FRIH: A face recognition framework using image hashing

Mahsa Ghasemi[1] · Hamid Hassanpour[1]

Abstract

Face recognition is one of the most important research topics in computer vision. Indeed, the face is an important means of communication with humans and it is highly needed for daily contact. Face recognition technology is applied in many biometric applications such as security, video surveillance, access control systems, and forensics. In this technology, hashing has recently made encouraging progress due to its fast retrieval speed and low storage cost. In this work, we propose an effective face recognition framework based on hashing functions. It attempts to leverage a cascaded architecture with two stages of analyzing different visual information based on image hashing. Specifically, we first introduce a filter to overlook a large number of dissimilar identities in terms of local visual information. Similar identities are found quickly through random independent hash functions inspired by Locality Sensitive Hashing (LSH). Next, we further refine candidates and recognize the most similar identities according to global visual information. The global feature is obtained by hashing each face into a high-quality binary feature space using Discrete Cosine Transform (DCT) coefficients. The proposed method is evaluated on three well-known and one combined face dataset. The obtained results, and the provided face recognition application program, demonstrate that the proposed framework improves the recognition rate and significantly reduces recognition time.

Keywords Face recognition · Image hashing · LSH · DCT based hashing algorithm

1 Introduction

Face recognition has always been efficacious in biometric technologies since dealing with the human face [1–3]. It has applications in various areas such as security access control, human–computer interaction, surveillance, and social media [4, 5]. Over the past decades, different face recognition algorithms have been proposed in the literature, such as Deep learning-based face recognition [6], Local pattern-based face recognition [7–9], 3D face recognition [10], and Geometrical face recognition [11].

✉ Hamid Hassanpour
   h.hassanpour@shahroodut.ac.ir

1   Faculty of Computer Engineering & IT, Shahrood University of Technology, Shahrood, Iran

Face recognition methods generally include two stages: feature extraction and classification. The classifier returns an identity from the face gallery by evaluating the feature vector which refers to a test image. In the literature, several feature descriptors have been introduced to provide a robust way of representing faces that are invariant to uncontrolled conditions, such as misalignment, illumination, and pose of the face. Existing descriptors referring to facial images are in two categories: holistic descriptors and local descriptors [27]. The holistic descriptors, such as Principal Component Analysis (PCA) [28] and Linear Discriminant Analysis (LDA) [29], represent the appearance features in a vector. These descriptors are sensitive to local characteristics such as expression and occlusion. The local descriptors consider fiducial points like the nose, eyes, and mouth. If some fiducial points are not clearly available in the image, other points can be employed for recognition. Common local feature descriptors include local binary patterns (LBP) [16], and Gabor filters [18].

Success in face recognition requires consideration of both the local and global features of the face, as human beings do to identify individuals. Hence, some works use different image encoding methods to extract both local and global features from the images [30]. Chen et al. [31] proposed a face recognition system using a combination of support vector machine (SVM) and PCA. In their method, face images are compressed by PCA, i.e. a combination of local and global features is employed, and classified using SVM. Currently, deep learning-based methods are advanced in face recognition and can achieve highly successful results [2, 12, 13]. However, these methods have high computational costs and need considerable training data [6, 14].

Existing feature extraction methods can be classified into two categories: hand-crafted features [16–18] and learning-based features [19–21]. The hand-crafted features work well against the variations of the controlled scenarios. In contrast, the learning-based features work well against the variations of the uncontrolled scenarios. Among them, the binary codes obtained by learning-based hashing methods are advantageous over other descriptors [15]. The main goal of image hashing is to use a small representation of samples to provide low storage costs and fast computational and matching speed [22, 23]. Due to their advantages, hashing methods have been studied extensively in recent years [24–26].

Focusing on the benefits provided by image hashing, this paper proposes a new framework to boost the performance of face recognition. The main purpose of this paper is to provide robust face recognition with high accuracy and a reduced computational cost. The proposed method consists of two stages. In the first stage, it refines the candidates by overlooking a large number of dissimilar identities in terms of local features. For this matter, it chooses candidate subjects quickly through hashing tables. Finally, it uses a robust image hashing method to further refine the result according to the holistic features and to recognize the test image.

Thanks to this new multi-task framework, the performance of face recognition can be remarkably improved even when there is only one sample per person to train the system. Analyzing different visual information in two filtering stages improves accuracy under semi-unconstrained conditions. Experimental results show high performance where only a single sample per person is available. We have also prepared an application to show the recognition step of our method.[1]

The rest of the paper is organized as follows: Section 2 presents related works, Section 3 reviews the literature, and Section 4 shows the proposed approach. Also, experimental results and discussions are presented in Section 5. Finally, the conclusion is drawn in Section 6.

---

[1] https://www.dropbox.com/sh/fwuexmevandd49s/AABirrubOXbG5yHUZD3SbZT0a?dl=0.

2 Related works

There are various face recognition methods in the literature. They can be classified into three categories: holistic-based methods [1], component-based methods [12, 13], and those based on features from both of the former categories [8, 14, 15].

The number of face images per identity in the gallery set plays a key role in the recognition rate. Many of the face recognition techniques are not promising when there is only a single sample per person (SSPP). For this reason, this challenge has recently come into sharp focus [16]. Lu et al. [17] developed a patch-based SSPP face recognition method, namely Discriminative Multi-Manifold Analysis (DMMA). The training samples were segmented to obtain patches and have been used to learn discriminative features. Abdelmaksoud et al. [18] proposed a deep learning method that overcomes the SSPP problem in a face recognition system. They used 3D face reconstruction to increase the reference gallery set with different poses. The authors in [19] developed a scheme using a combination of traditional and deep learning methods and introduced a well-trained deep convolutional neural network (DCNN) model for face recognition.

Another challenge in face recognition is recognition under uncontrolled conditions [4, 39, 40]. The authors in [20] presented the feature-based method using speeded-up robust features (SURF) and scale-invariant feature transform (SIFT). They achieved high accuracy for face recognition. Deng et al. [21] proposed Extended Sparse Representation-based Classification (ESRC) for face recognition. Their method achieves high recognition results regarding the challenges of occlusion and non-occlusion. The authors in [22] developed the Subspace ESRC (SESRC) and Discriminative Feature Learning (LDF) for face recognition. The authors in [23] proposed the Recurrent Regression Neural Network (RRNN) framework to unify two classic tasks of cross-pose face recognition. In [1], the authors introduced an Evolutionary Single Gabor Kernel (ESGK) based filter approach and extracted Gabor energy feature vectors from face images. The authors in [13] proposed a hybrid approach based on a combination of Probabilistic Neural Networks (PNNs) and Improved Kernel Linear Discriminant Analysis (IKLDA) to solve some of the face recognition problems.

