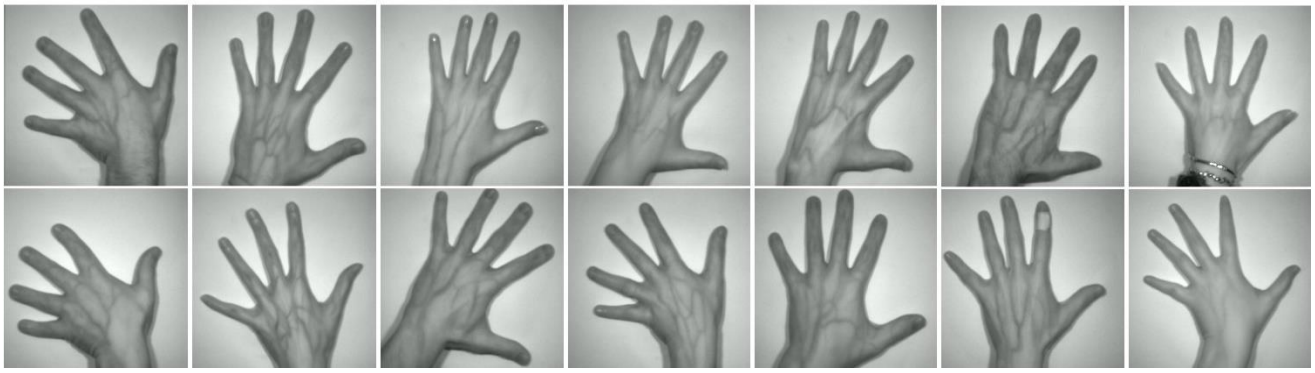

Fig. 2. Some image samples of the Bogazici hand vein database

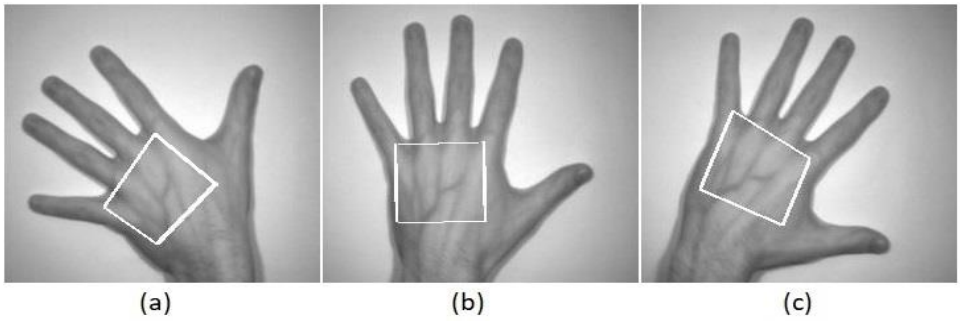

Fig. 3. FROI was extracted for a hand in three states of wrist spin. (a) Wrist spin to the left. (b) Normal. (c) Wrist spin to the right

As it can be seen, in this approach, unlike the conventional methods [42], in which the shape of ROI is a fixed quadrilateral for all wrist spins, the final form of ROI is not always square or rectangular. After selecting and extracting FROI, like [4], a mask designed in Fig. 4 with relation (18) is used for extracting the vein pattern.

$$\begin{cases} \boldsymbol{P} = \frac{1}{25}\sum_{i=1}^{25} P_i \\ \boldsymbol{Q} = \frac{1}{56}\sum_{j=1}^{56} Q_j \end{cases}. \quad (15)$$

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9
Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18
Q19	Q20	P1	P2	P3	P4	P5	Q21	Q22
Q23	Q24	P6	P7	P8	P9	P10	Q25	Q26
Q27	Q28	P11	P12	P13	P14	P15	Q28	Q30
Q31	Q32	P16	P17	P18	P19	P20	Q33	Q34
Q35	Q36	P21	P22	P23	P24	P25	Q37	Q38
Q39	Q40	Q41	Q42	Q43	Q44	Q45	Q46	Q47
Q48	Q49	Q50	Q51	Q52	Q53	Q54	Q55	Q56

Fig. 4: The designed mask for extracting the vein pattern from the FROI

The mask moves over each pixel of the extracted FROI. If $\boldsymbol{P} > \boldsymbol{Q}$, the pixel of FROI located under the central cell of the mask (P13) is recognized as a vein; otherwise, it is considered the background pixel. Fig. 5 indicates the vein patterns extracted for this filter's three images in Fig. 3. As it can be seen, the veins extracted by this method are much similar, despite a large wrist spin difference in the three images.

