Hindawi
IET Biometrics
Volume 2023, Article ID 9253739, 10 pages
https://doi.org/10.1049/2023/9253739

Research Article

Encoding Coefficient Similarity-Based Multifeature Sparse Representation for Finger Vein Recognition

Lizhen Zhou,[1,2] **Lu Yang**,[3] **Deqian Fu**,[4] **and Gongping Yang**[5]

[1]*Department of Information, ZiBo Normal College, Zibo 255130, China*
[2]*Center for International Education, Philippine Christian University, Manila 1004, Philippines*
[3]*School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China*
[4]*School of Information Science and Engineering, Linyi University, Linyi 276000, China*
[5]*School of Software, Shandong University, Jinan 250101, China*

Correspondence should be addressed to Lu Yang; yangluhi@163.com

Received 7 July 2023; Revised 25 September 2023; Accepted 26 September 2023; Published 25 October 2023

Academic Editor: Christian Rathgeb

Finger vein recognition is a promising biometric technology that has received significant research attention. However, most of the existing works often relied on a single feature, which failed to fully exploit the discriminative information in finger vein images, and therefore led to a limited recognition performance. To overcome this limitation, this paper proposes an encoding coefficient similarity-based multifeature sparse representation method for finger vein recognition. The proposed method not only uses multiple features to extract comprehensive information from finger vein images, but also obtains more discriminative information through constraints in the objective function. The sparsity constraint retains the key information of each feature, and the similarity constraint explores the shared information among the features. Furthermore, the proposed method is capable of fusing all kinds of features, not limited to specific ones. The optimization problem of the proposed method is efficiently solved using the alternating direction multiplier method algorithm. Experimental results on two public finger vein databases HKPU-FV and SDU-FV show that the proposed method achieves good recognition performance.

1. Introduction

Finger vein recognition is a promising biometric technology that captures the unique pattern of blood vessels beneath the skin surface using near-infrared light in a noninvasive way. Unlike other biometric technologies, finger vein patterns are difficult to replicate or forge, which makes finger vein recognition reliable and secure [1, 2]. Recently, finger vein recognition has drawn great attention from researchers.

Researchers have developed numerous methods to enhance the performance of finger vein recognition [3]. However, many existing methods relied on a single feature, which failed to fully capture discriminative information in finger vein images, thus limiting their recognition performance. To address these limitations, researchers have proposed multifeature methods [4–7]. These methods utilized multiple features to represent images and provided more comprehensive information for recognition. However, traditional multifeature methods paid little attention to the similarity between features. In reality, when multiple features describe one image, they inherently share consistent or similar information, which has significant discriminative capability. Furthermore, this shared similarity information can enhance the robustness of recognition. In cases where a feature is disturbed by noises, the similarity information can reduce the negative effect of noises, which brings the robustness of the recognition method.

In order to make full use of multiple features, this paper proposes an encoding coefficient similarity-based multifeature sparse representation (SMSR) method for finger vein recognition. The proposed method introduces an objective function consisting of three components: the reconstruction error, the sparsity constraint on each feature, and the similarity constraint between features. The reconstruction error measures the error between the original data and the reconstruction based on the encoding coefficient and dictionary. The sparsity constraint causes most encoding coefficients to be zeros, while the nonzero coefficients represent the key information of each feature. This helps to reduce the redundancy of feature representation. The similarity constraint encourages the similarity between the encoding coefficients of different features, which effectively captures the common information among features. In the matching stage, the reconstruction errors of multiple features are weighted and fused to further improve the recognition performance.

The SMSR method provides several advantages. First, it considers multiple features simultaneously, to enrich the feature representation. Second, the method obtains more discriminative information for recognition by employing sparsity constraint and similarity constraint. Finally, the SMSR method exhibits strong applicability since it is not limited to specific features.

The paper is organized as follows: Section 2 presents an overview of related work. Section 3 introduces the proposed method. Section 4 reports the experimental results, and Section 5 provides the conclusions.

2. Related Works

2.1. Single-Feature Finger Vein Recognition. In single-feature finger vein recognition, a single feature is extracted to represent an image. Frequently used single-feature extraction methods include the vein pattern method, local descriptor method, deep learning method, and sparse representation method.

(1) Vein pattern method: the vein pattern method is widely used in finger vein recognition, which captures and utilizes vein patterns for recognition. There are several commonly used vein pattern extraction methods, that is, mean curvature (MeanC) [8], anatomy structure analysis-based vein extraction [9], and Gabor [10].

(2) Local descriptor method: the local descriptor method focuses on capturing the texture of local regions of finger vein image. This kind of method utilizes various popular local descriptors, including local binary pattern (LBP) [11], multiscale uniform local binary pattern [12], local line binary pattern [13], and scale-invariant feature transform (SIFT) [14].

(3) Deep learning method: in recent years, deep learning techniques have been widely used to extract finger vein features. In one research, a convolutional neural network [15] was employed to perform feature extraction. Another research introduced a pretrained Xception architecture [16] for both feature extraction and classification. In addition, a convolutional autoencoder [17] was utilized to learn features from finger vein images.

(4) Sparse representation method: there are several finger vein recognition methods based on sparse representation. These methods include nearest centroid neighbor-based sparse representation classification [18], mutual sparse representation classification [19], and sparse reconstruction error constrained low-rank representation (SRLRR) [20]. In particular, SRLRR achieved excellent recognition performance.