Recently, many deep learning techniques, specifically convolutional neural network, have been proposed for face recognition [2, 12, 24–26]. These methods have made a huge leap and improved the performance of face recognition systems significantly. Almabdy et al. [19] examined the performance of the AlexNet model with a multi-class SVM classifier for face recognition and their approach could enhance recognition rates. Anand et al. [27] applied the GoogleNet deep learning model in face recognition and they achieved fairly good accuracy in recognizing faces. Yee et al. [28] proposed a recognition system by doing transfer learning on SqueezeNet to classify faces. In their work, they tried to compress the size of the network while preserving accuracy. Santoso et al. [29] introduced a method for face recognition using OpenFace. They achieved an acceptable result in the face recognition system. Gruber et al. [30] presented an application of 50-layer deep neural network ResNet architecture to face recognition and they showed the great potential of residual learning for face recognition.

In recent years, many works have been developed in the field of computer vision about image hashing [31–33]. In existing hashing strategies, LSH is a classic hashing approach that generates hashing functions by random projections [34]. Dehghani et al. [35] have evaluated different versions of LSH according to different hash families. Cassio et al. [36] introduced an efficient hashing method for face identification using the LSH and the Partial

Least Squares (PLS) approaches. They showed a notable reduction in the number of identities and provided high speed for face identification. From another point of view, Dai et al. [37] proposed a novel method for face recognition called Bayesian Hashing to generate binary codes for achieving competitive performance with significant memory costs. DCT has been extensively used in image processing, such as image feature extraction and image hashing [38]. Tang et al. [39] introduced a framework to generate robust image hashes using DCT coefficients. Chen et al. [40] developed a learning-based hashing model to learn local binary descriptors in the Hamming space. They developed local feature hashing with a binary auto-encoder (LFH-BAE) to directly learn local binary descriptors in the Hamming space. Tang et al. [41] proposed a supervised hashing method for scalable face image retrieval. Lei et al. [42] developed a method called Spherical Hashing-based Binary Codes (SHBC) to learn a robust binary face descriptor. Tang et al. [41] proposed a novel Deep Hashing based on Classification and Quantization errors (DHCQ) for scalable face image retrieval. This method is a deep convolutional network that generates hash codes by learning discriminative feature representations. Tuncer et al. [43] presented a novel face recognition method based on the perceptual hash using a quintet triple binary pattern (QTBP) and Discrete Wavelet Transform (DWT). Kong et al. [44] proposed a novel face recognition face inspired by a deep learning model principal component analysis network (PCANet). They used two-directional PCA for feature extraction and binary hashing, block-wise histograms, and linear SVM for the output stage. Their results showed their approach is a promising method for face recognition.

3 Literature review

In this section, we first describe the techniques employed in the proposed approach. As mentioned earlier, we use both local and global features for face recognition. We employ the SURF descriptor to extract local features which are then mapped to a hash table using the LSH algorithm. The DCT-based hashing algorithm is also applied to the face image to consider global features.

3.1 The SURF Descriptor

The SURF [45] is recognized as a fast and efficient scale and rotation invariant descriptor to generate robust features in image deformations. It was introduced to overcome the computational complexity of the Scale Independent Feature Transform (SIFT) descriptor [46]. The SURF descriptors use the Haar box filters to approximate the Laplacian of Gaussian in which convolution with these box filters can be calculated quickly.

The SURF is a local descriptor for an image, which is formulated in two main steps: First, points of interest are selected from distinctive regions like T-junctions, corners, and blobs. The SURF descriptor finds points of interest by using a Hessian matrix. Second, the neighborhood around each interesting point is described by a distinctive feature vector. Also, this descriptor is robust to detection errors, noise, and photometric and geometric deformations. In this regard, for orientation assignment before forming the key point descriptor, it uses the responses to Haar Wavelets. For feature description, a neighborhood around the key point is divided into 4×4 sub-regions, then for each sub-region i, the horizontal and vertical Wavelet responses are taken to make SURF feature descriptor V_i as follows:

$$V_i = \left[\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y| \right] \tag{1}$$

The horizontal and vertical directions of the Haar Wavelet responses are d_x and d_y, respectively. Finally, the feature vectors are joined from each sub-region to make a SURF descriptor with a 64-dimensional vector as follows:

$$SURF = [V_1, \dots, V_{16}] \tag{2}$$

Each descriptor is robust to image scaling, translation, rotation, and to some extent is robust to 3D projection and illumination changes [45]. Also, it can improve speed and discriminative power.

3.2 LSH Hashing

LSH is a hashing framework to approximate high-dimensional similarity searches without any sub-linear dependence on the size of data. LSH extends dynamic insertion and deletion using the hashing scheme without the complexity of tree structures and attains a large speed over a tree-based data structure. The approach proposed in this work employs voting-based hashing instead of searching trees. LSH maps similar feature descriptors to the same hash bucket and keeps dissimilar features in different buckets of the table. The LSH output is formed in three steps: first, it builds a family of hash functions in which the possibility of a collision for each function is higher for features that are close to each other than far apart features. Second, given a query feature, LSH applies various hashing functions from the created family and finds the nearest neighbors by hashing this feature. Finally, elements saved in buckets indexed by the query feature are retrieved [47, 48].

P-stable LSH (PLSH) is an extension of LSH. It can be employed in d-dimensional Euclidean space and has a better query response [49]. By considering domain S of the points set with distance measure D, an LSH family is defined as:

Definition 1 An LSH family $H = \{h : S \to U\}$ is called (r, cr, p_1, p_2)-sensitive for D if for any $v, q \in S$

$$\begin{cases} \text{if } v \in B(q, r) \text{ then } P_r\big[h(q) = h(v)\big] \geq p_1 \\ \text{if } v \notin B(q, cr) \text{ then } P_r\big[h(q) = h(v)\big] \leq p_2 \end{cases}$$

where $p_1 \geq p_2$, $c = 1 + \varepsilon$, and $B(v, r) = \{q \in X | D(v, q) \leq r\}$.

The process of making a hash table is the process of applying a hash operation to each vector. Local feature hash tables in P-stable LSH are made using three variables a, b and r. a $(r, cr, p_1, p_2)-$ sensitive family based on a 2-stable distribution is built, and we define each hash function using $h_j(v) = \left\lfloor \frac{a.v+b}{r} \right\rfloor$, where v is a feature point, $j = 1, \dots, n$ and a is a d-dimensional vector with entries selected randomly and independently from a stable distribution. b is a real number selected uniformly from the range $[0, r]$. First, the high-dimension vector v is projected onto a real line which is cleaved into equal parts of size r. Then, regarding the segment it projects onto, the vector v is assigned a hash value. The probability of two vectors v_1 and v_2 colliding under a randomly assigned hash function from this family is computed using:

$$P_{a,b}\left[h_{a,b}(v) = h_{a,b}(q)\right] = \int_0^r \frac{1}{||v-q||_p} f_p\left(\frac{x}{||v-q||_p}\right)\left(1 - \frac{x}{r}\right) d_x \tag{3}$$

f_p denotes the probability density function of absolute value for p-stable distribution.