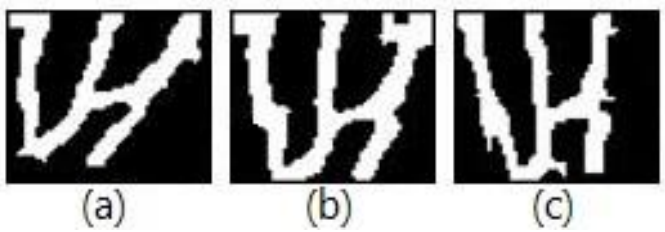

Fig. 5: The designed mask extracted vein patterns for three images of Fig. 3

B. Comparison

After extracting the vein pattern for all images and converting them to the same size images (in this work, 64*80 pixels), as mentioned in the previous section, we used the D-LoBCoD for vein pattern Classification. We randomly divided the samples into the training and testing sets five times to highlight the proposed method's advantages. The number Ts indicates the training samples per class; the rest are the testing samples. The results of recognition accuracies for each method are achieved for ten runs, based on random splits of the training and testing group. Then the average experimental results of each method for every Ts are considered the average recognition rates. Moreover, in all experiments, we have used the parameter values that the authors suggested in the original papers for a fair comparison. For instance, we have chosen $\alpha = \beta = 10^{-5}$ and $\gamma = 10^{-2}$ in the DFEDL method. Besides, for the compared FDDL, LRSDL, and DFEDL algorithms, the classification parameter μ is set to 0.5, and the classification parameter θ is selected from the set $\{10^{-3}, 10^{-2}, 10^{-1}\}$. Therefore, the best results of all the methods are reported. It should be not that the toolbox provided in [10] is used to produce the results of SRC, FDDL, LC-KSVD, and LRSDL. Moreover, we reimplement the codes of MSRC, WDCCR, DFEDL, and CSCC provided by the corresponding authors, and the best parameters are searched such that the experimental results are valid. As suggested in the original papers, the atom number of each class for FDDL, LRSDL, and DFEDL algorithms are the same as the number of training samples per class. For SRC, CRC, MSRC, WDCCR, and LC-KSVD, the atom number of the dictionary are the same; the number of training samples per class is multiplied by the number of classes on the database. Also, for the CSCC algorithm and our proposed method, we used 81 filters with 64 (8×8) dimensions. A platform with an Intel Core i7-7500U CPU and 8.0 GB RAM was used to run the experimental analysis by Matlab 2020b software.

In our experiments on the Bogazici database, the values of Ts are 4 to 8 with step 1. The achieved average recognition rates from ten runs for all competitive methods are reported in Table 1. Besides, as an example, the trained local dictionary ($\mathbf{D}_L$) and the classifier ($\mathbf{W}_L$) obtained from our proposed method on this database at Ts = 6 are shown in Fig. 6.

TABLE I
Each Method's Recognition Rates (Mean ± Std %) In Different Ts On The Bogazici Hand Vein Database

Method	Number of Ts (training sample)				
	4	5	6	7	8
SRC [5]	48.13 ± 1.13	55.62 ± 1.31	64.91 ± 1.62	73.05 ± 1.12	81.72 ± 1.47
MSRC [2]	49.12 ± 1.47	57.14 ± 1.22	64.32 ± 1.61	73.81 ± 1.52	82.33 ± 1.24
WDCCR [8]	50.01 ± 1.32	58.17 ± 1.42	64.69 ± 1.44	75.43 ± 1.39	83.61 ± 1.43
LC-KSVD [6]	53.62 ± 1.43	60.15 ± 1.13	66.61 ± 1.48	72.01 ± 1.24	80.17 ± 1.22
FDDL [7]	55.03 ± 1.37	61.26 ± 1.34	67.61 ± 1.27	74.65 ± 1.34	82.43 ± 1.44
DFEDL [9]	63.14 ± 1.13	68.47 ± 1.22	73.78 ± 1.06	79.72 ± 1.39	86.85 ± 1.32
LRSDL [10]	59.15 ± 1.52	65.31 ± 1.45	71.58 ± 1.28	79.08 ± 1.51	85.15 ± 1.07
CSCC [35]	66.19 ± 1.09	72.61 ± 1.45	76.83 ± 1.24	83.14 ± 1.47	87.19 ± 1.24
D-LoBCoD	**71.34 ± 1.42**	**75.19 ± 1.22**	**81.34 ± 1.51**	**86.21 ± 1.08**	**89.43 ± 1.12**