2.2. Multifeature Finger Vein Recognition. In multifeature methods, multiple features are extracted to enhance recognition performance. Several researches have explored the use of multiple-feature methods for finger vein recognition. For instance, one method integrated the enhanced maximum curvature method with a histogram of oriented gradient (HOG) descriptor for finger vein recognition [4]. Another method focused on finger vein recognition using local phase quantization and local derivative pattern [5]. In addition, one method employed SIFT and speeded-up robust features as features [7].

Furthermore, there are many multimodal recognition methods that improve performance by fusing features from different modalities [2]. These include fusion of finger vein and fingerprint features [21], fusion of finger vein and knuckle pattern features [22], fusion of finger vein and face features [23], and fusion of finger vein and electrocardiogram features [24].

3. The Proposed Method

We propose an encoding coefficient SMSR method to obtain more discriminative information from finger vein images. Figure 1 illustrates the flowchart of the proposed method. Subsequently, we will provide the detailed descriptions of the proposed method.

3.1. Feature Extraction. In this step, various features are extracted from the finger vein image. Specifically, this study employs two features, that is, Grayvalue and LBP. The Grayvalue utilizes the gray values of an image as a feature. By collecting the gray values of all pixels, a Grayvalue feature vector can be obtained, to represent the gray information of the image. In addition, LBP is used to capture the texture information of an image. The computation of the LBP feature is based on comparing the gray value of each pixel with its neighboring pixels. The binary codes are generated based on the comparison results. These binary codes are then converted into decimal values, which are used as the second feature. By utilizing both the Grayvalue feature and the LBP feature, the proposed method obtains the gray and texture information of the finger vein image.

3.2. Encoding Coefficient Similarity-Based Multifeature Sparse Representation. The encoding coefficient SMSR defines an objective function that integrates the sparsity constraint for each feature and the similarity constraint between features. The defined objective function is given as follows:

$$\min_{C_i} \frac{1}{2}\sum_{i=1}^{M}\|X_i - D_iC_i\|_F^2 + \alpha\sum_{i=1}^{M}\|C_i\|_1 + \frac{\beta}{2}\sum_{i=1}^{M}\sum_{j=1,j\neq i}^{M}\left\|C_i - C_j\right\|_F^2. \tag{1}$$

In this formula, M represents the number of features. X_i denotes the feature matrix of the testing images, where i is the feature index. Each column of X_i corresponds to a feature vector of a testing image. C_i represents the encoding coefficient matrix, and D_i denotes the dictionary consisting of

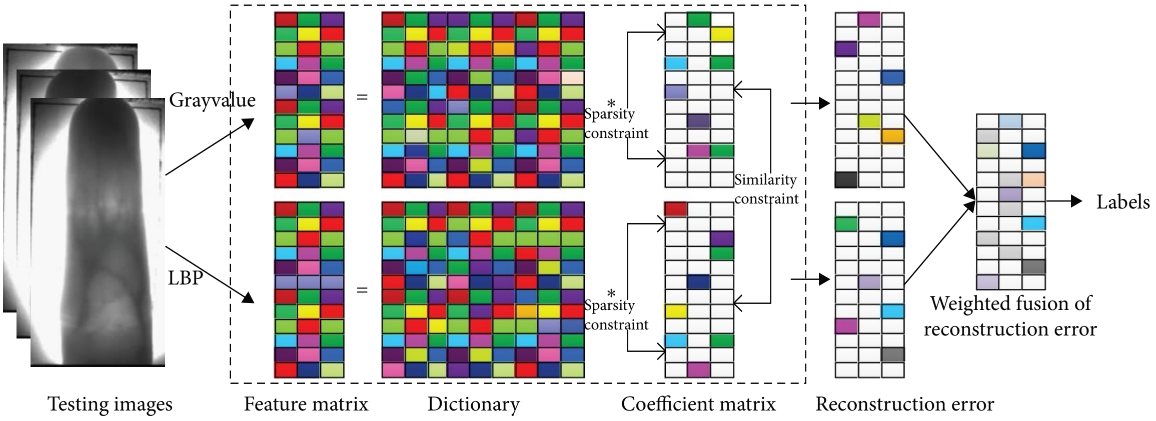


FIGURE 1: Recognition flowchart of testing images.

training image features. $\|\cdot\|_F^2$ denotes the squared Frobenius norm of a matrix, and $\|\cdot\|_1$ represents the $L1$ norm.

The objective function consists of three components:

(1) Reconstruction error: $\frac{1}{2}\sum_{i=1}^{M}\|X_i - D_iC_i\|_F^2$ measures the difference between the original data X_i and its reconstruction using the corresponding dictionary D_i and encoding coefficient matrix C_i. By minimizing the reconstruction error, the objective is to obtain a sparse encoding coefficient matrix for reconstructing the original data.

(2) Sparsity constraint: $\alpha\sum_{i=1}^{M}\|C_i\|_1$ promotes the sparsity of the coefficient matrix C_i. By minimizing the sparsity constraint, the objective function encourages most of the elements in the coefficient matrix C_i to be zero, thus selecting the most important information.

(3) Similarity constraint: $\frac{\beta}{2}\sum_{i=1}^{M}\sum_{j=1,\, j\neq i}^{M}\left\|C_i - C_j\right\|_F^2$ encourages the similarity between coefficient matrices C_i and C_j. By minimizing the similarity constraint, the coefficients between different features tend to be similar, and the shared information between different features of an image is explored.

The optimization problem aims to find the coefficient matrix C_i that minimizes the reconstruction error and keeps the sparsity and similarity constraints.