3.3 Image hashing in DCT domain

One of the effective algorithms for image hashing with high quality is DCT-based hashing. With regard to observation, prevailing DCT coefficients can express the visual contents of the image. Changing the magnitude of a low-frequency DCT coefficient leads to obvious changes in the image. A DCT-based hashing algorithm works on the principle of partially separating the image into various frequencies. During hashing, the less important frequencies are removed and only the most important frequencies of the image are kept [50–52].

We consider the hashing method described by Tang et al. [39] to generate robust image hashing with dominant DCT coefficients. In this work, image hashing is comprised of three steps: In the first step, we convert the input image to a normalized image using digital operations, including brightness adjustment, bilinear interpolation, and Gaussian filtering. To adjust the brightness of an image, we improve the value of all pixels to a certain extent by a constant value. The bilinear interpolation changes the image size to a standard $M \times M$. Effects on image hashing such as JPEG compression and noise contamination are alleviated by Gaussian filtering. In the second step, the image is segmented into non-overlapping blocks, then we use them to create feature matrices by DCT coefficients since they capture the structure of image blocks. Finally, a binary string is produced by compressing the feature matrices [39, 50]. As described above, the image is divided into $m \times m$ non-overlapping blocks firstly. The total number of blocks of the input image is $N = \left(\frac{M}{m}\right)^2$ by considering M as an integral multiple of. B_i is the i th block indexed top to bottom and from left to right ($1 \le i \le N$), and $B_i(j,k)$ is the value in the $(j+1)$ th row and $(k+1)$ th column of B_i. Secondly, DCT is applied to each block. A coefficient in the $(u+1)$ th row and $(v+1)$ th column is obtained by $C_i(u,v)$:

$$\begin{aligned} C_i(u,v) &= a(u)a(v) \sum_{j=0}^{m-1} \sum_{k=0}^{m-1} B_i(j,k) \cos \alpha_1 \cos \alpha_2 \\ u,v &= 0, 1, \ldots, m-1 \\ \alpha_1 &= \left[\frac{(2k+1)v\pi}{2m}\right], \alpha_2 = \left[\frac{(2j+1)u\pi}{2m}\right] \end{aligned} \tag{4}$$
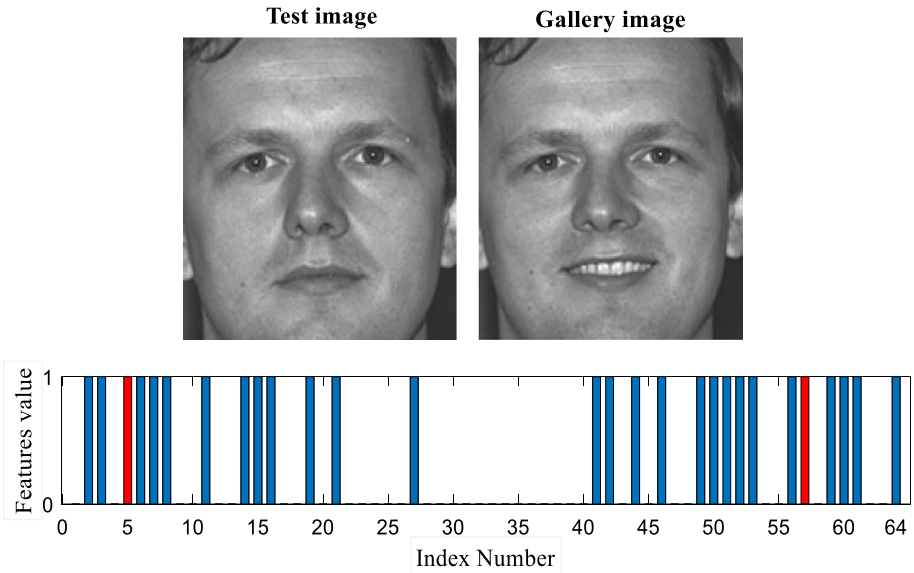
where $a(u)$ is defined as:

$$a(u) = \begin{cases} \sqrt{1/m}, & if u = 0 \\ \sqrt{2/m} & Otherwise \end{cases} \tag{5}$$

Here, DCT coefficients in the first column and first row for each block are retrieved as below:

$$\begin{aligned} C_i(1,v) &= \frac{\sqrt{2}}{m} \sum_{j=0}^{m-1} \sum_{k=0}^{m-1} B_i(j,k) cos\left[\frac{(2k+1)v\pi}{2m}\right] \\ v &= 0, \ldots, m-1 \end{aligned} \tag{6}$$

Test image   **Gallery image**

Fig.1 An example of a global binary feature vector extracted by the DCT based image hashing for two images of the same identity. The value of the features of the two images in the red columns is different

$$C_i(u, 1) = \frac{\sqrt{2}}{m} \sum_{j=0}^{m-1} \sum_{k=0}^{m-1} B_i(j, k) cos\left[\frac{(2j+1)u\pi}{2m}\right] \tag{7}$$
$$u = 0, \dots, m-1$$

The coefficients from the first column/ row to the nth element ($n \leq m$) are exploited to the generated feature matrices. To generate the hash, first, data normalization is applied to each feature matrix using the mean and standard deviation. Then, the L2 norm of these feature matrices are extracted. At the end, the binary hash is generated by quantizing all L2 norms for each block as $d = \{d_1, d_2, \dots, d_N\}$ following this rule:

$$h_i = \begin{cases} 0 & if \ d_i < T \\ 1 & Otherwise \end{cases} \tag{8}$$

where T is the mean value of the sorted result of the sequence of all L2 norms. Consequently, the image hash is defined as $H = [h_1, h_2, \dots, h_{2N}]$. Figure 1 shows an example of a global feature extracted by the DCT based image hashing. In this figure, it can be seen that the features obtained from the gallery and test images are close to each other. In fact, the hamming distance between them is very small. The value of the features of the two images in the red columns is different.

