Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 8817571, 16 pages
https://doi.org/10.1155/2020/8817571

Research Article

Unconstrained Face Identification Based on 3D Face Frontalization and Support Vector Guided Dictionary Learning

Zhi Zhang,[1,2,3] Xin Xu,[4] Jiuzhen Liang,[4] and Bingyu Sun[1]

[1]*HFIPS, Chinese Academy of Sciences, Hefei 230031, China*
[2]*University of Science and Technology of China, Hefei 230026, China*
[3]*Traffic Management Research Institute of the Ministry of Public Security, Wuxi 214151, China*
[4]*School of Computer Science and Artificial Intelligence, Changzhou University, No.21 Gehu Middle Road, Changzhou 213164, China*

Correspondence should be addressed to Jiuzhen Liang; jzliang@cczu.edu.cn

Received 28 August 2020; Accepted 19 October 2020; Published 3 November 2020

Academic Editor: Jerzy Baranowski

Face identification aims at putting a label on an unknown face with respect to some training set. Unconstrained face identification is a challenging problem because of the possible variations in face pose, illumination, occlusion, and facial expression. This paper presents an unconstrained face identification method based on face frontalization and learning-based data representation. Firstly, the frontal views of unconstrained face images are automatically generated by using a single, unchanged 3D face model. Then, we crop the face relevant regions of the frontal views to segment faces from the backgrounds. At last, to enhance the discriminative capability of the coding vectors, a support vector-guided dictionary learning (SVGDL) model is applied to adaptively assign different weights to different pairs of coding vectors. The performance of the proposed method FSVGDL (frontalization-based support vector guided dictionary learning) is evaluated on the Labeled Faces in the wild (LFW) database. After decision fusion, the identification accuracy yields 97.17% when using 7 images per individual for training and 3 images per individual for testing with 158 classes in total.

1. Introduction

As one of the biometric technologies, face recognition has developed rapidly in recent decades. Compared with other biometric technologies such as fingerprint and iris recognition, face recognition has clear advantages of being natural and noncontact. Face recognition includes face verification and face identification, and the objective of face verification is to predict whether an image pair is from the same person or not, and the objective of face identification is to put a label on an unknown face with respect to some training set.

During the last few decades, a lot of research studies have been carried out on face recognition and considerable progress can be seen. Despite this, unconstrained face recognition is a very challenging problem because of the possible variations in face pose, expression and illumination, etc. Traditional methods for frontal face images cannot handle with these variations; therefore, recently tremendous efforts have been put on developing robust face descriptors, face frontalization methods, and learning-based face recognition methods.

Popular face descriptors can be divided into two categories: global representing method such as Principle Component Analysis (PCA) [1], Linear discriminate Analysis (LDA) [2], hand-crafted local features such as Local Binary Pattern (LBP) [3], and Gabor wavelets [4]. However, these face descriptors are sensitive not only to pose changes but also to occlusion and expression variations.

Face frontalization is the process of synthesising a new, frontal view of a face image with varying face pose. Recent reports have suggested that an effective face frontalization method may greatly improve the performances of face recognition systems. Given a detected face and its landmarks, Fontaine et al. [5] proposed a computationally efficient 2D mesh-warp method to align it to a reference face. Morphable-based methods [6–8] attempt to estimate 3D shape and texture of faces from single image by fitting a statistical, morphable 3D face model to images, while these morphable-based methods require near-frontal of unoccluded faces. Kemelmacher-Shlizerman and Basri [9] proposed a shape from the shading method which uses harmonic representations of lighting to recover the 3D shape of the given face, and this method can produce outstanding facial details. Despite this, the method is sensitive to occlusions, and it requires segmenting the faces from the backgrounds carefully. An effective face frontalization method was proposed in [10] by using a single, unchanged 3D face model to generate the frontal view of a given face. This method is not only computationally efficient but also outstanding in preserving facial details.

Sparse representation has seen a rapid development both in theory and algorithms during the past few years. The Sparse Representation-based Classification (SRC) scheme that Wright et al. [11] proposed first applied the ideas of sparse coding to face recognition which have made great success. While in [12], Zhang et al. claimed that it is the collaborative representation but not the ℓ_1-norm sparsity that makes SRC powerful for face recognition. Although these methods have shown their ability to face recognition, directly using the whole training samples as the dictionary atoms may make them not effective enough to represent the query images. Dictionary Learning (DL) is a learning-based method that obtains a dictionary by learning from the training samples. Dictionary learning methods can be divided into two categories: supervised and unsupervised methods. One of the representatives of unsupervised DL is the K-SVD algorithm [13]; Jiang et al. [14] added label consistency constraint to the original K-SVD algorithm to enforce the discrimination of coefficient vectors. However, the dictionary that K-SVD learned is an overcomplete dictionary, which means it could only faithfully represent the training samples but is not suitable for classification. Therefore, lots of supervised dictionary methods have been proposed. Sprechmann and Sapiro [15] proposed a method that uses the sparse representation coefficients to learn a subdictionary for each class and successfully applied it to face recognition and clustering. Based on Fisher discrimination criterion, Yang et al. proposed a Fisher discrimination dictionary learning (FDDL) algorithm in [16]. This class-specific dictionary learning method learns a subdictionary for 60 each class, and fisher discrimination criterion is also performed on the coefficient vectors. However, this method needs sufficient training samples for each class to achieve good performance, and so it is not suitable for small training sample problems. Recently, a parametrization method (Support Vector-Guided Dictionary Learning, SVGDL) which can adaptively determine the weight of each coding vector pair was proposed in [17]. This method only imposes constrains on the coefficients and has achieved good performance on face identification.

In this paper, we focus on the challenging problem of unconstrained face identification. Our contributions can be summarized as follows:

(1) We propose a frontalization-based support vector guided dictionary learning (FSVGDL) method to cope with unconstrained face identification. On one hand, face frontalization can reduce the impact of pose variations by automatic synthesis, a new, frontal view of a face image. The process of frontalization also helps segment the faces from their backgrounds since it synthesizes the frontal views in the center of the image. The removal of backgrounds can simplify unconstrained face identification. To address other variations in unconstrained images, we learn a support vector-guided dictionary which can adaptively assign different weights to different pairs of coding vectors for face representation. Therefore, the coding vectors have strong discriminative capability to represent face images well.

(2) Decision fusion generally leads to performance gain; therefore, we perform fusion on several decisions. We use voting to do the fusion, despite its simplicity, and the experiment results show the effectiveness of decision fusion.

The rest of the paper is organized as follows. In Section 2, we provide a brief view of the related works. The detailed process of the proposed FSVGDL (frontalization-based support vector-guided dictionary learning) method is introduced in Section 3. The experimental results on LFW (Labeled Faces in the wild) database are presented in Section 4. Finally, Section 5 concludes this paper.

2. Related Works

2.1. Whitened Principal Component Analysis. Whitened Principal Component Analysis is a method that performs whitening after Principal Component Analysis. Principal Component Analysis (PCA) is a method that we commonly used to reduce the noise in feature vectors; whitening can make the feature vectors more discriminative. PCA computes the d eigenvectors with the largest eigenvalues of the covariance matrix. The covariance matrix of PCA is defined by

$$\Sigma_{\mathrm{PCA}} = \sum_{i=1}^{n} (t_i - m)(t_i - m)^T, \tag{1}$$

where t_l denotes an image feature vector, m is the mean of the image feature vectors, $\wedge = \{\lambda_1, \ldots, \lambda_d\}$ denotes the d eigenvalues, and $\vee = (v_1, \ldots, v_d)$ denotes the corresponding d eigenvectors of Σ_{PCA}. The whitening process is denoted by

$$\hat{t} = \operatorname{diag}\left(\lambda_1^{-1/2}, \ldots, \lambda_d^{-1/2}\right) \vee^T t, \tag{2}$$

where $\hat{t}$ denotes the feature vector after whitening PCA. We can see from equation (2) that the features are weighted by the inverse of the eigenvalues. Through this method, we can penalize the eigenvectors with large eigenvalues and therefore make the input feature vectors less redundant.

2.2. General Discriminative Dictionary Learning. Suppose there are K classes of subjects $T = [T_1, T_2, \ldots, T_K]$ and $T_i \in \mathfrak{R}^{m \times n_i} Ti = [\hat{t}_1, \hat{t}_2, \ldots, \hat{t}_{n_i}]$ is a m-dimensional training vector from class i. Denote $N = \sum_{i=1}^{K} n_i$ the total number of training samples in the dataset, and n_i is the subset number of training samples from class i. The learned dictionary is denoted as $D = [D_1, D_2, \ldots, D_K]$, and $V = [V_1, V_2, \ldots, V_N]$ are the coefficient vectors of T over D. The general discriminative dictionary learning (DDL) model can be described as follows:

$$\langle D, V\rangle = \arg\min_{D,V} \Re(T, D, V) + \lambda_1 \|V\|_p^p + \lambda_2 L(V), \qquad (3)$$

where λ_1 and λ_2 are the trade-off parameters, $\Re(T, D, V)$ is the reconstruction term, p denotes ℓ_p-norm, and $L(V)$ denotes the discrimination term.

Note that, to enhance the discriminative capability of a dictionary learning-based method, we can either introduce a discrimination term on the coefficient vectors (e.g., SVGDL [17] method) or perform structure constraints on the learned dictionary (e.g., FDDL [16] method).

3. Frontalization-Based Support Vector-Guided Dictionary Learning

In this section, we will introduce the proposed frontalization-based support vector-guided dictionary learning method in detail. The framework of our method is shown in Figure 1.

3.1. Face Frontalization Method. Given a query image IQ and its landmarks, we can synthesis its frontal view. The following are the detailed steps for face frontalization. By specifying a projection matrix [18] $C_M = A_M[R_M T_M]$, where matrix A_M denotes the intrinsic parameters and $[R_M T_M]$ denotes the extrinsic parameters consisting of a rotation matrix R_M and a translation vector T_M, we can produce a synthetic view of the textured 3D face model. In this paper, we select an extrinsic matrix C_M to produce the frontal view IR of the 3D face model, as shown in Figure 2. This frontal view IR will be used as our reference image as well as the reference coordinate system next.