3.3. Optimization Process. The alternating direction method of multipliers (ADMM) [25, 26] is a powerful optimization technique that has gained significant attention due to its ability to decompose complex optimization problems into simpler subproblems. Therefore, we use ADMM to solve the objective function Equation (1).

By introducing the auxiliary variable L_i, the objective function can be transformed into:

$$\min_{C_i,L_i} \frac{1}{2}\sum_{i=1}^{M}\|X_i - D_iC_i\|_F^2 + \alpha\sum_{i=1}^{M}\|L_i\|_1 + \frac{\beta}{2}\sum_{i=1}^{M}\sum_{j=1,j\neq i}^{M} \left\|C_i - C_j\right\|_F^2 \text{ s.t. } L_i = C_i, i = 1, 2, \ldots, M. \tag{2}$$

The augmented Lagrange function is obtained by introducing the Lagrange multiplier matrix Y_i:

$$L(C_i, L_i, Y_i) = \frac{1}{2}\sum_{i=1}^{M}\|X_i - D_iC_i\|_F^2 + \alpha\sum_{i=1}^{M}\|L_i\|_1 + \frac{\beta}{2}\sum_{i=1}^{M}\sum_{j=1,j\neq i}^{M}\left\|C_i - C_j\right\|_F^2 + \sum_{i=1}^{M}\langle Y_i, C_i - L_i\rangle + \frac{\mu}{2}\sum_{i=1}^{M}\|C_i - L_i\|_F^2, \tag{3}$$

where $\langle\cdot,\cdot\rangle$ represents the inner product of two matrices and μ is a penalty factor. The function of Equation (3) can be solved using an iterative update method. When a variable is updated, other variables are fixed. The update steps are given as follows:

(1) Update C_i: fix other variables, C_i can be updated by solving the following problems:

$$C_i^{k+1} = \arg\min_{C_i} \frac{1}{2}\|X_i - D_iC_i\|_F^2 + \frac{\beta}{2}\sum_{j=1,j\neq i}^{M}\left\|C_i - C_j^k\right\|_F^2 + \left\langle Y_i^k, C_i - L_i^k\right\rangle + \frac{\mu^k}{2}\left\|C_i - L_i^k\right\|_F^2. \tag{4}$$

By taking the derivative of C_i and setting it to zero, we obtain

$$C_i^{k+1} = \left[D_i^TD_i + \beta(M-1)I + \mu^kI\right]^{-1}\left(D_i^TX_i + \beta\sum_{j=1,\, j\neq i}^{M}C_j^k + \mu^kL_i^k - Y_i^k\right), \tag{5}$$

where I is the identity matrix.

(2) Update L_i: fix other variables, L_i can be updated by solving the following problems:

$$L_i^{k+1}= \arg\min_{L_i} \alpha\|L_i\|_1 + \left\langle Y_i^k, C_i^{k+1} - L_i\right\rangle + \frac{\mu^k}{2}\left\|C_i^{k+1} - L_i\right\|_F^2. \tag{6}$$

By taking the derivative of L_i and setting it to zero, the following formula can be obtained.

$$\alpha\text{sign}(L_i) - Y_i^k + \mu^k\left(C_i^{k+1} - L_i\right) = 0, \tag{7}$$

where $\text{sign}(L_i)$ is defined as follows:

$$\text{sign}(L_i) = \begin{cases} 1, & \text{if } L_i > 0 \\ -1, & \text{if } L_i < 0 \\ 0, & \text{if } L_i = 0 \end{cases}. \tag{8}$$

Using the soft threshold method, the update formula of L_i is obtained.

$$L_i^{k+1}= \max\left(0, C_i^{k+1} + \frac{Y_i^k}{\mu^k} - \frac{\alpha}{\mu^k}\right) + \min\left(0, C_i^{k+1} + \frac{Y_i^k}{\mu^k} + \frac{\alpha}{\mu^k}\right). \tag{9}$$

(3) Update Y_i and μ: the Lagrange multiplier Y_i and μ are updated.

$$Y_i^{k+1} = Y_i^k + \mu\left(C_i^{k+1} - L_i^{k+1}\right), \tag{10}$$

$$\mu^{k+1} = \min\left(\rho\mu^k, \mu_{\max}\right), \tag{11}$$

in which the parameter ρ is used to control the update speed of the penalty parameter μ. We set ρ to 1.6, and the maximum value of μ is $\mu_{\max} = 10^8$.

In each iteration, we compute the difference between C_i^{k+1} and L_i^{k+1} as r_i. The norms of r_i are then calculated and compared to the threshold value. If the maximum norm of r_i is less than the threshold or the iteration reaches the maximum value, the algorithm terminates. Otherwise, the value of μ is updated, and the algorithm continues with the next iteration.

The optimization process is given in Algorithm 1.

```
**Input**: Feature matrix $X_i$ of testing images, feature matrix $D_i$ of training images
**Output**: Coefficient matrix $C_i$
Initialize: $C_i = 0, L_i = 0, Y_i = 0, \rho = 1.6, \mu^0 = 0.1, \mu_{\max} = 10^8$
**For** $K = 1$ to 200 **do**
    Fix other variables and update $C_i$ as Equation (5)
    Fix other variables and update $L_i$ as Equation (9)
    Fix other variables and update $Y_i$ as Equation (10)
    If $\max(\|r_i\|) < 10^{-6}$ or $k >= 200$ break
    Else update $\mu$
**Endfor**
```

ALGORITHM 1: SMSR optimization process.