4 Proposed face recognition method

This paper proposes a novel two-stage face recognition framework for analysing different visual information based on image hashing. In order to recognize the test image, first, using a special voting system and a hashing function, we find the most similar identities to the
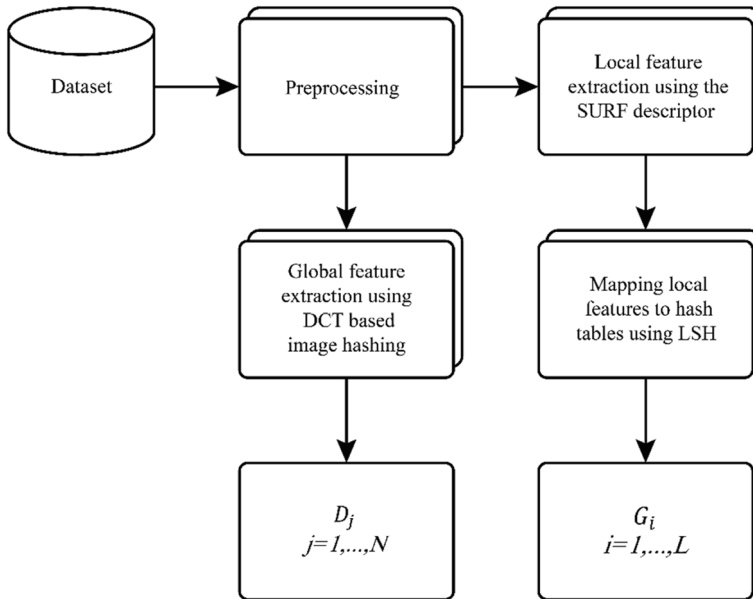
Fig. 2 Diagram of the feature extraction step in our method. Each face is hashed into a binary code and saved in D_j, $j = 1, ..., N$ for the recognition step, where N is the number of images in the dataset. Also, all local features of all images are mapped to hash tables G_i, $i = 1, ..., L$, where L is the number of hash tables

test image in terms of local visual information. Then, we recognize the test image among the best candidates according to their global visual information by employing another hashing function.

The proposed model consists of feature extraction and recognition steps. In the feature extraction step, we capture and analyse different visual information about the training faces using image hashing methods. In the recognition step, we use a new cascading framework to find the identity of the test image. We describe the feature extraction and recognition steps in the following subsections.

4.1 Feature extraction

Feature extraction plays a key role in designing a face recognition system. In our method, we analyse facial images using different visual information, locally and globally. For this purpose, we employ two feature sets extracted from the images using hashing functions. If these two features are complementary to each other, the performance of face recognition can be improved. Figure 2 shows the feature extraction step of our method. The two feature sets are extracted in parallel using image hash as stated below.

In order to analyse the local visual information, stable affine-invariant local features are obtained from the face gallery using the SURF descriptor [45]. Using the SURF, each face image is proposed by a set of local feature descriptors. However, by employing SURF, the number of extracted local features from all the images in the dataset is very high, especially when facing a large dataset, and this results in increasing the time of feature matching

during the recognition process. However, valuable image hashing can mitigate the hurdles associated with time and space complexity. Here, a simple and fast hashing framework like the PLSH descriptor works to the model's advantage. As mentioned earlier, the PLSH approximates the high-dimensional similarity search without any sub-linear dependence on the size of the data. So, to have a very fast feature matching in the recognition step, we build a set of randomized hash tables using these robust local feature vectors. The p-stable LSH scheme maps the feature vectors to a set of buckets stored in different hash tables G_i, $i = 1, ..., L$ where L is the number of hash tables. It is worth mentioning that, using the PLSH function, close local feature vectors will collide with the same buckets with high probability [34]. We use this property in the recognition step for finding those candidates in the face gallery that are similar to the test image in terms of local features, as the local features of the candidates' faces are surely similar to the test image. Although LSH functions can preserve the local relations of the data and find similarities between local features very fast, they sometimes tend to have low collision probabilities. That is why we cannot trust the result of recognizing the test face identity only using this stage. Therefore, instead of choosing one identity, we choose a small percentage of identities as the best candidates. And then we provide a second stage to analyse these candidates in more detail. Therefore, for more refinement, these best candidates should be examined again in order to find the most similar identity to the test image in terms of holistic features. As one of the aims of this paper is fast face recognition, we use a hashing function to extract the global features of a face. Here, we employ an easy and high-qualified hashing function based on DCT coefficients to analyse global visual information. Thus, global feature vectors are extracted from the face gallery using the DCT hashing algorithm, and they are kept in D_j, $j = 1, ..., N$ for the recognition step, where N is the number of images in the dataset. Finally, after analysing training faces locally and globally, test faces can be recognized.

4.2 Recognition

The overall pipeline of the recognition step is shown in Fig. 3. Given a test image, we initially adjust the test image using digital operations, including brightness adjustment and image resizing. To detect the facial area, we take advantage of the Multi-Task cascaded Convolutional Neural Network (MTCNN) algorithm in this work for its strong feature-extracting ability [53]. Then, the pre-processed image is input to the following two-stage framework:

Stage 1: In the first stage, we analyse the test image according to local visual information. We find identities whose local features are most similar to the local features of the test image. We first employ SURF to extract local features from the test image. After that, we exploit LSH tables to obtain similar candidates. As mentioned before, we use hash tables in order to increase the speed of feature matching. In this regard, each feature vector is hashed to the buckets corresponding to a set of the face gallery feature vectors. In order to choose the best similar candidates fairly, we use a special voting system.

In this voting system, the identity of each feature vector in the hashing table (G) of the gallery faces is a candidate for being the nearest neighbor to the query feature vector of the test face and will produce a vote weighed by the reverse of the normalized distance between itself and the query feature of the test image. We accumulate all votes from local
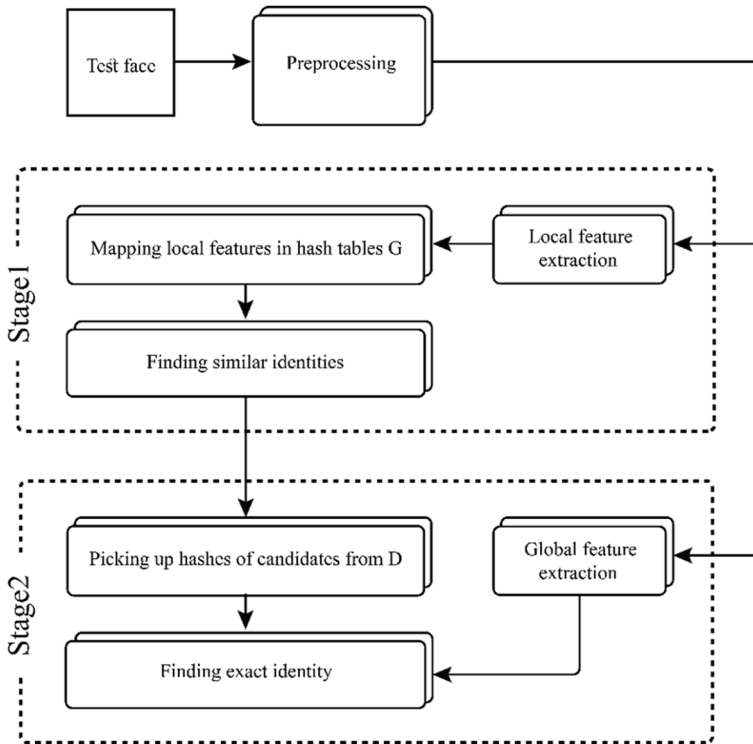
Fig.3 Overview of the recognition step in our method. In the first stage, local features are extracted and mapped to hash tables G. Then, the most similar identities are found as candidates. In the second stage, we choose a candidate whose global feature is most similar to the global feature of the test image using the saved binary codes in table D

feature vectors of the query image. Finally, the small percentage of identities with the most votes are filtered as similar candidates and go to the second stage to analyse other visual information. It is worth mentioning that if the difference between the number of votes of a candidate is remarkably more than other candidates, the second stage is not necessary, which makes the process of recognition even faster.