During the process of producing reference view, for each pixel p' in the reference view, we store its corresponding 3D point coordinates $P = (X, Y, Z)^T$ located on the 3D face model. The correspondence between P and p' can be obtained by

$$p' \sim C_M P. \qquad (4)$$

We denote $P = (X, Y, Z)^T$ as the facial feature points of the query image IQ, and $p' = (x_i', y_i')^T$ are the same facial feature points of the reference view IR. From equation (4), we have the 3D coordinate point $P_i = (X_i, Y_i, Z_i)^T$ corresponding to the facial feature $p_i = (x_i, y_i)^T$ of the query image. This provides the correspondences $p_i^T, P_i^T = (x_i, y_i, X_i, Y_i, Z_i)$ between the query image and the 3D face model; thus, we can estimate the projection matrix $M_Q = A_Q[R_Q T_Q]$, which is approximately the one used to capture the query image.

With the projection matrix MQ, by using the expression,

$$p \sim M_Q P. \qquad (5)$$

We have the correspondences between each pixel of the query image and each 3D coordinate of the 3D face model. Therefore, we have correspondences between each pixel of the query image and each pixel of the reference face. By using bilinear interpolation, we sample the intensities of the query image at pi and then assign it to the pixel coordinates in the reference view. After sampling, we have the frontalized view of the query image.

3.1.1. Face Segmentation. Since the face of synthetic reference image appears in the center of image, after hard frontalization, all the synthetic frontal views of the query images also appear in the center of the image. This helps us to crop the face relevant regions of the frontal views to segment faces from the backgrounds. Following the idea in [10], we crop the 90 × 90 face relevant region of the frontalized image.

3.2. Support Vector-Guided Dictionary Learning. In Section 2.2, we have introduced the general discriminative dictionary learning model. In this section, we will introduce how the constrains are imposed on the coefficients. Commonly, we can assess the discrimination through calculating the similarity of the coefficient vector pairs from the same class and the dissimilarity of the coefficient vector pairs from different classes. Therefore, we can indicate the discrimination capability by applying weight to the squared distance of the coefficient vector pairs v_i, v_j, and we have the following discrimination term $L(V)$:

$$L(V, w_{ij}) = \sum_{i,j\in N} \left\| v_i - v_j \right\|_2^2 w_{ij}. \qquad (6)$$

Suppose the weights w_{ij} can be parameterized as a function $w_{ij}(\beta)$ with respect to a variable β, and it satisfies the following three properties: $(a) w_{ij}(\beta) = w_{ij}(\beta)$; $(b) w_{ij}(\beta) \geq 0$ if v_i and v_j are from the same class $(s_i = s_j)$, and $w_{ji}(\beta) \leq 0$ if v_i and v_j are from different classes $(s_i \neq s_j)$; $(c) \sum_{j=1}^{n} w_{ij}(\beta) = 0, \forall_i$ to balance the contributions of positive weights and negative weights, where $s_i \in \{-1, 1\}$ represents the label of a two-class classification problem; all the coefficient vectors from the same class are labeled 1, and all the coefficient vectors from different classes are labeled -1.

Denote $w_{ij}(\beta) = s_i s_j \beta_i \beta_j$; when $\sum_{j=1}^{n} s_i \beta_j = 0$, the discrimination term can be rewritten as follows:

$$L(V, wij(\beta)) = -2 \sum_{i,j\in N} s_i s_j \beta_i \beta_j v_i^T v_j = \beta^T \Omega \beta, \qquad (7)$$

where Ω is a negative semidefinite matrix; to obtain the extremum of β, we maximize equation (7):

$$\begin{aligned} \langle \beta \rangle &= \arg\max_{\beta} \beta^T \Omega \beta + \gamma(\beta), \\ &\text{s.t.} \beta_i \geq 0, \quad \forall i, \\ &\sum_{j=1}^{n} s_j \beta_j = 0. \end{aligned} \qquad (8)$$

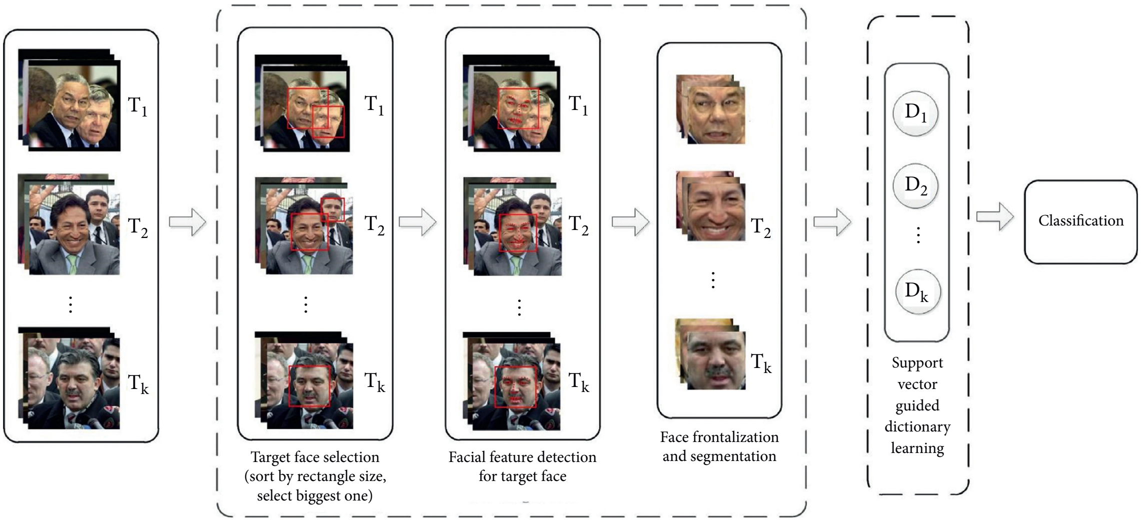

T$_1$
T$_2$
T$_k$
T$_1$
T$_2$
T$_k$
Target face selection (sort by rectangle size, select biggest one)
T$_1$
T$_2$
T$_k$
Facial feature detection for target face
T$_2$
T$_k$
Face frontalization and segmentation
D$_1$
D$_2$
D$_k$
Support vector guided dictionary learning
Classification

Figure 1: Illustration of the proposed method.

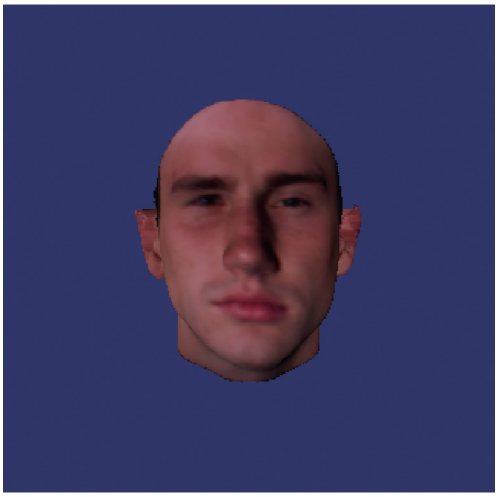


Figure 2: The synthetic reference face of the 3D face model.

By introducing a regularization term $\gamma(\beta) = 4\sum_{i=1}^{n}\beta_i$, we have the parameterized formulation of DDL:

$$\begin{aligned}\langle D, V\rangle &= \arg\min_{D,V}\Bigg(\|T - DV\|_F^2 + \lambda_1\|v\|_p^p + \\ &\qquad \lambda_2 \max_{\beta}\Bigg(\sum_{i,j\in N}\left\|v_i - v_j\right\|_2^2 w_{ij}(\beta) + r(\beta)\Bigg)\Bigg), \\ &= \arg\min_{D,V}\Bigg(\|T - DV\|_F^2 + \lambda_1\|V\|_p^p + \\ &\qquad \lambda_2 \max_{\beta}\Bigg(4\sum_{i=1}^{n}\beta_i - 2\sum_{i,j\in N} s_j s_j \beta_i \beta_j v_i^T v_j\Bigg)\Bigg), \\ &\text{s.t.}\beta_i \ge 0, \quad \forall i, \\ &\sum_{j=1}^{n} s_i\beta_j = 0.\end{aligned} \tag{9}$$

From the settings of $w_{ij}(\beta)$, we can see that the problem on weight assignment is actually the Lagrange dual form of linear support vector machines (SVM) [19]. We impose an additional constraint $\beta_i \le (1/2)\theta$ on β for all i to further reduce the effect of outliers, and θ is a constant parameter. By replacing the subproblem β with its primal SVM, we have the following support vector-guided dictionary learning (SVGDL) model:

$$\begin{aligned}\langle D, V, u, b\rangle &= \arg\min_{D,V,u,b} \|T - DV\|_F^2 \\ &\quad + \lambda_1\|V\|_p^p + 2\lambda_2 L(V, s, u, b),\end{aligned} \tag{10}$$

and $L(V, s, u, b)$ is defined as follows:

$$L(V, s, u, b) = \|u\|_2^2 + \theta\sum_{i=1}^{n}\ell(v_i, s_i, u, b), \tag{11}$$

where u is the normal to the hyperplane of SVM, b is the corresponding bias, $s = [s_1, s_2, \ldots, s_n]$ is the label vector, and $\ell(v_i, s_i, u, b)$ is the hinge loss function. The solution of $\langle u, b\rangle$ can be represented as the linear combination of a few coefficient vectors (support vectors), which means we only assign nonzero weights for pairwise support vectors.

For multiclass classification, we simply adopt one-vs-all strategy by learning C hyperplanes $U = [u_1, u_2, \ldots, u_C]$ and corresponding bias $b = [b_1, b_2, \ldots, b_C]$ and formulate SVGDL as follows:

$$\begin{aligned}\langle D, V, u, b\rangle = \arg\min_{D,V,u,b} & \|T - DV\|_F^2 \\ & + \lambda_1 \|V\|_2^2 + 2\lambda_2 \sum_{C=1}^{C} L(V, s, u, b),\end{aligned} \tag{12}$$

where $S^c = [S_1^c, S_2^c, \ldots, S_n^c]$ if $S_1^c = 1$, and otherwise $S_1^c = -1$.

3.2.1. Optimization. The objective function 12 is not a convex optimization problem because there are four variables D, V, $\langle u, b\rangle$. Therefore, we divide three subprocedures: (a) computing $\langle u, b\rangle$ by fixing D and V; (b) updating V by fixing D and $\langle u, b\rangle$; and (c) updating D by fixing V and $\langle u, b\rangle$. The details are as follows:

Step 1 (initialize D and V and $\langle u, b\rangle$): initialize each class of the dictionary D as the training samples after feature extraction and WPCA transformation with unit ℓ_2 norm. We use zero matrices and zeros vectors to initialize V and $\langle u, b\rangle$.