3.4. *Classification.* In this section, we present the classification of a testing image. For the feature matrix X_i of all the testing images, the coefficient matrix C_i is obtained by solving function Equation (1), which represents the linear combination of X_i using the dictionary D_i, that is, $X_i \approx D_i C_i$. Specifically, for a testing image x, its ith feature vector is denoted as x_i, and the ith coefficient vector is denoted as c_i. The reconstruction error of x_i on D_i can be calculated:

$$s_{x_i} = \|x_i - D_i c_i\|^2. \tag{12}$$

Subsequently, the reconstruction errors of different features are fused, denoted by s_x.

$$s_x = \sum_{i=1}^{M} w_i s_{x_i}, \tag{13}$$

where w_i is the weight, which represents the contribution of the ith feature to the final reconstruction error s_x.

Finally, the testing image x is assigned to the category with the smallest reconstruction error, so x is assigned to the category labeled $\text{label}(x)$ according to the following formula:

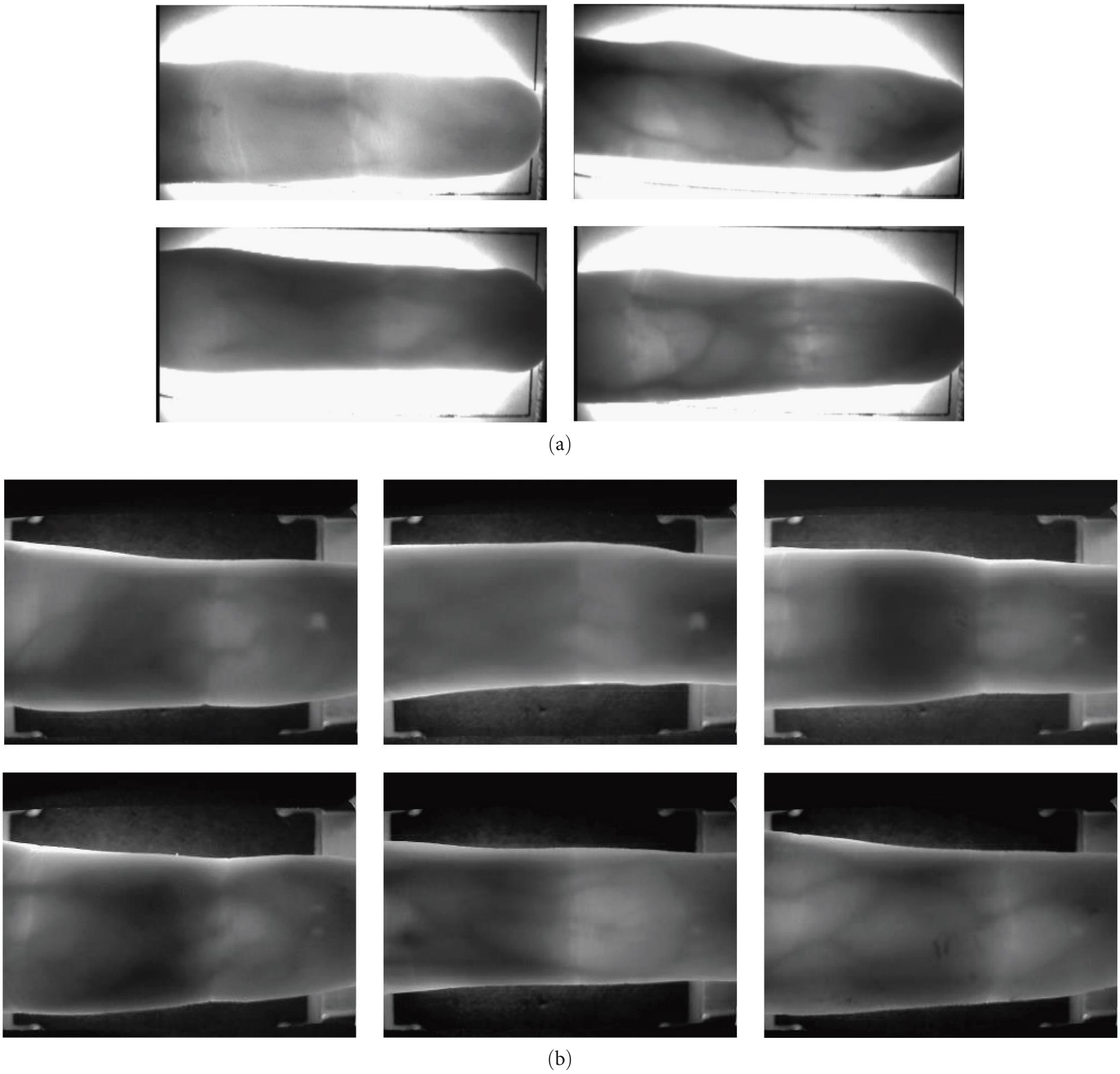
$$\text{label}(x) = \arg\min(s_x). \tag{14}$$

4. Results and Discussion

4.1. Experiment Settings. In order to ensure the reliability of the experimental results, we conduct experiments on two public finger vein databases: HKPU-FV database from Hong Kong Polytechnic University [27] and SDU-FV database from Shandong University [28]. Some finger vein images from two databases are shown in Figure 2.

(1) HKPU-FV database contains finger vein images from 156 volunteers. The acquisition process consists of two stages. In each stage, the volunteers provided two fingers for image acquisition, and six images were obtained for each finger. Since only 105 volunteers participated in the second stage, the number of finger vein images per person in the database varies. Therefore, we only use the finger vein images collected in the first stage, totaling 1,872 (312 × 6) finger vein images from 312 fingers.