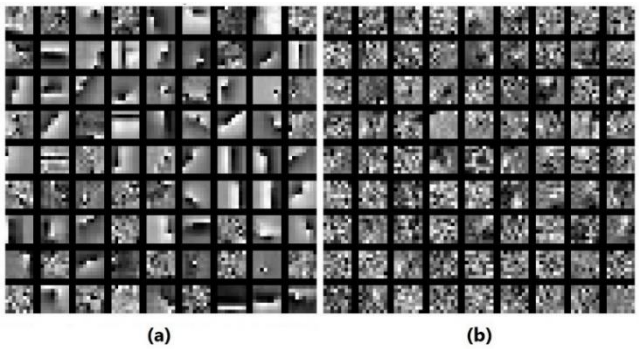

Fig. 6: (a) The trained local dictionary $\mathbf{D}_L$ and (b) the classifiers $\mathbf{W}_L$ which is obtained from our proposed method at the middle Ts (Ts=6) for the Bogazici database.

C. Analysis of the Experimental Results

The experimental results on the Bogazici database were shown in the previous subsection. We further discuss them in this subsection. As can be seen from Table 1, the average recognition rates of each method increase with adding the number of training samples for each class on all the databases. We can observe that the proposed D-LoBCoD method nearly performs better than the other methods. This experimental fact implies that by learning a shift-invariant dictionary composed of many filters for better extracting the local patterns at translated positions of the samples by convolution, as well as the simultaneous use of the label information of the training samples, we can achieve better classification results. Although the LC-KSVD algorithm uses the label information of the atoms to constrain the coding coefficients, it usually cannot preserve specific characteristics of images for each class. The class-specific dictionary learning algorithms like FDDL are better than the LC-KSVD because it uses the representation residuals and the fisher criterion of the coding coefficients. The LRSDL algorithm exploits the shared dictionary to capture the similar characteristics of training samples and the specific class dictionary to preserve the intraclass information of each class. This algorithm works better than the FDDL on a database with fewer training samples (like the Bogazici database).

On the other hand, the DFEDL algorithm exploits the interclass and intraclass information of the atoms, the interclass and intraclass characteristics of the profiles, and coding coefficients to construct the discriminative Fisher embedding terms. As can be seen, the results of this algorithm are higher than FDDL and LRSDL algorithms. By combining the convolutional sparse coding framework with the classification strategy, the CSCC algorithm has achieved better results. But as this algorithm trains the filter banks for each class separately and relies on the ADMM approach in the Fourier domain, it requires a lot of memory and computations. Also, it makes the local characterizations of the image ignored. This is while our method trains a shared dictionary and has better storage complexity than the CCSC.

Furthermore, using a localized strategy, we were also able to use the advantages of the local characterizations of the image. For the SRC, CRC, MSRC, and WDCCR algorithms, increasing the average recognition rate is more tangible when the number of training samples increases. This demonstrates that in these algorithms, the number of training samples and how to choose them to play an important role in recognition. This is less Tangible in the LC-KSVD, FDDL, LRSDL, DFEDL, CSCC, and especially our proposed method. Because in all of them, a dictionary is trained based on the training samples.

Moreover, in terms of time comparison, as the computational cost of the convolutional sparse coding method is more than the sparse coding method, the training phase time of our method is longer than other sparse-based methods (except the methods of [2,5, and 8] that do not have a training phase). But the time of the training phase in our method is shorter than [35] of a convolutional-based method. In fact, among all competing methods, the method [9] is the fastest in the training phase, but in the testing phase, the method [6] is faster than other methods.