Step 2 (fix D and V and solve $\langle U, b\rangle$): the minimization of $\langle u, b\rangle$ can be expressed as a multiclass linear SVM problem. Yang et al. [20] proposed a gradient-based method to learn u_c and b_c one by one.

Step 3 (fix D and $\langle u, b\rangle$ and update V): we optimize the coefficient matrix V column by column, and we use the expression

$$\langle v_i\rangle = \arg\min_{v_i} \left\|\hat{t}_i - Dv_i\right\|_2^2 + \lambda_1 \left\|v_i\right\|_2^2 + 2\lambda_2 \sum_{C=1}^{C} \ell\left(v_i, s_i^c, u_c, b_c\right), \tag{13}$$

to formulate the optimization of each v_i.

Step 4 (fix V and $\langle u, b\rangle$ and update D): the optimization of dictionary D can be expressed as

$$\begin{aligned}\langle D\rangle &= \arg\min_{D} \|T - DV\|_F^2, \\ s.t. \left\|d_k\right\|^2 &\leq 1, \quad \forall k \in 1, 2, \ldots, K.\end{aligned} \tag{14}$$

Step 5: go back to Step 2 until the dictionary elements no longer change in two iterations or reach the maximum iteration number.

3.2.2. Classification. After optimization, we have the learned dictionary D and the SVM classifier $\langle u, b\rangle$. For a test image, after feature extraction and Whitening PCA, we have its feature vector $\hat{t}$. We use a matrix $\widehat{P} = (D^T D + \lambda_1 I)^{-1} D^T$ to further project t:

$$\bar{\hat{t}} = \widehat{P}\hat{t}. \tag{15}$$

Then, we use the C linear classifier $\langle u_c, b_c\rangle$ to predict the label of $\overset{\tilde{\wedge}}{t}{}^{\overset{\sim}{\wedge}}$:

$$\text{label} = \arg\max_{c\in 1,2,\ldots,C} u_c^T v + b_c. \tag{16}$$

4. Experiments

Labeled Faces in the Wild (LFW [21]) database is one of the main databases for face images. This database is constructed for studying face recognition problems under unconstrained environment. All the face images are collected from the Internet; therefore, this database shows dramatic variations in illuminations, expressions, skin shades, age groups, and backgrounds. There are more than 13,000 images in this database and nearly 5749 individuals are contained.

In this database, the number of face images of each individual is different, for example, 4069 individuals have only one image, while other 1680 individuals have two or more images. Thus, this database is more suitable for face verification rather than face identification. To evaluate our proposed method on this database, we need to rearrange the original LFW database. To ensure sufficient training samples, we choose the 158 individuals that have at least 10 images for experiments. We randomly select 10 images from each individual and construct a new sub-database denote as LFW-subdatabase. The original size of the images in LFW database is 250 × 250. As mentioned in Section 3.1, we cropped the 90 × 90 pixels face relevant region for all the images in LFW-subdatabase.

4.1. Facial Feature Detection and Frontalization. Before producing the frontal view of the query image, we need to detect the facial features of the query image and the synthetic reference image IR. In recent years, many effective facial feature detection methods were proposed, such as Active Appearance Model (AAM) [22, 23], ZhuRamanan Method [24], Supervised Descent Method (SDM) [25],and Dlib library [26]. In this paper, considering both speed and accuracy, we choose two state-of-the-art methods, SDM [25] and Dlib library [26], to detect the facial features of the images.

4.1.1. Dlib Library. Figure 3 shows the distribution of the 68 facial features that is detected by the Dlib library; the features are mainly located on the eyes, eyebrows, nose, mouth, and jawline.

It is common that an unconstrained image has more than one face in the image, as shown in Figure 4. To synthesis the frontal view of the subject, we need to provide the right facial features of the target subject, which means we need a preprocessing step to select the target face correctly. Suppose there is an image IQ that appears more than one face, by using face detector, we can have the position of each face (usually draw a rectangle on the face, as shown in Figure 5,

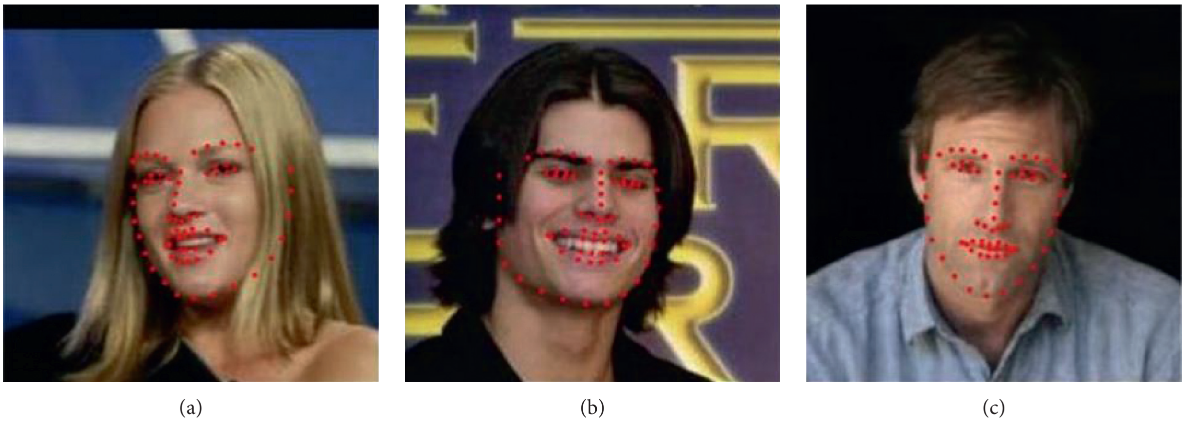


Figure 3: The distribution of the 68 feature points detected by dlib.

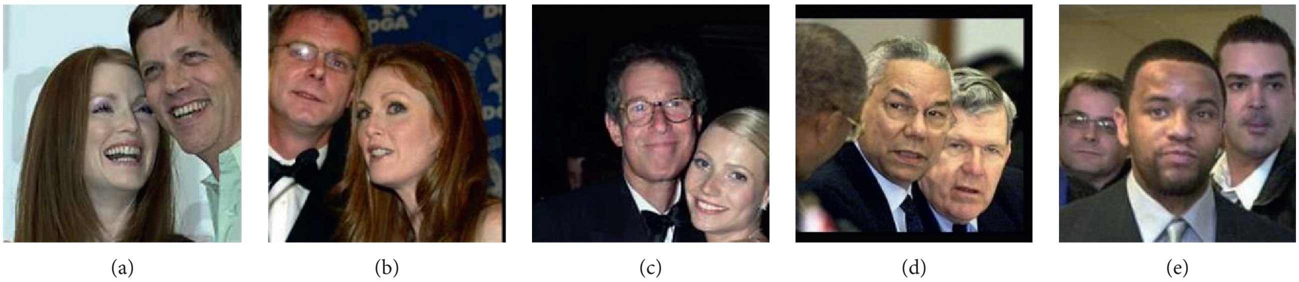


Figure 4: Unconstrained images in the LFW database that have more than one face.

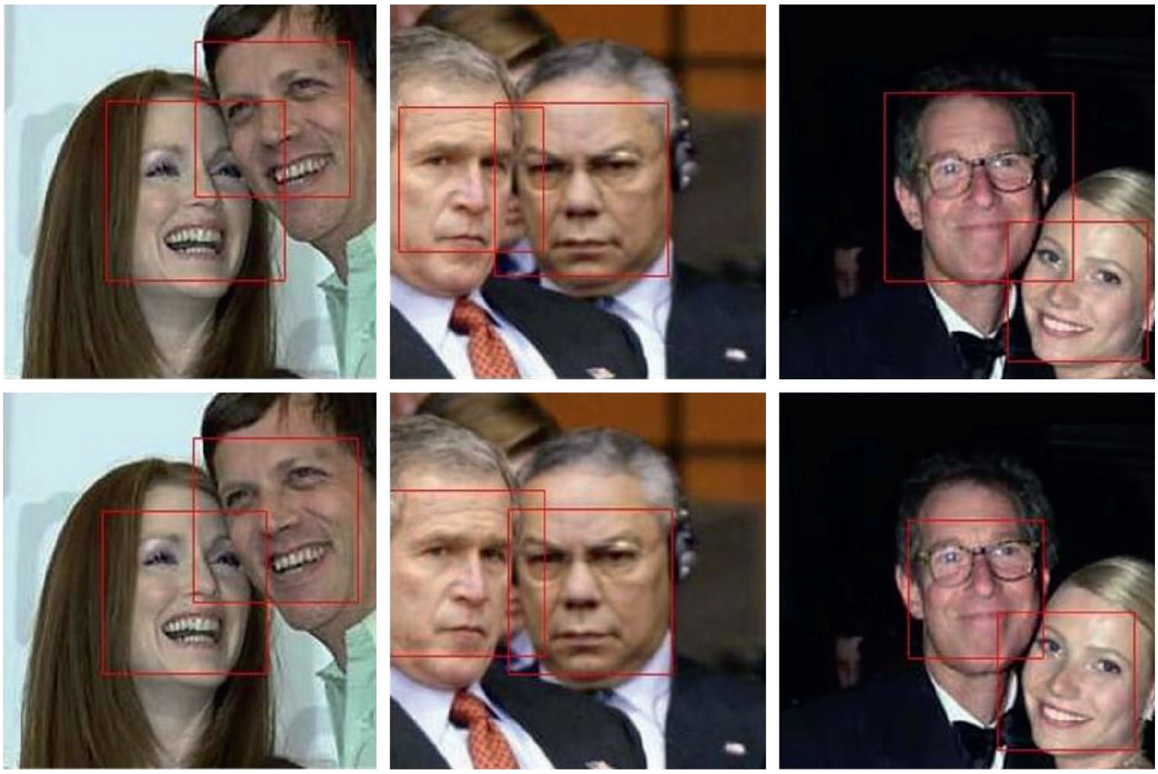


Figure 5: Face frames detected and marked by different face detectors: the above images are detected by dlib face detector and the following images are detected by Viola–Jones face detector.

which means we can calculate the size of the rectangles). In general, the target face usually occupies a relatively larger space with respect to the other faces in the image; thus, the face with bigger size can be regarded as the target face.