(2) SDU-FV database involves 106 volunteers. Each volunteer provided finger vein images from six fingers, and six images were obtained for each finger. Therefore, the database has a total of 3,816 (636 × 6) finger vein images from 636 fingers.



(a)

(b)

Figure 2: Finger vein images from two public databases. (a) Images from HKUP-FV and (b) images from SDU-FV.

For the experiments, the first half images of each finger are used as training data, and the remaining images are used as testing data. Therefore, HKPU-FV database and SDU-FV database have 936 (312 × 3) training images and 1,908 (636 × 3) training images.

To reduce interference from the image background, features are extracted from the region of interest (ROI) image. The ROI is obtained using the method in Yang et al. [29]. Initially, the Sobel operator is used to detect the finger boundaries, and then the midpoints of the boundaries are identified. These midpoints are utilized to correct the skew angle in the image. Next, a sliding window technique is employed to determine the height of the ROI, while the width is determined by using the internal tangents of the finger edges. Once both the height and width of the ROI are determined, the ROI of the finger vein image is obtained. For the convenience of processing, the size of the ROI is normalized into 96 × 64 pixels using bilinear interpolation.

In our experiments, we test the performance of our method in both identification and verification modes. In the identification mode, we use the recognition rate as a benchmark, which measures the rate of correctly recognized images overall testing images. In the verification mode, we utilize the equal error rate (EER) as a performance evaluation metric.

4.2. Comparison with Single-Feature Sparse Representation. This experiment aims to compare the proposed SMSR

Table 1: Performance comparison between SFSR and SMSR methods.

		HKPU-FV		SDU-FV	
Method	Feature	Recognition (%)	EER (%)	Recognition (%)	EER (%)
SFSR	Grayvalue	91.67	2.24	72.38	7.72
	Gabor	91.88	3.11	60.06	18.99
	LBP	98.93	0.22	84.07	5.35
	HOG	98.82	0.63	77.94	8.68
Proposed SMSR	Grayvalue + Gabor	98.08	0.73	78.67	9.95
	Grayvalue + HOG	99.68	0.33	90.15	4.25
	Gabor + LBP	99.68	0.32	91.72	3.44
	Gabor + HOG	99.25	0.44	87.74	5.50
	LBP + HOG	99.79	**0.11**	91.82	3.41
	Grayvalue + LBP	**99.89**	0.12	**93.87**	**2.61**

Bold values represent the best results.

method with the single-feature sparse representation (SFSR) method. The experiment utilizes Grayvalue, Gabor, LBP, and HOG features, and the recognition results are presented in Table 1. These results clearly demonstrate that, compared to the SFSR method, the SMSR method achieves significantly higher recognition rates and lower EERs on two databases. For example, on HKPU-FV database, the recognition rates for Grayscale and Gabor based on SFSR are 91.67% and 91.88%, and EER values are 2.24% and 3.11%, respectively. When our Grayscale and Gabor-based SMSR method is used, the recognition rate increases to 98.08% and the EER reduces to 0.73%. Similar trends can be observed from other features. The SMSR method outperforms the SFSR method in recognition performance, which is attributed to a more comprehensive representation of finger vein images by multiple features. In addition, this experiment shows that SMSR methods can fuse different features, rather than specific ones.

Furthermore, on HKPU-FV database, the SMSR method, utilizing the combination of Grayvalue and LBP features, achieves the highest recognition rate of 99.89% with a lower EER of 0.12%. Similarly, on SDU-FV database, the same combination also yields the highest recognition rate of 93.87%, and the lowest EER of 2.61%. Considering the outstanding performance of the combination of Grayvalue and LBP features on two databases, subsequent experiments will utilize the Grayscale and Gabor-based SMSR method.

4.3. Effectiveness of Similarity Constraint. This experiment is conducted to evaluate the impact of the similarity constraint on the objective function of the SMSR method. To assess the contribution of this constraint, we remove it from the objective function to obtain the nonconsistent multifeature sparse representation (NC-MFSR) method. The objective function of the NC-MFSR method is given as follows:

$$\min_{C_i} \frac{1}{2}\sum_{i=1}^{M}\|X_i - D_iC_i\|_F^2 + \alpha\sum_{i=1}^{M}\|C_i\|_1. \tag{15}$$

The experimental results are presented in Table 2, which clearly demonstrates the performance difference between the SMSR and NC-MFSR methods. On HKPU-FV database, the SMSR method achieves a recognition rate of 99.89% and an EER of 0.12%, outperforming the NC-MFSR method, which has a lower recognition rate of 98.93% and a higher EER of 0.56%. Similarly, on SDU-FV database, the SMSR method achieves a recognition rate of 93.87% and an EER of 2.61%, which also surpasses the NC-MFSR method. These results show the effectiveness of the similarity constraint in the SMSR method. The similarity constraint in the SMSR method captures the shared information between different features, which enhances the discriminative ability of the method. Furthermore, when a feature is affected by noise, the similarity constraint is able to compensate for the affected feature by exploiting the stable information. This mutual compensation can also improve the overall recognition performance.