Stage 2: From Stage 1, we are sure that the chosen candidates are similar to the test image in regard to local features. However, this needs to see which of the candidates is similar to the test image in regard to global features as well. Robust image hashing based on DCT coefficients is employed to obtain the global feature. At this stage, the number of candidates is few and a simple approach can recognize the identity of the test face easily. Thus, given the binary hash of the remaining identities and the test image, the k-nearest neighbors (KNN) find the best matching candidate identity for the test image among the best candidates.

5 Experiments

In this section, after having introduced the datasets used in our experiments, the results are discussed in the subsequent sections.

Fig. 4 Sample images of the second subset, with illumination and pose variation, from FERET dataset

5.1 Datasets

In experiments, we evaluated the performance of our proposed method on some standard face image datasets such as FERET, ORL, AR, MUCT, PICS, FEI, and Face94. FERET has a large number of face images and the images were captured in a semi-controlled environment. This dataset contains 14,126 facial images from 1199 individuals. Images have variations in scale, the position of the face in the image, illumination, and pose. Some samples of this subset are shown in Fig. 4. The ORL dataset consists of 10 face images each of 40 different identities. In this dataset, the images were taken at different times, varying the lighting condition, facial details (glasses / no glasses), and facial expressions (smiling / not smiling, open/closed eyes). Some samples of this dataset are shown in Fig. 5. The AR dataset contains over 4000 face images corresponding to 126 people. Images were taken on two different days with various illumination conditions, facial expressions, and occlusions (scarf and sunglasses). Some samples of this dataset are shown in Fig. 6.

In the literature, a combination of different datasets has also been used, containing more images to examine the scalability of methods [54]. We have also collected images from different datasets and evaluated our proposed method with more identities. Images were collected from the FERET, MUCT, PICS, FEI, and Face94 datasets. These faces are often frontal, and only facial expressions and brightness in the images are different. The total number of identities in this collection is 1684 with two samples per person (one sample for training). Some samples of this dataset are shown in Fig. 7.

5.2 Recognition rate

In this work, we have compared our proposed method with various existing face recognition methods, including PLSH [35], ESGK [1], LDF [22], PCA [54], IKLDA + PNN [13], DMMA [17], TDL [55], KCFT [56], FFT [57], NNMF [58], LFH [40], ESRC

Fig. 5 Sample images from the ORL dataset

Fig. 6 Sample images from the AR dataset

Fig. 7 Sample images from the combinatorial dataset

Table 1 The recognition rate of different methods on ORL and AR dataset. The number of samples per person for the training step in Experiments 1 and 2 is 60% and 50%, respectively

Name of method	Experiment 1		Experiment 2	
	AR	ORL	AR	ORL
PCA + SVM [59]	-	95.63	-	92
SIFT + SURF [20]	-	95.5	-	91.4
CSGF(2D)2PCANet [44]	98.58	97.5	96.4	96.3
Alexnet [19]	-	100	-	99.17
EL-LBP [60]	98.27	97.12	97	97.15
LTP [61]	95.33	90	88.33	90
CNN [24]	-	99.74	-	98.54
Symbolic [62]	95.75	99.75	94.35	99.55
IKLDA + PNN [13]	97.46	93.95	95.86	91.44
PLSH[49]	80	94.16	78	92.33
ESGK[1]	-	97.5	-	96
Proposed method	99	100	97	100

[21], RRNN [23], and SESRC&LDF [22] PCA + SVM [59], SIFT + SURF [20], CSGF(2D)2PCANet [44], Alex Net [19], EL-LBP [60], LTP [61], CNN [24], symbolic [62]. Moreover, we have also compared our method with some of the popular deep net methods such as Alex Net [19], Google Net [27], Squeeze Net [28], Open Face [63], and Res Net [30].

As mentioned earlier, different datasets are used for different purposes. One of the most important aims of face recognition is to cope with the problem of uncontrolled conditions. In this regard, several methods have been introduced to solve this challenge. These methods have been evaluated using AR and ORL datasets as these datasets contain different lighting conditions and facial details. Like some of the methods in the literature, we have considered two different experiments for the evaluation of our method using AR and ORL datasets, each of which with two different numbers of datasets. Since the number of training images has a direct effect on the recognition rate, by considering two different experiments, the result can be more trustworthy. In the first experiment, we randomly considered 60 percent of images from each identity for the feature extraction step and the rest for the recognition step. In the second experiment, 50 percent of images were randomly used for the feature extraction step and the remaining were selected for testing. Table 1 presents the recognition rate of several approaches on the ORL and AR datasets. It shows that our method performs better than other methods with a recognition rate of 100% in both experiments on the ORL dataset, and 99% and 97% on the AR dataset. The hyphen (-) in the table means that no result was reported in the literature for the associated method in that experiment. As it is clear from the comparisons, the highest accuracy rate was achieved by our proposed method. Moreover, although some methods such as Alex Net [19] and Symbolic [62] have achieved high accuracy close to ours, they lost their stability in the second experiment and got less accuracy than our method. So, having two experiments shows that our method is more stable when the number of identities increases. Here, we achieved better results even better than the state-of-the-art methods like Symbolic [62].

Another challenge in face recognition systems is the lack of sufficient samples per person in the face gallery. Some of the methods that coped with this problem used

Table 2 Recognition rate of different methods on the first and the second subset of the FERET dataset

Method	Number of images in dataset	Number of identities	Accuracy
DMMA [17]	400	200	93
TDL [55]	400	200	93.9
KCFT [56]	400	200	93.16
Proposed	400	200	97.38
LCMOG [12]	400	200	88.83
FFT [57]	1980	990	90.04
NNMF [58]	1980	990	92.73
LFH [40]	1980	990	96.6
HOG [68]	1980	990	90.1
Proposed	1980	990	96.47

different subsets of the FERET dataset to evaluate their methods. As mentioned earlier, the FERET dataset is one of the most popular datasets in face recognition and it has different subsets that contain various challenges. As with other methods in the literature, we have considered different subsets of this dataset to evaluate our method under different conditions. The first subset contains 200 identities with 400 images and each identity has only two images that are frontal with two different expressions for each individual. The second subset is like the first subset, but it contains 990 identities with 1988 face images. Using these subsets, we can evaluate our method when there is only one training sample per person. The third subset contains 1400 face images of 200 individuals and there are more samples per person. Samples include a variety of expressions, ages, and illumination and pose variations of $\pm 15°$ and $\pm 25°$. The aim of considering this dataset is to assess our new framework while facing more challenges. We also evaluated our method when there were 4017 face images from 994 individuals. Considering the first subset of the FERET dataset, some methods, which considered a single sample per person in their experiments, have been compared with our method.