However, from Figure 5, it can be seen that the region of the target face drawn by Dlib library is sometimes smaller than other faces in the image (via calculation), while Viola–Jones algorithm [27] performs much better (via calculation). Therefore, we use Viola–Jones algorithm to draw the regions of the faces and sort the face regions by size to find the target face (the largest region). Then, we use Dlib to detect the facial features of the target face and store the feature points of the target face for frontalization. After frontalization, we cropped the 90 × 90 pixels face relevant region; thus, we get a new database consisted of frontalized and cropped face images, and we denote it "LFW3D-dlib" database.

4.1.2. Supervised Descent Method (SDM). Unlike Dlib library, the SDM method detects only 49 facial feature points (without the jawline), as shown in Figure 6.

For SDM face detection method, we also sort the size of the rectangles for all the faces in one image and regard the face with bigger size as the target face. Then, we save the feature points of the target face for frontalization. After frontalization, we cropped the 90 × 90 pixels face relevant region, and we get a new database consisted of frontalized and cropped face images, and we denote it "LFW3D-sdm" database.

4.1.3. Failure of Feature Detection. As introduced above, we need to provide the facial feature of a given image to synthesize its frontal view, and we evaluate the frontalization method on the LFW-subdatabase to generate the frontal views of the images. We know in this database that there are 158 individuals, and each individual has 10 images. However, during facial feature detection, some facial features (only a few number of images) may fail to be detected because of pose variations or other reasons, and thus we cannot produce their frontal views. To ensure that every individual has the same number (10 images per person) of images for experiments, we use their original nonfrontalized images (90 × 90 pixels face relevant region) to fill the vacant positions.

4.1.4. Improvement. Out-of-plane rotation of the head can cause some facial features to be less visible than others, particularly those on the sides of the nose and head. In [10], the authors suggest using conditional soft symmetry to deal with the visibility problem. The process of conditional soft symmetry is borrowing appearances (exclude the eyes) from corresponding symmetric sides of the face where facial features are poorly visible due to pose changes. In this paper, based on LFW database, the authors produced a new database which consisted of frontalized, symmetrized, and cropped face images. We denote this database as "LFW3D-hassner" database.

Although conditional symmetry can deal with the pose-related visibility problems to some extend, it can also introduce face structure problems. For example, if one side of the face is occluded, symmetry can replicate the occlusion, leaving the final result unrecognizable. Therefore, in this paper, we slightly modified the original frontalization method. Firstly, we did not apply the conditional soft symmetry process on the initial frontalized faces. Secondly, we not only consider SDM method but also consider the state-of-the-art detector Dlib Library.

4.2. Notations. In this section, we introduce the scheme of our identification system in detail. We consider several data preprocessing techniques: method illustration, database illustration, feature extraction, feature dimensionality reduction, parameter settings, and our experiment platform.

4.2.1. Method Illustration. We proposed a frontalization-based support vector-guided dictionary learning (FSVGDL) method in this paper. In Section 4.1, we selected two facial feature detection methods: Dlib and SDM for frontalization. Therefore, combining these two detection methods with our FSVGDL method, we have two combined methods named FSVGDL-sdm and FSVGDL-dlib, respectively, for face identification.

4.2.2. Database Illustration. As introduced in Section 4.1, after the frontalization process, we can generate two new databases consisted of frontalized and cropped face images. Therefore, we totally have four versions of LFW database in our experiments denoted as LFW-sub, LFW3D-hassner [10], LFW3D-sdm, and LFW3D-dlib, respectively, as shown in Figure 7. The details of the four databases are as follows:

(1) LFW-subdatabase: this database contains 158 individuals and each individual has 10 images. All the images are segmented from the backgrounds, and the size of each image is 90 × 90.

(2) LFW3D-hassner database: this database is provided by Hassner et al. [10], and all the images are frontalized and cropped into 90 × 90 pixels. We rearranged this database to get the face identification database. We choose the 158 individuals that have at least 10 images, and randomly selected 10 images per individual for experiment.

(3) LFW3D-sdm database: this database is produced by the FSVGDL-sdm method. It includes 1580 images taken from 158 individuals (10 images for each individual). The size of each image is 90 × 90.

(4) LFW3D-dlib database: this database is produced by the FSVGDL-dlib method. It includes 1580 images taken from 158 individuals (10 images for each individual). The size of each image is 90 × 90.

4.2.3. Feature Extraction. In this paper, we select two methods to extract face features: local Gabor feature [4] and hand-craft LBP [3] descriptor which are less sensitive to expression and illumination changes.

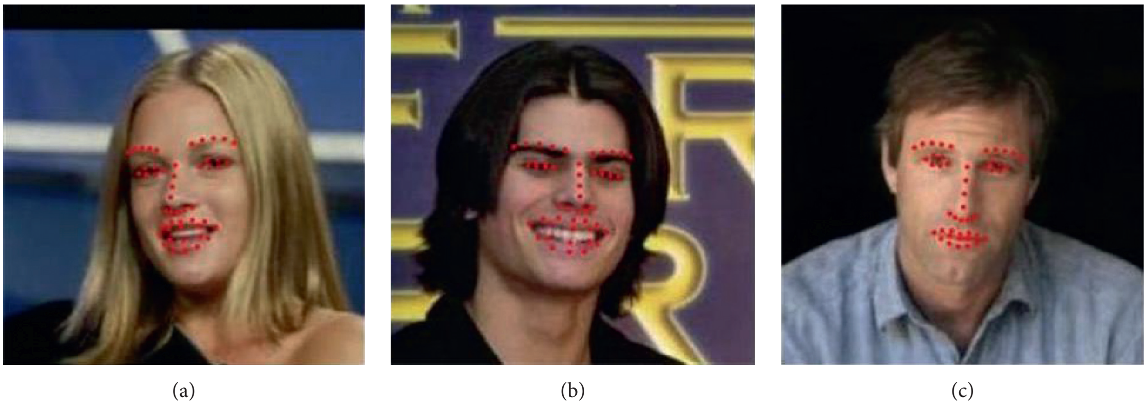


Figure 6: The distribution of the 49 feature points detected by SDM.

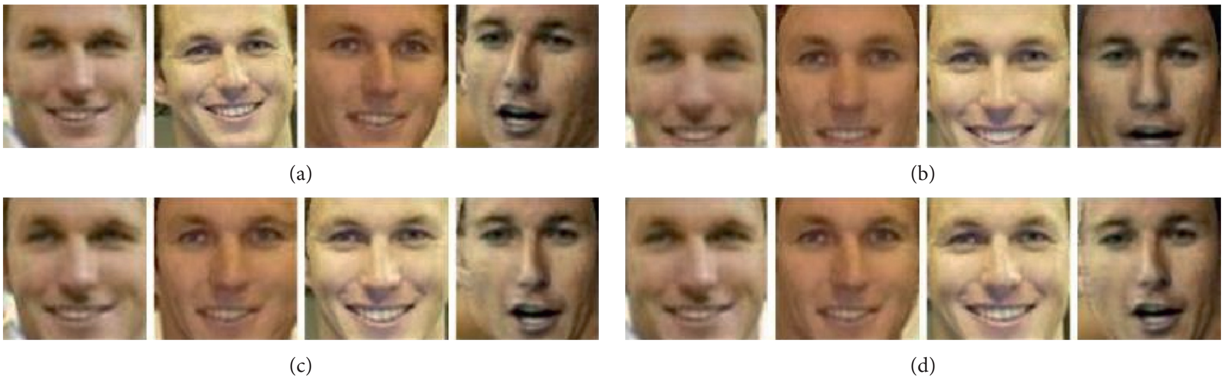


Figure 7: The differences of the same person on the four databases. (a) LFW-sub, (b) LFW3D-hassner, (c) LFW3D-sdm, and (d) LFW3D-dlib.

4.2.4. Feature Dimensionality Reduction. Directly taking facial feature vectors for leaning may cause computational problem. Therefore, we use WPCA to reduce the dimensionality of the facial feature vectors before learning. The transformation matrix of whitening is computed only from the training samples.

4.2.5. Parameter Settings. For both the FSVGDL-sdm and FSVGDL-dlib methods, we set the constant parameter $\theta = 5$, and there are two parameters λ_1, λ_2 to be tuned in the experiments. The parameters are evaluated by 5-fold crossvalidation, and we fix $\lambda_1 = 2 \times 10^{-3}$ and $\lambda_2 = 1 \times 10^{-6}$ for both methods in all the experiments.

4.2.6. Platform. All the experiments are performed on a computer with a 5-core CPU, 8 GB RAM, and 64 bit operating system.

4.3. Effectiveness of FSVGDL Using Gabor Wavelet. We experiment five state-of-the-art methods SRC [11], CRC [12], SVDL [28], FDDL [16], and SVGDL [17] and our FSVGDL method on the LFW-subdatabase to study their performances of face identification. In this experiment, we use Gabor wavelet to extract the features from the face images. The dimension of the extracted Gabor feature is 4840 ($5 \times 8 \times 11 \times 11$, with 5 scales and 8 orientations and the down sampling rate is 8). For the 158 individuals, in order to compare the performance of the proposed method with other methods with varying number of training samples, we randomly select τ images from each individual to construct the training set, and the rest images are used to construct the testing set. To show how our algorithm would benefit from more training samples, we set the value of τ to 4, 5, 6, and 7, respectively. To explore the relationship between feature dimensionality and identification accuracy, we use WPCA to vary the feature dimension. We test the identification performance in the reduced feature dimension of (250; 300; 350; 400; 450) for $\tau = 4$, (300; 350; 400; 450; 500), fr $\tau = 5$, (200, 300, 400, 500, 600) for $\tau = 6$, and (200, 300, 400, 500, 600) for $\tau = 7$, respectively.

We design two subexperiments. (1) In this experiment, we perform experiments on LFW-subdatabase. (2) In this experiment, to show the effectiveness of our improved face frontalization method, we perform experiments on the LFW3D-hassner database for all the competing algorithms, and our FSVGDL method is still evaluated on the LFW-subdatabase:

(1) In this experiment, we evaluate all the algorithms on the LFW-subdatabase. Table 1 lists the best identification accuracy and the corresponding feature

dimension for all the algorithms with τ training samples per individual using Gabor wavelet. Figure 8 shows the plot of identification accuracy versus the different feature dimensions for all the algorithms with τ training samples using Gabor wavelet.

From Table 1, we can see that, for all value of τ, our FSVGDL outperforms all other competing algorithms. When the number of training samples per person increases, all the algorithms perform better.