V. Conclusion

In this paper, we proposed a novel D-LoBCoD algorithm that combines a CSC framework with a classification strategy for vein recognition. Because in sparse coding, each data sample is represented as a linear combination of a few atoms, it cannot capture shifted local patterns common in image samples. Thus, by using a CSC model, local patterns at translated positions of the samples can be extracted. We have used the local block coordinate descent algorithm to pursue the global CSC model while operating locally on image patches. Moreover, we have improved memory

requirements and computational time by using additional parameters and parallel computations. Finally, by using this method and embedding a classifier in it and using an effective method in ROI extraction, we could show the superiorities of our proposed method compared with some state-of-the-art algorithms.

REFERENCES

[1] J. Wang, Z. Pan, G. Wang, M. Li, and Y. Li, "Spatial pyramid pooling of selective convolutional features for vein recognition," *IEEE Access,* vol. 6, pp. 28563-28572, 2018.

[2] S. Shazeeda and B. A. Rosdi, "Finger vein recognition using mutual sparse representation classification," *IET Biometrics*, vol. 8, no. 1, pp. 49-58, 2019.

[3] H. Shaaban, "Enhanced Region of Interest Extraction method for Finger Vein Images," *Artificial Intelligence & Robotics Development Journal*, pp. 13-25, 2021.

[4] A. N. Pour, E. Eslami, and J. Haddadnia, "A new method for automatic extraction of region of interest from infrared images of dorsal hand vein pattern based on floating selection model," *International Journal of Applied Pattern Recognition,* vol. 2, no. 2, pp. 111-127, 2015.

[5] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE transactions on pattern analysis and machine intelligence,* vol. 31, no. 2, pp. 210-227, 2008.

[6] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent K-SVD: Learning a discriminative dictionary for recognition," *IEEE transactions on pattern analysis and machine intelligence,* vol. 35, no. 11, pp. 2651-2664, 2013.

[7] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Sparse representation based fisher discrimination dictionary learning for image classification," *International Journal of Computer Vision,* vol. 109, no. 3, pp. 209-232, 2014.

[8] J. Gou, L. Wang, Z. Yi, Y. Yuan, W. Ou, and Q. Mao, "Weighted discriminative collaborative, competitive representation for robust image classification," *Neural Networks,* vol. 125, pp. 104-120, 2020.

[9] Z. Li, Z. Zhang, J. Qin, Z. Zhang, and L. Shao, "Discriminative fisher embedding dictionary learning algorithm for object recognition," *IEEE transactions on neural networks and learning systems,* vol. 31, no. 3, pp. 786-800, 2019.

[10] T. H. Vu and V. Monga, "Fast low-rank shared dictionary learning for image classification," *IEEE Transactions on Image Processing,* vol. 26, no. 11, pp. 5160-5175, 2017.

[11] N. Zhou, H. Jiang, L. Gong, and X. Xie, "Double-image compression and encryption algorithm based on co-sparse representation and random pixel exchanging," *Optics and Lasers in Engineering,* vol. 110, pp. 72-79, 2018.

[12] H. Pan, Z. Jing, L. Qiao, and M. Li, "Discriminative structured dictionary learning on Grassmann manifolds and its application on image restoration," *IEEE transactions on cybernetics,* vol. 48, no. 10, pp. 2875-2886, 2017.

[13] E. Miandji, S. Hajisharif, and J. Unger, "A unified framework for compression and compressed sensing of light fields and light field videos," *ACM Transactions on Graphics (TOG),* vol. 38, no. 3, pp. 1-18, 2019.

[14] G. Ma, T.-Z. Huang, J. Huang, and C.-C. Zheng, "Local low-rank and sparse representation for hyperspectral image denoising," *IEEE Access,* vol. 7, pp. 79850-79865, 2019.

[15] Z. Ding, M. Shao, and Y. Fu, "Deep, robust encoder through locality preserving low-rank dictionary," in *European Conference on Computer Vision,* 2016: Springer, pp. 567-582.

[16] Y. Xu, Z. Li, J. Yang, and D. Zhang, "A survey of dictionary learning algorithms for face recognition," *IEEE Access,* vol. 5, pp. 8502-8514, 2017.

[17] E. Zisselman, J. Sulam, and M. Elad, "A local block coordinate descent algorithm for the CSC model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8208-8217.

[18] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Scalable online convolutional sparse coding," *IEEE Transactions on Image Processing,* vol. 27, no. 10, pp. 4850-4859, 2018.

[19] Y. Romano and M. Elad, "Patch-disagreement as a way to improve K-SVD denoising," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015: IEEE, pp. 1280-1284.