4.4. Comparison of Different Fusion Methods. The proposed SMSR method is compared with several commonly used multifeature fusion methods [30] to demonstrate its superior ability. The compared methods are also based on sparse representation. Detailed descriptions of the compared multifeature fusion methods are given as follows:

(1) Parallel feature fusion: this method requires that the feature vectors of an image have the same dimensionality. Therefore, PCA is applied to reduce the dimensionality of two feature vectors, and the two vectors are weighted and fused to obtain a combined feature vector. Sparse representation is subsequently employed to obtain encoding coefficients, and the reconstruction error is used for classification.

(2) Serial feature fusion: in this method, the two feature vectors of an image are concatenated to form a larger feature vector. Similarly, sparse representation is used to obtain encoding coefficients, and the reconstruction error is utilized for classification.

(3) Score fusion: this method uses a SFSR method to obtain the encoding coefficients of each feature. Each feature is individually reconstructed, and the reconstruction errors of two features are weighted and fused for classification.

Table 2: Performance under different constraints.

	HKPU-FV		SDU-FV	
Method	Recognition rate (%)	EER (%)	Recognition rate (%)	EER (%)
NC-MFSR	98.93	0.56	92.35	3.07
Proposed SMSR	**99.89**	**0.12**	**93.87**	**2.61**

Bold values represent the best results.

Table 3: Performance of different fusion methods.

	HKPU-FV		SDU-FV	
Method	Recognition rate (%)	EER (%)	Recognition rate (%)	EER (%)
Parallel feature fusion	98.82	0.23	84.28	5.18
Serial feature fusion	99.36	0.21	83.91	5.29
Score fusion	99.47	0.32	84.54	5.23
Proposed SMSR	**99.89**	**0.12**	**93.87**	**2.61**

Bold values represent the best results.

The results presented in Table 3 clearly demonstrate that the recognition performance of the SMSR method is superior to other multifeature fusion methods. Among these compared methods, the score fusion method achieves better recognition performance. However, it still cannot catch up with the SMSR method, as it fails to fully use the discriminative information from multiple features.

The objective function of the SMSR method simultaneously considers the sparsity constraint of each feature and the similarity constraint between features. In this way, it not only fully exploits key information from each feature but also extracts common information between features. The obtained information is jointly employed for recognition, enhancing the overall discriminative capability of the method. Therefore, for the multifeature fusion task, the SMSR method can obtain more discriminative information from multiple features and achieves good recognition performance.

4.5. Parameter Discussion. This experiment involves several parameters: the sparsity constraint parameter α, the similarity constraint parameter β, and the weight parameter w_i.

The parameter α controls the degree of sparsity in the representation. By setting α to a larger value, the optimization process is more inclined to generate a sparser encoding coefficient matrix C_i. The parameter β determines the strength of the similarity constraint. A higher value of β promotes greater similarity between coefficient matrices C_i and C_j. The weight parameter w_i is used to fuse the reconstruction errors. As two features are used, and $w_2 = 1 - w_1$, the experiment only needs to focus on the value of w_1.

Initially, we set α to 0.001 and β to 0.01 in order to adjust the parameter w_1. According to the results in Table 4, $w_1 = 0.5$ yields the best recognition rate on HKPU-FV database, and $w_1 = 0.3$ achieves the highest recognition rate on SDU-FV database. Therefore, in the experiments of adjusting α and β, we set $w_1 = 0.5$ on HKPU-FV database, and $w_1 = 0.3$ on SDU-FV database.

Next, α is fixed at 0.001, and the value of β is adjusted. Based on the results shown in Table 5, the highest recognition rate is achieved when β is set to 0.1 on both HKPU-FV and SDU-FV databases.

Finally, with β fixed at 0.1, the value of α is varied. The results in Table 6 indicate that the optimal recognition rate

Table 4: Recognition rates for different w_1 values.

	HKPU-FV	SDU-FV
Value of w_1	Recognition rate (%)	Recognition rate (%)
0.1	97.76	90.99
0.2	98.18	91.72
0.3	98.72	**92.87**
0.4	99.47	92.66
0.5	**99.79**	92.40
0.6	99.68	91.40

Bold values represent the best results.

Table 5: Recognition rates for different β values.

	HKPU-FV	SDU-FV
Fix $\alpha = 0.001$, vary β	Recognition rate (%)	Recognition rate (%)
0.3	99.79	93.71
0.2	99.79	93.71
0.1	**99.89**	**93.87**
0.01	99.79	92.87
0.001	99.57	92.14
0.0001	99.57	92.09

Bold values represent the best results.

Table 6: Recognition rates for different α values.

	HKPU-FV	SDU-FV
Fix $\beta = 0.1$, vary α	Recognition rate (%)	Recognition rate (%)
0.1	97.65	90.04
0.01	**99.89**	93.50
0.001	99.89	**93.87**
0.0001	99.89	93.55
0.00001	99.79	93.29
0.000001	99.79	93.24

Bold values represent the best results.

Table 7: Performance of different methods.

	HKPU-FV		SDU-FV	
Method	Recognition rate (%)	EER (%)	Recognition rate (%)	EER (%)
LLBP [31]	93.58	5.33	68.08	15.47
LDC [32]	98.08	3.07	76.57	10.77
MeanC [8]	86.97	6.75	67.09	18.26
Weighted vein code indexing[a] [33]	98.82	1.11	86.90	7.92
ASAVE[a] [9]	99.47	0.60	85.27	7.29
SRLRR[a] [20]	99.79	0.33	90.41	3.75
Proposed SMSR	**99.89**	**0.12**	**93.87**	**2.61**

Note: [a]Recognition rates and EER are cited from literature [20]. Bold values represent the best results.

on HKPU-FV database is achieved with $\alpha = 0.01$, and the highest recognition rate on SDU-FV database is obtained with $\alpha = 0.001$. Consequently, on HKPU-FV database, we use $\alpha = 0.01$ and $\beta = 0.1$, and on SDU-FV database, we use $\alpha = 0.001$ and $\beta = 0.1$.