For $\tau = 4$, the proposed FSVGDL-dlib achieves 83.88% and FSVGDL-sdm achieves 83.12% when the feature dimension is 350, while the maximum identification accuracy of all the competing algorithms is 50.54%.

For $\tau = 5$, the proposed FSVGDL-dlib achieves 89.15% when the feature dimension is 500, and the proposed FSVGDL-sdm achieves 87.45% when the feature dimension is 400. It is worth noting that the identification accuracy increases more than 4% for the proposed methods. While other competing algorithms still does not exceed 60%.

For $\tau = 6$, it can be seen that the proposed FSVGDL-dlib method yields 92.65% when the feature dimension is 400 and is 30% higher than other competing algorithms.

For $\tau = 7$, the identification accuracy increases a lot compared to $\tau = 4$. We can see that FSVGDL-dlib achieves 94.34% when the feature dimension is 600.

(2) In this experiment, all the competing algorithms are performed on the LFW3D-hassner database to show the effectiveness of our improved frontalization method. In other words, we combined the original frontalization process in paper [10] with the competing algorithms to study the performance. We denote the original competing algorithms SRC, CRC, SVDL, FDDL, and SVGDL as SRC-hassner, CRC-hassner, FDDL-hassner, and SVGDL-hassner, respectively. Table 2 lists the best identification accuracy and the corresponding feature dimension for all algorithms with τ training samples per individual using Gabor wavelet. Figure 9 shows the plot of identification accuracy versus the different feature dimensions for all the algorithms with τ training samples using Gabor wavelet, respectively.

Comparing the results in Tables 1 and 2, we can see the accuracy of all the competing algorithms increase more than 20%. This shows the effectiveness of the original frontalization process. However, even though the performance of the competing algorithms improved a lot, our method still achieves the best accuracy no matter how many training samples are used, which shows the effectiveness of our improvement to the original face frontalization method. As shown in Figure 8, it can be seen that when the feature dimension changes, the accuracy of the algorithms also changes.

For $\tau = 4$, the proposed FSVGDL-sdm achieves 83.12% when the feature dimension is 400, and FSVGDL-dlib achieves 85.29% when the feature dimension is 350. The maximum identification accuracy of all the competing algorithms is 81.05%, which is still 2.07% lower than FSVGDL-sdm and 4.24% lower than FSVGDL-dlib.

For $\tau = 5$, we can see that FSVGDL-sdm yields 86.67% and FSVGDL-dlib yields 88.37% when the feature dimension is 400. The identification accuracy of proposed methods is still higher than the maximum accuracy of all the competing algorithms.

For $\tau = 6$, the proposed FSVGDL-sdm achieves 89.54% and FSVGDL-dlib yields 90.85% when the feature dimension is 400, which is slightly higher than the identification accuracy of SVGDL-hassner algorithm.

For $\tau = 7$, it can be seen that, compared with the results when $\tau = 4$, the identification accuracy of the proposed methods increases nearly 10%. This phenomenon indicates that the increasing of training samples can lead to performance gain.

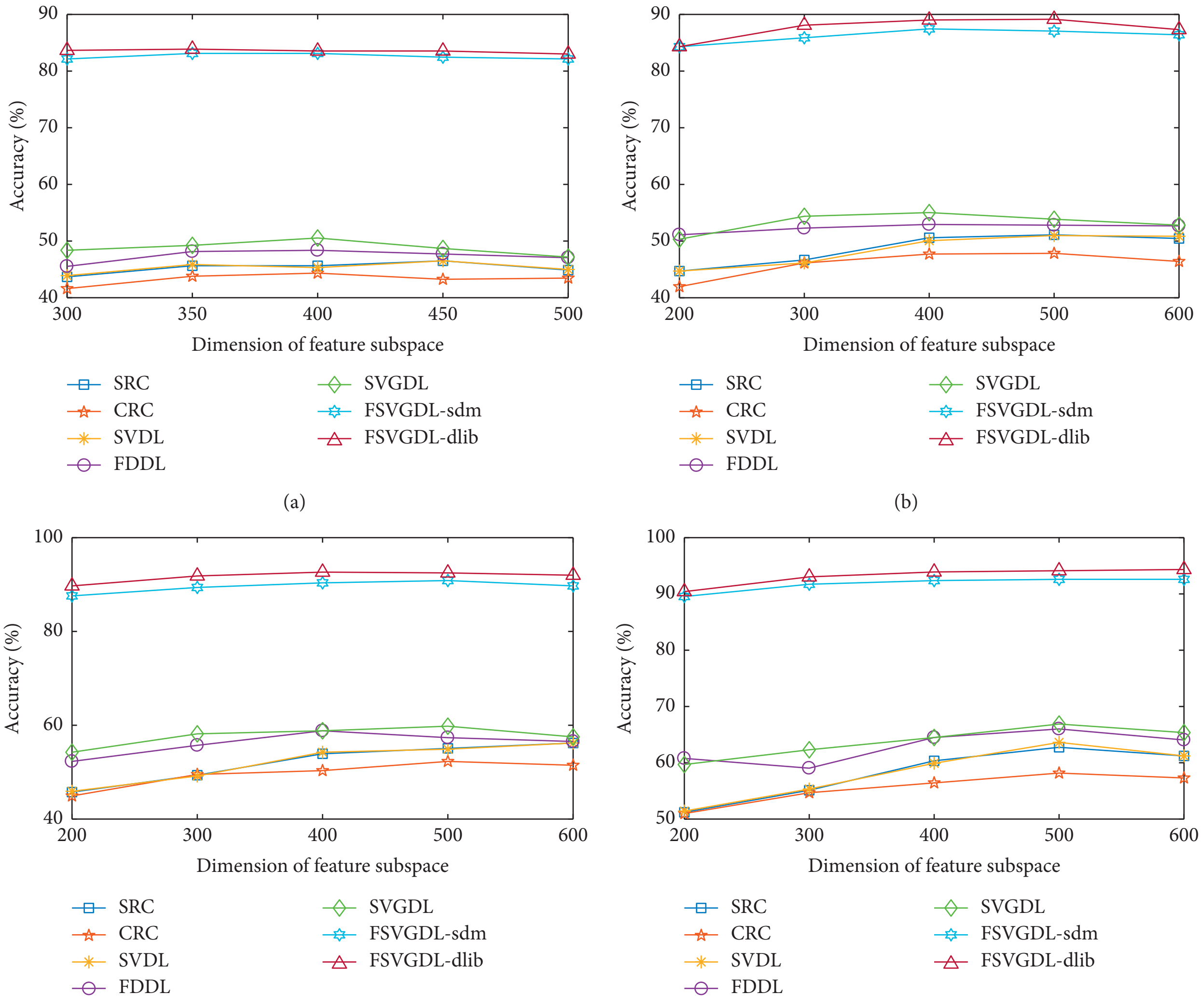
4.4. *Effectiveness of FSVGDL Using LBP Descriptor.*

We experiment five state-of-the-art methods SRC [11], CRC [12], SVDL [28], FDDL [16], and SVGDL [17] and our FSVGDL method to study the performance of face identification on the LFW-subdatabases. In this experiment, we use Local Binary Pattern (LBP) to extract the features from the face images. The feature dimension of LBP is 8496 ($12 \times 12 \times 59$), and the face is divided into 8×8 nonoverlapping blocks. Similar to experiment that uses Gabor wavelet, for the 158 individuals, in order to compare the performance of the proposed method with other methods with varying number of training samples, we randomly select τ images from each individual to construct the training set, and the rest images are used to construct the testing set. Similarly, we set the value of τ to 4, 5, 6, and 7 to show how our algorithm would benefit from more training samples, and by using WPCA, the feature dimension is reduced to (250, 300, 350, 400, 450) for $\tau = 4$, (300, 350, 400, 450, 500) for $\tau = 5$, (200, 300, 400, 500, 600) for $\tau = 6$, and (200, 300, 400, 500, 600) for $\tau = 7$, respectively. We also design two subexperiments. (1) In this experiment, we perform experiments on the LFW-subdatabase for all the algorithms. (2) In this experiment, to show the effectiveness of our improved frontalization process, we perform experiments on the LFW3D-hassner database for all the competing algorithms.

(1) In this experiment, all the algorithms are evaluated on the LFW-subdatabase. Table 3 lists the best identification accuracy and the corresponding feature dimension for all the algorithms with τ training samples per individual using LBP descriptor. Figure 10 shows the plot of identification accuracy versus the different feature dimensions for all the

Table 1: Comparisons of best identification accuracy and the corresponding feature dimension between FSVGDL and other algorithms on LFW database with varying τ number of training samples using Gabor wavelet. Bold are the best performers for each value of τ.

Methods	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$
SRC	46.51% (450)	51.11% (500)	56.21% (600)	62.75% (500)
CRC	44.34% (400)	47.84% (500)	52.29% (500)	58.17% (500)
SVDL	46.51% (450)	50.98% (500)	56.21% (600)	63.62% (500)
FDDL	48.37% (400)	52.94% (400)	58.82% (400)	66.01% (500)
SVGDL	50.54% (400)	53.86% (500)	59.80% (500)	66.88% (500)
FSVGDL-sdm	83.12% (350)	87.45% (400)	90.85% (500)	92.59% (500)
FSVGDL-dlib	**83.88%** (350)	**89.15%** (500)	**92.65%** (400)	**94.34%** (600)



(a)

(b)

(c)

(d)

Figure 8: Identification accuracy versus dimensions of feature subspace of all algorithms with varying τ number of training samples per individual using Gabor wavelet. (a) $\tau = 4$, (b) $\tau = 5$, (c) $\tau = 6$, and (d) $\tau = 7$.