[20] P. Rodríguez, "FAST CONVOLUTIONAL SPARSE CODING WITH ℓ 0 PENALTY," in *2018 IEEE XXV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 2018: IEEE, pp. 1-4.

[21] V. Papyan, Y. Romano, J. Sulam, and M. Elad, "Convolutional dictionary learning via local processing," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5296-5304.

[22] J. He, L. Yu, Z. Liu, and W. Yang, "Image super-resolution by learning weighted convolutional sparse coding," *Signal, Image and Video Processing*, pp. 1-9, 2021.

[23] F. Heide, W. Heidrich, and G. Wetzstein, "Fast and flexible convolutional sparse coding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5135-5143.

[24] H. Chang, J. Han, C. Zhong, A. M. Snijders, and J.-H. Mao, "Unsupervised transfer learning via multi-scale convolutional sparse coding for biomedical applications," *IEEE transactions on pattern analysis and machine intelligence,* vol. 40, no. 5, pp. 1182-1194, 2017.

[25] L. Wang *et al.*, "Multimodal medical image fusion based on nonsubsampled shearlet transform and convolutional sparse representation," *Multimedia Tools and Applications*, pp. 1-21, 2021.

[26] A. Cogliati, Z. Duan, and B. Wohlberg, "Context-dependent piano music transcription with convolutional sparse coding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* vol. 24, no. 12, pp. 2218-2230, 2016.

[27] H.-W. Liao and L. Su, "Monaural source separation using ramanujan subspace dictionaries," *IEEE Signal Processing Letters,* vol. 25, no. 8, pp. 1156-1160, 2018.

[28] M. Šorel and F. Šroubek, "Fast convolutional sparse coding using matrix inversion lemma," *Digital Signal Processing,* vol. 55, pp. 44-51, 2016.

[29] B. Wohlberg, "Efficient algorithms for convolutional sparse representations," *IEEE Transactions on Image Processing,* vol. 25, no. 1, pp. 301-315, 2015.

[30] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[31] K. Degraux, U. S. Kamilov, P. T. Boufounos, and D. Liu, "Online convolutional dictionary learning for multimodal imaging," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017: IEEE, pp. 1617-1621.

[32] J. Liu, C. Garcia-Cardona, B. Wohlberg, and W. Yin, "Online convolutional dictionary learning," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017: IEEE, pp. 1707-1711.

[33] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. Cun, "Learning convolutional feature hierarchies for visual recognition," *Advances in neural information processing systems,* vol. 23, pp. 1090-1098, 2010.

[34] Y. Zhou, H. Chang, K. Barner, P. Spellman, and B. Parvin, "Classification of histology sections via multispectral convolutional sparse coding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3081-3088.

[35] B. Chen, J. Li, B. Ma, and G. Wei, "Convolutional sparse coding classification model for image classification," in *2016 IEEE international conference on image processing (ICIP)*, 2016: IEEE, pp. 1918-1922.

[36] J. Jin and C. P. Chen, "Convolutional sparse coding for face recognition," in *2017 4th International Conference on Information, Cybernetics and Computational Social Systems (ICCSS)*, 2017: IEEE, pp. 137-141.

[37] S. Chen, S. A. Billings, and W. Luo, "Orthogonal least squares methods and their application to non-linear system

identification," *International Journal of Control*, vol. 50, no. 5, pp. 1873-1896, 1989.

[38] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129-159, 2001.

[39] D.-S. Pham and S. Venkatesh, "Joint learning and dictionary construction for pattern recognition," in *2008 IEEE conference on computer vision and pattern recognition*, 2008: IEEE, pp. 1-8.

[40] A. Yuksel, L. Akarun, and B. Sankur, "Hand vein biometry based on geometry and appearance methods," *IET computer vision*, vol. 5, no. 6, pp. 398-406, 2011.

[41] A. Nozaripour and H. Soltanizadeh, "Robust Vein Recognition against Rotation Using Kernel Sparse Representation," *Journal of AI and Data Mining*, 2021.

[42] C.-L. Lin, S.-H. Wang, H.-Y. Cheng, K.-C. Fan, W.-L. Hsu, and C.-R. Lai, "Bimodal biometric verification using the fusion of palmprint and infrared palm-dorsum vein images," *Sensors*, vol. 15, no. 12, pp. 31339-31361, 2015.