In summary, on HKPU-FV database, the highest recognition rate is achieved with $\alpha = 0.01$, $\beta = 0.1$, and $w_1 = 0.5$. And on SDU-FV database, the highest recognition rate is achieved with $\alpha = 0.001$, $\beta = 0.1$, and $w_1 = 0.3$.

4.6. Comparison with State-of-the-Art Methods. The purpose of this experiment is to compare the performance of the SMSR method with the state-of-the-art finger vein recognition methods on two databases. The results are summarized in Table 7.

The experimental results indicate that the SMSR method achieves better recognition performance on both the HKPU-FV and SDU-FV databases. On HKPU-FV database, it achieves the highest recognition rate of 99.89% and the lowest EER of 0.12%. Similarly, on SDU-FV database, it achieves a recognition rate of 93.87% with an EER of 2.61%. By contrast, the classical methods such as ASAVE, weighted vein code indexing, and SRLRR exhibit relatively good performance but do not catch up with the SMSR method. The promising performance of our SMSR is mainly attributed to its effectiveness in extraction of discriminative information.

4.7. Time Performance. This experiment aims to assess the time cost of the SMSR method on HKPU-FV database. It runs on a PC equipped with a 3.80 GHz processor and 32.00 GB of memory, using MATLAB 2018 software.

Table 8 presents the time spent by the SMSR method to classify 936 testing images. The SMSR classification involves both offline and online steps. The most time-consuming step is the reconstruction error calculation, which takes approximately 11.47 s, and the classification step only requires 0.03 s. In total, the SMSR method takes about 18.22 s for online recognition of 936 testing images, averaging about 0.02 s per image. This demonstrates the high efficiency of the proposed SMSR method.

In addition, Table 9 presents the time costs of our proposed SMSR method and two classical methods, that is, LDC [32] and MeanC [8]. In Table 9, for the SMSR method, the feature extraction step includes all online steps except classification. The SMSR method spends more time in the feature extraction step, approximately 18.19 s. This is mainly due to the cost of the sparse representation and reconstruction error calculation of all testing images. In the classification step, LDC and MeanC methods spend more time, that is, 43.10 and 80.39 s, respectively. This is because they need to match each testing image with all 936 registered images. In summary, the SMSR method is superior to the LDC and MeanC methods in this time cost test.

Table 8: Time cost(s) of SMSR.

	Substeps	Time cost
Offline	Dictionary construction	2.02
Online	Grayscale feature extraction	0.84
	LBP feature extraction	1.18
	Sparse representation	4.70
	Reconstruction error calculation	11.47
	Classification	0.03

Table 9: Computational time(s) comparison in identification mode.

Method	Feature extraction	Classification	Total
LDC	3.65	39.45	43.10
MeanC	1.39	79.00	80.39
Proposed SMSR	18.19	0.03	18.22

5. Conclusion

In this paper, we propose an encoding coefficient SMSR method for finger vein recognition. This method uses multiple features to comprehensively represent finger vein images. To extract more discriminative information, we introduce the sparsity and similarity constraints into the objective function. The sparsity constraint retains the unique information of each feature, and the similarity constraint explores the common information among the features. In the matching stage, we weight and fuse the reconstruction errors of multiple features to further enhance recognition performance. Our experimental results on two public finger vein datasets, HKPU-FV and SDU-FV, demonstrate the advantages of our proposed method in recognition performance and efficiency.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (grant number 62076151) and the Shandong Provincial Natural Science Foundation (grant number ZR2022MF331), in part by the Taishan Scholar Project of Shandong Province (grant number tsqn202211182) and Youth Innovation Team of Shandong Province Higher Education Institutions (grant number 2022KJ205).

References

[1] K. Shaheed, H. Liu, G. Yang, I. Qureshi, J. Gou, and Y. Yin, "A systematic review of finger vein recognition techniques," *Information*, vol. 9, no. 9, Article ID 213, 2018.

[2] K. Shaheed, A. Mao, I. Qureshi, M. Kumar, S. Hussain, and X. Zhang, "Recent advancements in finger vein recognition technology: methodology, challenges and opportunities," *Information Fusion*, vol. 79, pp. 84–109, 2022.

[3] G. K. Sidiropoulos, P. Kiratsa, P. Chatzipetrou, and G. A. Papakostas, "Feature extraction for finger-vein-based identity recognition," *Journal of Imaging*, vol. 7, no. 5, Article ID 89, 2021.

[4] M. A. Syarif, T. S. Ong, A. B. J. Teoh, and C. Tee, "Enhanced maximum curvature descriptors for finger vein verification," *Multimedia Tools and Applications*, vol. 76, no. 5, pp. 6859–6887, 2017.

[5] K. Kapoor, S. Rani, M. Kumar, V. Chopra, and G. S. Brar, "Hybrid local phase quantization and grey wolf optimization based SVM for finger vein recognition," *Multimedia Tools and Applications*, vol. 80, no. 10, pp. 15233–15271, 2021.

[6] D. Feng, S. He, Z. Zhou, and Y. Zhang, "A finger vein feature extraction method incorporating principal component analysis and locality preserving projections," *Sensors*, vol. 22, no. 10, Article ID 3691, 2022.