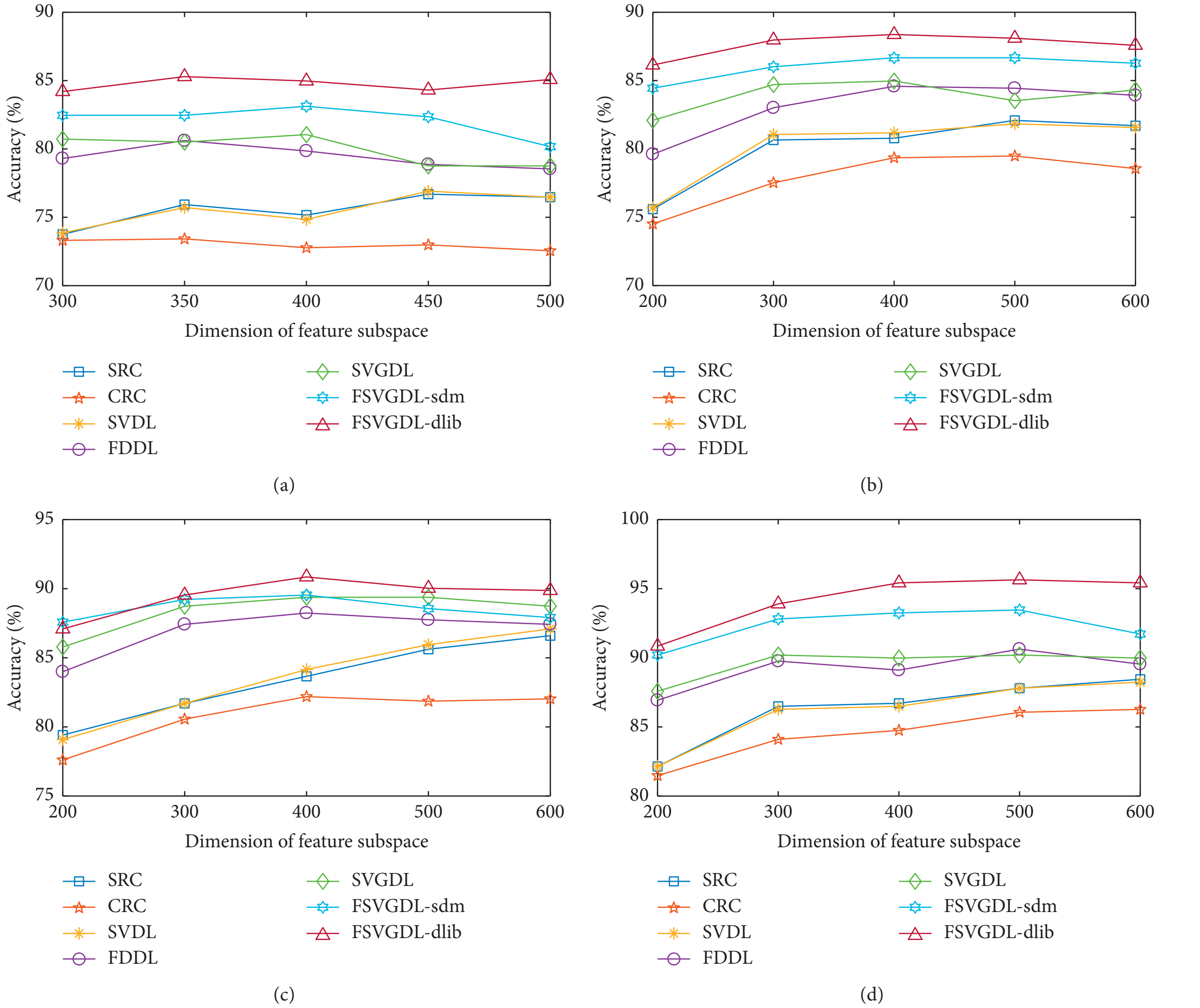
algorithms with τ training samples per individual using LBP descriptor.

From Table 3, by comparing the performance of all the algorithms, we can see the effectiveness of our method. The accuracy varies when the value of τ changes, and the more training samples are used, the better accuracy these algorithms achieve. We can see from Figure 10 that when the feature dimension changes, the accuracy of the algorithms also changes. Comparing the results in Tables 1 and 3, it can be seen that Gabor wavelet is more suitable for feature extraction. All the algorithms achieve better accuracy when using Gabor wavelet.

For $\tau = 4$, the proposed FSVGDL-dlib achieves 82.35% and FSVGDL-sdm achieves 79.96% when the feature dimension is 500, which is 30% higher than all the competing algorithms.

Table 2: Comparisons of best identification accuracy and the corresponding feature dimension between FSVGDL and other algorithms with varying τ number of training samples using Gabor wavelet. Bold are the best performers for each value of τ.

Methods	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$
SRC-hassner	76.69% (450)	82.09% (500)	86.60% (600)	88.45% (600)
CRC-hassner	73.42% (350)	79.48% (500)	82.19% (400)	86.27% (600)
SVDL-hassner	76.91% (450)	81.83% (500)	87.09% (600)	88.24% (600)
FDDL-hassner	80.61% (350)	84.58% (400)	88.24% (400)	90.63% (500)
SVGDL-hassner	81.05% (400)	84.97% (400)	89.38% (400)	90.20% (300)
FSVGDL-sdm	83.12% (400)	86.67% (400)	89.54% (400)	93.46% (500)
FSVGDL-dlib	**85.29%** (350)	**88.37%** (400)	**90.85%** (400)	**95.64%** (500)



(a)

(b)

(c)

(d)

Figure 9: Identification accuracy versus dimensions of feature subspace of all algorithms with varying τ number of training samples per individual using Gabor wavelet. (a) $\tau = 4$, (b) $\tau = 5$, (c) $\tau = 6$, and (d) $\tau = 7$.

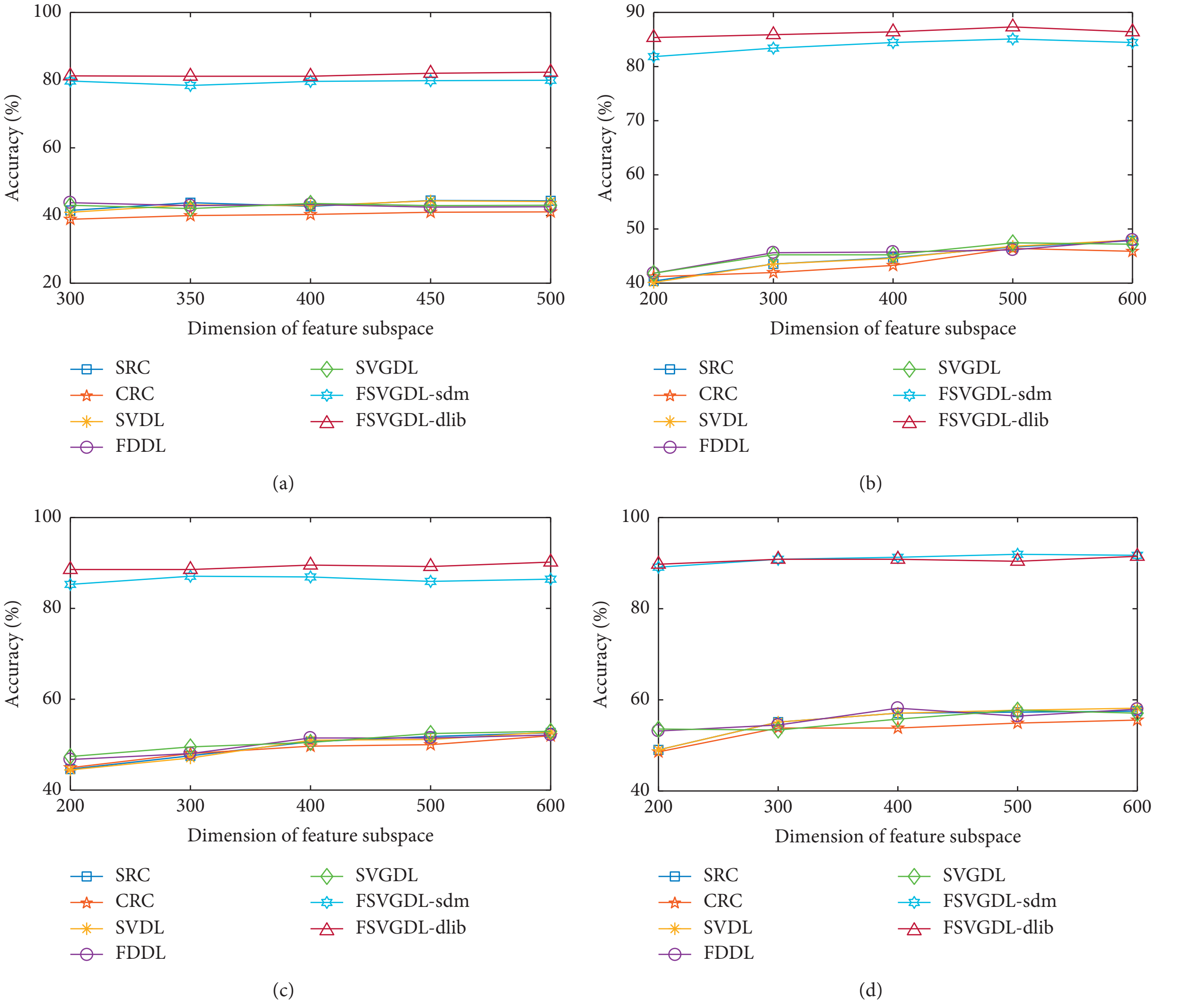
For $\tau = 5$, the proposed FSVGDL-dlib achieves 87.32% and FSVGDL-sdm achieves 85.10% when the feature dimension is 500, while the maximum identification accuracy of all the competing algorithms does not exceed 50%.

For $\tau = 6$, we can see that the proposed FSVGDL-dlib method yields 90.20% when the feature dimension is 600, and FSVGDL-sdm yields 87.09% when the feature dimension is 300, while the maximum identification accuracy of all the competing algorithms is 52.94% with 600 feature dimension.

For $\tau = 7$, the identification accuracy increases a lot compared to $\tau = 4$. It can be seen that FSVGDL-dlib achieves 91.50% when the feature dimension is 600, and FSVGDL-sdm achieves 91.94% when the feature dimension is 500. It is

Table 3: Comparisons of best identification accuracy and the corresponding feature dimension between FSVGDL and other algorithms on LFW database with varying τ number of training samples using LBP descriptor. Bold are the best performers for each value of τ.