Table 2 shows the results of comparing the proposed method with these methods. As we can see, in the first subset, when the number of identities is low, our method gets a better result. But, when this number gets bigger, the accuracy of our method decreases only by one percent and still has a better result than other methods. Furthermore, although the accuracy of our method and some of the approaches in Table 2 are close, our framework can recognize faces much faster. Therefore, according to the recognition rate that can be observed in Table 2, we can conclude that our method can recognize faces very well even with only a single sample per person. Also, even when the number of identities is large and it includes more challenges, our method achieves an excellent result.

Table 3 shows the results of comparing the proposed method with methods that consider more samples per person (especially under uncontrolled conditions). As can be observed, our method has improved the performance of face recognition and obtained the highest recognition rate on the FERET dataset.

We also compared our method with some of the deep learning methods and the results can be seen in Table 4. Deep learning has made significant contributions to the face recognition systems revolution. There are, however, certain limits to deep learning and neural networks. One problem is that in order to achieve good accuracy, they need a

Table 3 Recognition rate of different methods on the third subset of the FERET dataset

Method	Number of images in dataset	Number of identities	Accuracy
ESRC [21]	1400	200	51.60
RRNN [23]	1400	200	77.90
PCA [54]	1400	200	83.16
ESGK [1]	1400	200	84.5
SESRC& LDF [22]	1400	200	93.75
LTP [61]	1400	200	80.13
Spatial [69]	1400	200	94.48
SIFT + SURF [20]	1400	200	76.8
Proposed	1400	200	94.94
Proposed	4017	994	92.86

Table 4 Recognition rate of different methods on the combined dataset

Method	Number of images in dataset	Number of identities	Accuracy
NNMF [58]	3368	1684	89.89
Alex Net [19]	3368	1684	80.16
Open Face [63]	3368	1684	53
Google Net [27]	3368	1684	65.85
Res Net [30]	3368	1684	32.06
Squeeze Net [28]	3368	1684	72.92
Proposed	3368	1684	95.90

Table 5 Parameters of the techniques used in our proposed method

Parameters	Value
L, K	5
R	2
NN	10
G	3×3
N	31
C	%65

huge volume of training data and memory. As can be seen, when the number of training samples per person is insufficient, they cannot recognize faces very well.

We have used the setup parameter values in Table 5 to implement our method. The p-stable LSH technique contains parameters L, r, K and, NN, which have been explained in Section III. As defined earlier, parameter L indicates the number of hash tables. Parameter K represents the hash length of each bucket, and parameter r illustrates the width of each bucket. Also, parameter NN shows the number of nearest neighbors (the most similar local features) considered for each entry point. In the DCT-based image hashing technique, the size of the Gaussian filter is shown with G. The value for parameter N, which is used as a threshold in the DCT coefficients, is also shown in the

Table 6 Running time of each step of recognition step in the proposed method using the first experiment on the FERET dataset

Algorithm step	Recognition time (ms)
Pre-processing	4
Extracting local features using the SURF descriptor	7
Mapping local features to LSH tables and choosing the most similar identities	9
Extracting the global feature using DCT-based image hashing	10
Recognizing the test image among the candidates	7

table. The value of C represents the optimal threshold percentage for selecting candidates for the final stage of recognition.

5.3 Running time

The above experiments indicate that the proposed method achieved a better accuracy rate than other related works. However, for face recognition, not only accuracy is important but also time and the number of parameters are important. Therefore, another crucial factor in evaluation is recognition time. In this section, we calculate the recognition time of our proposed method and compare the result with some of the related works (using the second subset of the FERET dataset). All the experiments were carried out on a 64-bit Windows operating system using an Intel Core i5 with 4 GByte RAM. All programs were implemented in MATLAB. Table 6 shows the running time of recognition per test image at different stages of the proposed method on the first dataset from FERET. As mentioned before, in some cases, there is no need to perform the second stage of recognition. It means that there is no need to extract a DCT-based image hashing algorithm. Therefore, the image recognition speed will be even faster.

In addition, sometimes, when the number of identities increases, the recognition rate considerably drops [36, 64]. Figure 8 shows the effect of increasing the number of identities on recognition time and accuracy in our method. As shown in the figure, when the number of identities increases, the recognition time increases slightly. The recognition speed of the proposed method is less than 40 ms. Therefore, the use of image hashing in the proposed method increases recognition speed. Also, in order to show the advantage of our proposed method over other deep learning methods in terms of recognition time, we compared the results in Fig. 9.