[7] I. Kovač and P. Marák, "Finger vein recognition: utilization of adaptive Gabor filters in the enhancement stage combined with SIFT/SURF-based feature extraction," *Signal, Image and Video Processing*, vol. 17, no. 3, pp. 635–641, 2023.

[8] W. Song, T. Kim, H. C. Kim, J. H. Choi, H.-J. Kong, and S.-R. Lee, "A finger-vein verification system using mean curvature," *Pattern Recognition Letters*, vol. 32, no. 11, pp. 1541–1547, 2011.

[9] L. Yang, G. Yang, Y. Yin, and X. Xi, "Finger vein recognition with anatomy structure analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 8, pp. 1892–1905, 2018.

[10] Y. Zhang, W. Li, L. Zhang, X. Ning, L. Sun, and Y. Lu, "Adaptive learning Gabor filter for finger-vein recognition," *IEEE Access*, vol. 7, pp. 159821–159830, 2019.

[11] K. Shaheed and I. Qureshi, "A hybrid proposed image quality assessment and enhancement framework for finger vein recognition," *Multimedia Tools and Applications*, pp. 1–26, 2022.

[12] N. Hu, H. Ma, and T. Zhan, "Finger vein biometric verification using block multi-scale uniform local binary pattern features and block two-directional two-dimension principal component analysis," *Optik*, vol. 208, Article ID 163664, 2020.

[13] H. Liu, L. Yang, G. Yang, and Y. Yin, "Discriminative binary descriptor for finger vein recognition," *IEEE Access*, vol. 6, pp. 5795–5804, 2018.

[14] W. Kang, Y. Lu, D. Li, and W. Jia, "From noise to feature: exploiting intensity distribution as a novel soft biometric trait for finger vein recognition," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 4, pp. 858–869, 2019.

[15] R. Das, E. Piciucco, E. Maiorana, and P. Campisi, "Convolutional neural network for finger-vein-based biometric identification," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 2, pp. 360–373, 2019.

[16] K. Shaheed, A. Mao, I. Qureshi et al., "DS-CNN: a pre-trained Xception model based on depth-wise separable convolutional neural network for finger vein recognition," *Expert Systems with Applications*, vol. 191, Article ID 116288, 2022.

[17] B. Hou and R. Yan, "Convolutional auto-encoder based deep feature learning for finger-vein verification," in *2018 IEEE International Symposium on Medical Measurements and Applications (MEMEA)*, pp. 1–5, IEEE, 2018.

[18] S. Shazeeda and B. A. Rosdi, "Nearest centroid neighbor based sparse representation classification for finger vein recognition," *IEEE Access*, vol. 7, pp. 5874–5885, 2019.

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

[20] L. Yang, G. Yang, K. Wang, F. Hao, and Y. Yin, "Finger vein recognition via sparse reconstruction error constrained low-rank representation," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4869–4881, 2021.

[21] H. Ren, L. Sun, J. Guo, and C. Han, "A dataset and benchmark for multimodal biometric recognition based on fingerprint and finger vein," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2030–2043, 2022.

[22] S. Daas, A. Yahi, T. Bakir, M. Sedhane, M. Boughazi, and E.-B. Bourennane, "Multimodal biometric recognition systems using deep learning based on the finger vein and finger knuckle print fusion," *IET Image Processing*, vol. 14, no. 15, pp. 3859–3868, 2020.

[23] Y. Wang, D. Shi, and W. Zhou, "Convolutional neural network approach based on multimodal biometric system with fusion of face and finger vein features," *Sensors*, vol. 22, no. 16, Article ID 6039, 2022.

[24] B. A. El-Rahiem, F. E. A. El-Samie, and M. Amin, "Multimodal biometric authentication based on deep fusion of electrocardiogram (ECG) and finger vein," *Multimedia Systems*, vol. 28, no. 4, pp. 1325–1337, 2022.

[25] E. Esser, "Applications of Lagrangian-based alternating direction methods and connections to split Bregman," *CAM Report*, vol. 9, Article ID 31, 2009.

[26] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[27] A. Kumar and Y. Zhou, "Human identification using finger images," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2228–2244, 2012.

[28] Y. Yin, L. Liu, and X. Sun, "SDUMLA-HMT: a multimodal biometric database," in *Biometric Recognition: 6th Chinese Conference, CCBR 2011*, vol. 6, pp. 260–268, Springer, Beijing, China, December 3–4, 2011.

[29] L. Yang, G. Yang, Y. Yin, and R. Xiao, “Sliding window-based region of interest extraction for finger vein images,” *Sensors*, vol. 13, no. 3, pp. 3799–3815, 2013.

[30] U. G. Mangai, S. Samanta, S. Das, and P. R. Chowdhury, “A survey of decision fusion and feature fusion strategies for pattern classification,” *IETE Technical Review*, vol. 27, no. 4, pp. 293–307, 2010.

[31] B. A. Rosdi, C. W. Shing, and S. A. Suandi, “Finger vein recognition using local line binary pattern,” *Sensors*, vol. 11, no. 12, pp. 11357–11371, 2011.

[32] X. Meng, G. Yang, Y. Yin, and R. Xiao, “Finger vein recognition based on local directional code,” *Sensors*, vol. 12, no. 11, pp. 14937–14952, 2012.

[33] L. Yang, G. Yang, X. Xi, K. Su, Q. Chen, and Y. Yin, “Finger vein code: from indexing to matching,” *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1210–1223, 2018.