Methods	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$
SRC	44.44% (450)	47.84% (600)	52.61% (600)	57.52% (600)
CRC	41.07% (500)	46.41% (500)	51.96% (600)	55.56% (600)
SVDL	44.34% (450)	47.97% (600)	52.78% (600)	58.17% (600)
FDDL	43.25% (400)	47.97% (600)	52.12% (600)	58.17% (400)
SVGDL	43.57% (400)	47.45% (500)	52.94% (600)	57.73% (500)
FSVGDL-sdm	79.96% (500)	85.10% (500)	87.09% (300)	**91.94%** (500)
FSVGDL-dlib	**82.35%** (500)	**87.32%** (500)	**90.20%** (600)	91.50% (600)



(a)

(b)

(c)

(d)

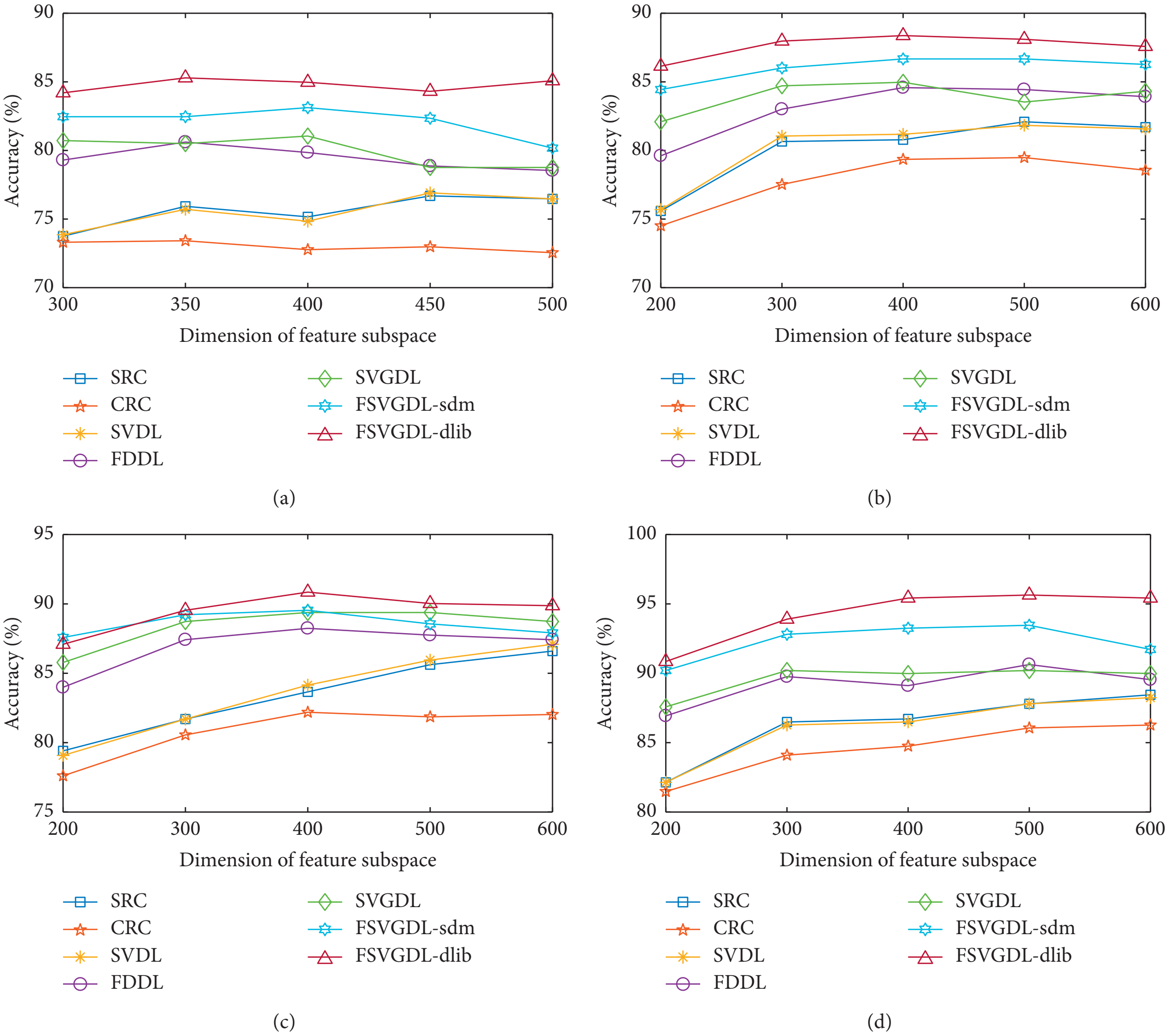
Figure 10: Identification accuracy versus dimensions of feature subspace of all algorithms with varying τ number of training samples per individual using LBP descriptor. (a) $\tau = 4$, (b) $\tau = 5$, (c) $\tau = 6$, and (d) $\tau = 7$.

worth noting that the identification accuracy increases about 10% for the all the algorithms comparing with the results when $\tau = 4$.

(2) In this experiment, all the competing algorithms are performed on the LFW3D-hassner database. Similarly, we denote the original competing algorithms SRC, CRC, SVDL, FDDL, and SVGDL as SRC-hassner, CRC-hassner, FDDL-hassner, and SVGDL-hassner, respectively. Table 4 lists the best identification accuracy and the corresponding feature dimension for all algorithms with τ training samples per individual using Gabor wavelet. Figure 11 shows the plot of accuracy versus the different feature dimensions for six methods with τ training samples using Gabor wavelet, respectively.

Table 4: Comparisons of best identification accuracy and the corresponding feature dimension between FSVGDL and other algorithms with varying τ number of training samples using LBP descriptor. Bold are the best performers for each value of τ.

Methods	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$
SRC-hassner	78.43% (350)	81.96% (600)	84.80% (600)	88.45% (600)
CRC-hassner	75.93% (500)	79.22% (400)	84.48% (300)	87.36% (500)
SVDL-hassner	78.65% (350)	81.96% (600)	84.97% (600)	88.67% (600)
FDDL-hassner	78.98% (450)	81.96% (500)	85.95% (400)	89.32% (300)
SVGDL-hassner	79.52% (300)	82.22% (600)	86.76% (300)	90.85% (300)
FSVGDL-sdm	82.90% (350)	85.10% (300)	**90.36%** (500)	91.50% (400)
FSVGDL-dlib	**85.51%** (300)	**87.84%** (400)	89.38% (600)	**93.03%** (500)



(a)

(b)

(c)

(d)

Figure 11: Identification accuracy versus dimensions of feature subspace of all algorithms with varying τ number of training samples per individual using LBP descriptor. (a) $\tau = 4$, (b) $\tau = 5$, (c) $\tau = 6$, and (d) $\tau = 7$.

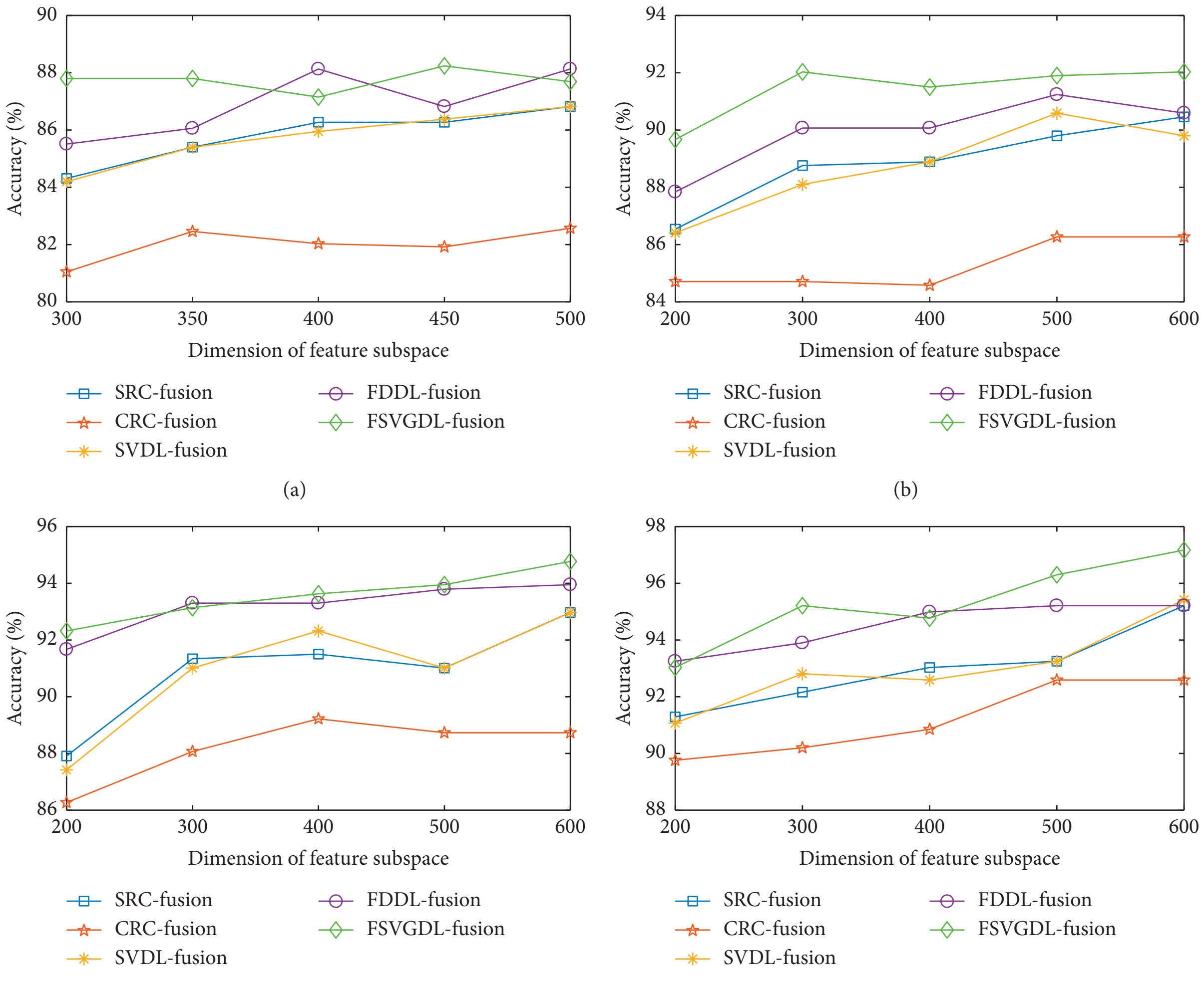
From Tables 3 and 4, it can be seen that the accuracy of all the competing algorithms increase more than 20%, which indicates the effectiveness of frontalization. The performance of our method is better than all other algorithms, which indicates the effectiveness of our improvement on face frontalization. It can be seen from Figure 10 that when the feature dimension changes, the accuracy of the algorithms also changes. Comparing the results in Tables 1 and 3, we can also see that the accuracy of all algorithms is higher when using Gabor wavelet.