As mentioned earlier, in the first stage of the recognition step, candidates that are not similar to the test image are ignored in terms of local features. In this regard, candidates whose vote value is above C percent of the highest vote value are selected for the final stage of recognition. Also, Fig. 10 shows the effect of hash size on the accuracy of the proposed model using the FERET dataset. It is clear that when the length of the hash is 64, the recognition accuracy is 96.47%. It then increases on a gentle slope and then decreases very slightly. Therefore, a hash size of 64 is better than other sizes in terms of length and accuracy.
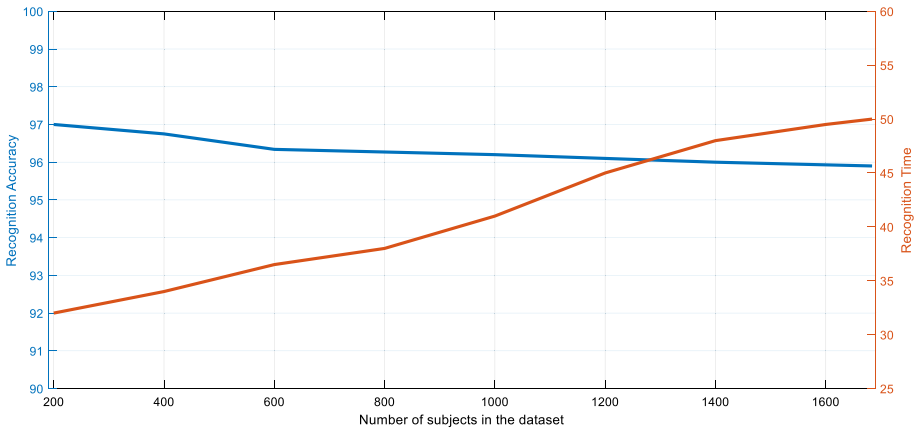
Fig. 8 Recognition time and recognition accuracy of the proposed method regarding to the number of identities in the combined dataset
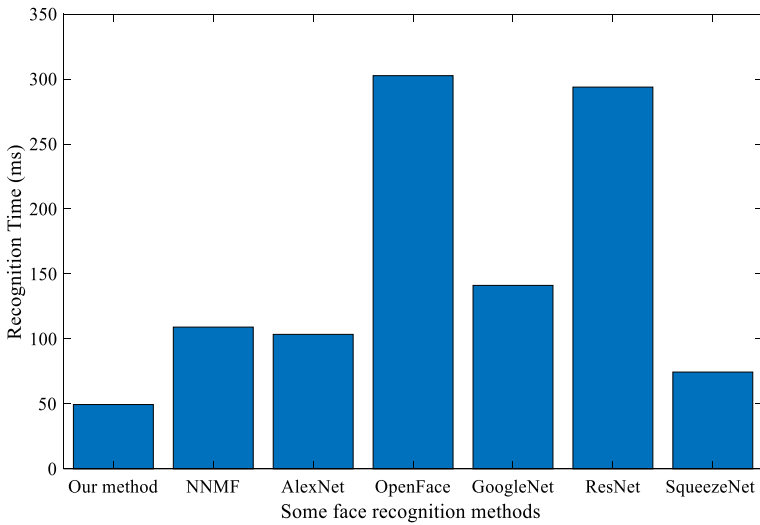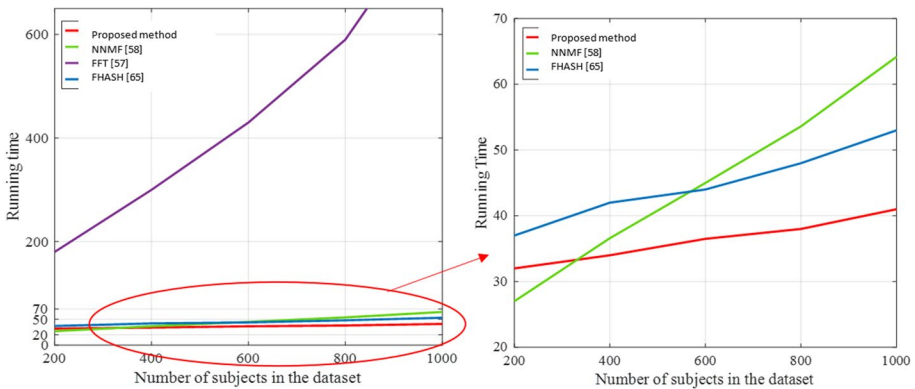

Fig. 9 Comparing the recognition time of various face recognition methods on the combined dataset

Next, the recognition time is calculated using a different number of identities in the dataset. In Fig. 11, a comparison has been made between the recognition time of the proposed method and some related methods, using the first subset of the FERET dataset. It is readily apparent that when the size of the dataset increases, the slope of reducing the recognition time of the proposed method is less than others, even in comparison with our previous work [65]. As another example, while NNMF [58] provided a recognition speed of 64.1 ms, recognition speed of the proposed method is 41 ms, but the accuracy obtained by our method is much better than NNMF [58]. From this
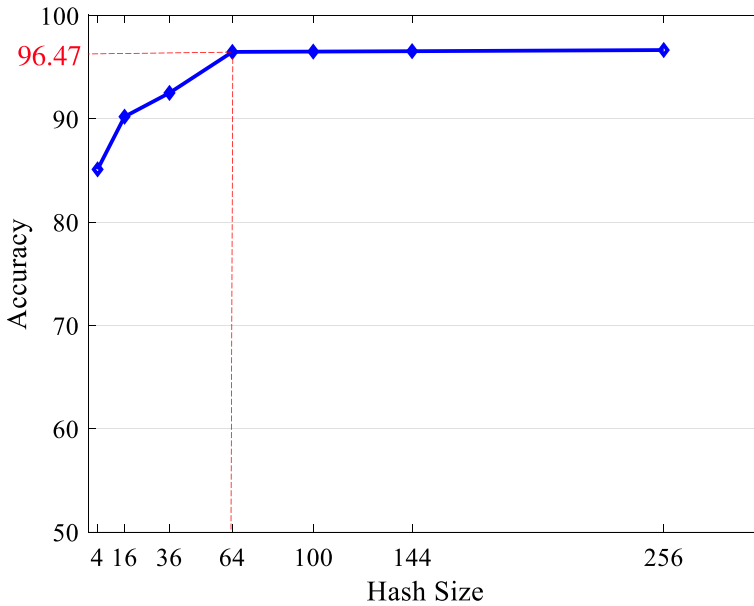
Fig. 10 The effect of different sizes of hash, which is extracted by the DCT based image hashing, on the performance of the proposed model

Fig. 11 A comparison between the recognition time of the proposed method and some related methods, considering the size of the dataset

observation, it can be concluded that the use of image hashing in the proposed method increases recognition speed.

Therefore, the use of hashing functions ensures that learning in the reduced feature space saves a significant amount of computational power without a noticeable loss of accuracy [66, 67]. Another upside to employing the hashing method is that it reduces the execution time and computational complexity of the search on large datasets [49]. Note that our framework does not depend on evaluating all identities in the face gallery.

Table 7 Results of Component Analysis

Datasets	Local Feature Analysis	Global Feature Analysis
The ORL dataset	93.75	92.5
The AR dataset	81.74	73.80
The first subset of the FERET dataset	86.5	82
The second subset of the FERET dataset	84.24	71.11
The third subset of the FERET dataset	78	68.5
The combined dataset	83.51	65.65

Therefore, if the number of identities in the gallery increases, our method will still be able to respond quickly.

5.4 Component analysis

In order to thoroughly evaluate the influence and efficiency of both local and global information, we carried out a set of ablation experiments in our suggested face recognition approach. The purpose of these experiments was to isolate and assess the contributions of each component while maintaining factors at a constant level. Table 7 provides a summary of the results.

5.4.1 Local feature analysis

In this experiment, we relied only on local features obtained through SURF and LSH, and disabled the use of global information. As it can be seen in Table 7, it is obvious that employing a local perspective alone provides good results, but it does not achieve the desired level of precision. Incorporating additional filtering in the second phase of our framework significantly improved the precision of our approach.