For $\tau = 4$, it can be seen that the proposed FSVGDL-dlib yields 85.51%, which is 2.61% higher than FSVGDL-sdm, while FSVGDL-sdm is 3.38% higher than the maximum identification accuracy of all the competing algorithms (79.52%).

For $\tau = 5$, compared with the results using 4 training samples per person, we can see that the best accuracy of all

Table 5: The best fusion accuracy and the corresponding feature dimension for all the algorithms with t training samples per individual. Bold are the best performers for each value of τ.

Methods	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$
SRC-fusion	86.82% (500)	90.46% (600)	92.97% (600)	95.21% (600)
CRC-fusion	82.57% (500)	86.27% (500)	89.22% (400)	92.59% (500)
SVDL-fusion	86.82% (500)	90.59% (500)	92.97% (600)	95.42% (600)
FDDL-fusion	88.13% (400)	91.24% (500)	93.79% (500)	95.21% (500)
FSVGDL-fusion	**88.24% (450)**	**92.03% (300)**	**94.77% (600)**	**97.17% (600)**



(a)

(b)

(c)

(d)

Figure 12: Plot of fusion accuracy versus the different feature dimensions for all the algorithms with τ training samples per individual.

the algorithms increased more than 2%. FSVGDL-dlib achieves 87.84% when the feature dimension is 400.

For $\tau = 6$, FSVGDL-sdm yields 90.36% when the feature dimension is 500 which is slightly higher than FSVGDL-dlib. Compared with the results when $\tau = 5$, the performance of all the algorithms also improved more than 2%.

For $\tau = 7$, the algorithms show a significant improvement compared with the results when $\tau = 6$, the proposed FSVGDL-dlib yields 93.03% when the feature dimension is 500 and FSVGDL-sdm achieves 91.50% when the feature dimension is 400.

We can see from the results in Tables 1–4 that compared with the results using Gabor wavelet, the performance of all the competing algorithms have dropped when using LBP descriptor, which indicates the superiority of Gabor wavelet. We also notice that when using Gabor wavelet, the performance of FSVGDL-dlib is consistently better than FSVGDL-sdm, while in the experiments that using LBP descriptor, FSVGDL-sdm sometimes outperforms FSVGDL-dlib.

4.5. Decision Fusion. Decision fusion generally leads to performance gain [29, 30]. In this experiment, we not only perform fusion on FSVGDL-sdm and FSVGDL-dlib methods but also perform fusion on the Gabor and LBP descriptors. Therefore, we totally have four decisions for a test image. For each test image, we use voting (minority obeying majority) to make the final decision, which class does the test image belongs to.

For all the competing algorithms, we do experiments on the LFW3D-sdm database and LFW3D-dlib database, respectively, wich means we apply our improved frontalization process on all the competing algorithms. With Gabor wavelet and LBP descriptor, each algorithm has four decisions: sdm-Gabor, sdm-LBP, dlib-Gabor, and dlib-LBP. We make decision fusion on these competing algorithms and our proposed FSVGDL with τ training samples per individual on different feature dimensions, respectively. Table 5 lists the best fusion accuracy and the corresponding feature dimension for all the algorithms with τ training samples per individual. Figure 12 shows the plot of fusion accuracy versus the different feature dimensions for all the algorithms with τ training samples per individual, respectively.

Even though the fusion method we applied (voting) is simple, the obtained results in Table 5 show the effectiveness of decision fusion. When using 7 samples Table 5, per person for training and 3 samples for testing, our method FSVGDL achieves 97.17% after fusion, which is 1.75% higher than the maximum identification accuracy of all the competing algorithms.

4.6. Further Analysis. In recent years, lots of deep learning-based approaches are proposed, and some of them (such as DeepID2+ [31], DeepID3 [32], and DSGNN [33]) have achieved a recognition rate of more than 99%. Despite this, deep learning-based approaches are usually data and computation hungry. Our proposed method is computationally efficient and does not need any specialized hardware. For example, it takes less than 1 second for one image to finish the whole frontalization process (including feature detection, frontalization, and segmentation). In addition to time advantage, our frontalization does not need any training compared to that of the DeepFace method in paper [34]. And in the experiment that uses Gabor wavelet (Section 4.3), when the feature dimension is 500, we use 7 image per person for training and 3 image per person for testing, and it takes only 30.6s to train the model and less than 0.0001s to classify one test image.

5. Conclusions

In this paper, we proposed a frontalization-based support vector-guided dictionary learning (FSVGDL) method for unconstrained face identification. The key of FSVGDL is to produce the frontal view of the face image and assign different weights and different pairs of coding vectors adaptively to enhance the discriminative capability.

Extensive experiments have been carried out on the LFW database. Five state-of-the-art algorithms SRC, CRC, SVDL, FDDL, and SVGDL are implemented to compare with the proposed FSVGDL method. The proposed method FSVGDL demonstrates its consistent superiority in its performance with respect to the state-of-the-art methods when considering different feature extraction methods. We also notice that the identification accuracy is better when using Gabor wavelet to extract features. The increasing number of training samples generally leads to the improvement of algorithm performance. The experiments of decision fusion show the effectiveness of fusion, and all the algorithms perform better after fusion. The proposed method yields 97.17% with 7 training samples per person with decision fusion. In the future, we will focus on finding an appropriate dimension of feature subspace to balance the computation cost and recognition accuracy.

Data Availability

Data are available in http://vis-www.cs.umass.edu/lfw/.

Conflicts of Interest

The authors declare that they have no known conflicts of interest.

Acknowledgments

This work was financially supported by the National Natural Science Foundation of China (no. 61773360), Zhejiang Public Welfare Technology Research Social Development Project (Grant no. 2017C33223), Open Project of Key Laboratory of Ministry of Public Security for Road Traffic Safety (no.2020ZDSYSKFKT03-2), and Jiangsu Engineering Research Center of Digital Twinning Technology for Key Equipment in Petrochemical Process (no. DT2020720).

References

[1] M. Turk and A. Pentland, “Eigenfaces for recognition,” *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. fisherfaces: recognition using class specific linear projection,” Tech. Rep., Yale University, New Haven, CT, USA, 1997.

[3] T. Ahonen, A. Hadid, and M. Pietik¨ainen, “Face recognition with local binary patterns,” in *Proceedings of the European Conference on Computer Vision*, pp. 469–481, Springer, Prague, Czech Republic, May 2004.

[4] C. Liu and H. Wechsler, “Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition,” *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society*, vol. 11, no. 4, pp. 467–476, 2002.

[5] X. Fontaine, R. Achanta, and S. S¨usstrunk, “Face recognition in real-world images,” in *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal*

Processing (ICASSP), pp. 1482–1486, IEEE, New Orleans, LA, USA, March 2017.

[6] V. Blanz, K. Scherbaum, T. Vetter, and H.-P. Seidel, "Exchanging faces in images," in *Computer Graphics Forum*, vol. 23, pp. 669–676, Wiley Online Library, Hoboken, NJ, USA, 2004.

[7] H. Tang, Y. Hu, Y. Fu, M. Hasegawa-Johnson, and T. S. Huang, "Real-time conversion from a single 2D face image to a 3D text-driven emotive audio-visual avatar," in *Proceedings of the 2008 IEEE International Conference on Multimedia and Expo*, pp. 1205–1208, IEEE, Hannover, Germany, June 2008.

[8] F. Yang, J. Wang, E. Shechtman, L. Bourdev, and D. Metaxas, "Expression flow for 3D-aware face component transfer," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4, p. 60, 2011.

[9] I. Kemelmacher-Shlizerman and R. Basri, "3D face reconstruction from a single image using a single reference face shape," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, p. 394, 2011.

[10] T. Hassner, S. Harel, E. Paz, and R. Enbar, "Effective face frontalization in unconstrained images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4295–4304, Boston, MA, USA, June 2015.

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

[12] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: which helps face recognition?" in *Proceedings of the 2011 International Conference on Computer Vision*, pp. 471–478, IEEE, Barcelona, Spain, November 2011.

[13] M. Aharon, M. Elad, A. Bruckstein et al., "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, p. 4311, 2006.

[14] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," in *Proceedings of the Computer Vision and Pattern Recognition (CVPR)*, pp. 1697–1704, IEEE, Providence, RI, USA, June 2011.

[15] P. Sprechmann and G. Sapiro, "Dictionary learning and sparse coding for unsupervised clustering," in *Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2042–2045, IEEE, Dallas, TX, USA, March 2010.

[16] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *Proceedings of the 2011 International Conference on Computer Vision*, pp. 543–550, IEEE, Barcelona, Spain, November 2011.

[17] S. Cai, W. Zuo, L. Zhang, X. Feng, and P. Wang, "Support vector guided dictionary learning," in *Proceedings of the European Conference on Computer Vision*, pp. 624–639, Springer, Zurich, Switzerland, September 2014.

[18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2003.

[19] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[20] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1794–1801, IEEE, Miami, FL, USA, June 2009.

[21] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: a database forstudying face recognition in unconstrained environments," in *Proceedings of the Workshop on Faces in "Real-Life" Images: Detection, Alignment, and Recognition*, Marseille, France, October 2008.

[22] G. J. Edwards, T. F. Cootes, and C. J. Taylor, "Face recognition using active appearance models," in *Proceedings of the European Conference on Computer Vision*, pp. 581–595, Springer, Freiburg, Germany, June 1998.

[23] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.

[24] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2879–2886, IEEE, Providence, RI, USA, June 2012.

[25] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 532–539, Portland, OR, USA, June 2013.

[26] D. E. King, "Dlib-ml: a machine learning tool kit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[27] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[28] M. Yang, L. Van Gool, and L. Zhang, "Sparse variation dictionary learning for face recognition with a single training sample per person," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, pp. 689–696, Sydney, Australia, December 2013.

[29] H. V. Nguyen and L. Bai, "Cosine similarity metric learning for face verification," in *Asian Conference on Computer Vision*, pp. 709–720, Springer, Berlin, Germany, 2010.

[30] Q. Cao, Y. Ying, and P. Li, "Similarity metric learning for face recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2408–2415, Sydney, Australia, December 2013.

[31] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2892–2900, Boston, MA, USA, June 2015.

[32] Y. Sun, D. Liang, X. Wang, and X. Tang, "Deepid3: face recognition with very deep neural networks," 2015, https://arxiv.org/abs/1502.00873.

[33] R. Wu, S.-i. Kamata, and T. Breckon, "Face recognition via deep sparse graph neural networks," 2017.

[34] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.