5.4.2 Global feature analysis

In this experiment, we exclusively relied on global information extracted through the DCT hashing algorithm, and there was no prior selection of candidates in the first step. After conducting this experiment, it becomes evident that relying solely on global information does not yield high accuracy, especially when dealing with large datasets. This highlights the importance of utilizing our voting system as the initial step in recognition.

Therefore, this part of the analysis showed that it is essential to analyze both local and global visual information since each component plays a critical role in achieving the desired outcome.

6 Conclusion

In this work, we proposed a novel framework using image hashing for face recognition. In particular, our framework leverages novel cascaded filtering with two stages of carefully designed hashing functions to recognize faces at a low computational cost. At each stage,

we ignore a large number of dissimilar identities based on different visual information. In the first stage, we analyze the local information using a hash function to cope with challenges such as pose and illumination variation. Then, we employ another hash function based on DCT to analyze identities related to global visual information. Experiments conducted in this research reveal the superiority of the proposed method to the existing ones in terms of robustness to appearance changes, scalability, and computational cost.

Data availability Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

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

Human and animal rights Authors have developed an efficient approach for Face Recognition in Large Datasets Using Image Hashing. For the experimental results, authors have considered three public datasets including FERET, ORL, and AR datasets.

References

1. Dora L, Agrawal S, Panda R, Abraham A (2017) An evolutionary single Gabor kernel based filter approach to face recognition. Eng Appl Artif Intell 62:286–301. https://doi.org/10.1016/j.engappai.2017.04.011
2. Wang M, Deng W (2021) Deep face recognition: A survey. Neurocomputing 429:215–244. https://doi.org/10.1016/j.neucom.2020.10.081
3. Modi P, Patel S (2022) A State-of-the-Art survey on face recognition method. Int J Comput Vision Image Process (IJCVIP) 12(1):1–19. https://doi.org/10.4018/IJCVIP.2022010101
4. Abdullah IA, Stephan JJ (2021) A Survey of Face Recognition Systems. Ibn AL- Haitham J Pure Appl Sci 34:144–160. https://doi.org/10.30526/34.2.2620
5. Lahasan B, Lutfi SL, San-Segundo R (2019) A survey on techniques to handle face recognition challenges: occlusion, single sample per subject and expression. Artif Intell Rev 52:949–979. https://doi.org/10.1007/s10462-017-9578-y
6. Liu W, Zhou L, Chen J (2021) Face recognition based on lightweight convolutional neural networks. Information (Switzerland) 12. https://doi.org/10.3390/info12050191
7. Yee Y, Rassem S, Mohammed T, Awang M, Suryanti (2020) Face Recognition Using Laplacian Completed Local Ternary Pattern (LapCLTP). https://doi.org/10.1007/978-981-15-1289-6_29
8. Ding C, Choi J, Tao D, Davis LS (2016) Multi-Directional Multi-Level Dual-Cross Patterns for Robust Face Recognition. IEEE Trans Pattern Anal Mach Intell 38:518–531. https://doi.org/10.1109/TPAMI.2015.2462338
9. Vishwakarma, V.P., Dalal S A novel non-linear modifier for adaptive illumination normalization for robust face recognition. Multimedia Tools Appl. https://doi.org/10.1007/s11042-019-08537-6
10. Abate AF, Nappi M, Riccio D, Sabatino G (2007) 2D and 3D face recognition: A survey. Pattern Recogn Lett 28:1885–1906. https://doi.org/10.1016/j.patrec.2006.12.018
11. Vishnu Priya R, Vijayakumar V. Tavares JMRS MQSMER: a mixed quadratic shape model with optimal fuzzy membership functions for emotion recognition. Neural Comput Appl. https://doi.org/10.1007/s00521-018-3940-0
12. Li C, Huang Y, Huang W, Qin F (2021) Learning features from covariance matrix of gabor wavelet for face recognition under adverse conditions. Pattern Recogn 119:108085. https://doi.org/10.1016/j.patcog.2021.108085
13. Ouyang A, Liu Y, Pei S et al (2020) A hybrid improved kernel LDA and PNN algorithm for efficient face recognition. Neurocomputing 393:214–222. https://doi.org/10.1016/j.neucom.2019.01.117
14. Karanwal S, Diwakar M (2021) Neighborhood and center difference-based-LBP for face recognition. Pattern Anal Appl 24:741–761. https://doi.org/10.1007/s10044-020-00948-8
15. Zhi H, Liu S (2019) Face recognition based on genetic algorithm. J Vis Commun Image Represent 58:495–502. https://doi.org/10.1016/j.jvcir.2018.12.012

16. Adjabi I, Ouahabi A, Benzaoui A, Taleb-Ahmed A (2020) Past, present, and future of face recognition: A review. Electronics (Switzerland) 9:1–53. https://doi.org/10.3390/electronics9081188

17. Lu J, Tan Y, Wang G (2011) Discriminative multi-manifold analysis for face recognition from a single training sample per person. International Conference on Computer Vision, pp 1943–1950. https://doi.org/10.1109/ICCV.2011.6126464

18. Abdelmaksoud M, Nabil E, Farag I, Hameed HA (2020) A Novel Neural Network Method for Face Recognition with a Single Sample per Person. IEEE Access 8:102212–102221. https://doi.org/10.1109/ACCESS.2020.2999030

19. Almabdy S, Elrefaei L (2019) Deep convolutional neural network-based approaches for face recognition. Appl Sci (Switzerland) 9. https://doi.org/10.3390/app9204397

20. Gupta S, Thakur K, Kumar M (2021) 2D-human face recognition using SIFT and SURF descriptors of face's feature regions. Vis Comput 37:447–456. https://doi.org/10.1007/s00371-020-01814-8

21. Deng W, Hu J, Guo J (2012) Extended SRC: Undersampled face recognition via intraclass variant dictionary. IEEE Trans Pattern Anal Mach Intell 34:1864–1870. https://doi.org/10.1109/TPAMI.2012.30

22. Liao M, Gu X (2019) Face recognition approach by subspace extended sparse representation and discriminative feature learning Mengmeng. Neurocomputing. https://doi.org/10.1016/j.neucom.2019.09.025

23. Li Y, Zheng W, Cui Z, Zhang T (2018) Face recognition based on recurrent regression neural network. Neurocomputing 297:50–58. https://doi.org/10.1016/j.neucom.2018.02.037

24. ZhangRuyang L-J (2021) Human Face Recognition Based on improved CNN Model with Multi-layers. J Korea Multimed Soc 24:701–708. https://doi.org/10.9717/kmms.2021.24.5.701

25. Guo G, Zhang N (2019) A survey on deep learning based face recognition. Comput Vis Image Underst 189:102805. https://doi.org/10.1016/j.cviu.2019.102805

26. AshutoshDhamija RBD (2022) A novel active shape model-based DeepNeural network for age invariance face recognition. J Vis Commun Image Represent. https://doi.org/10.1016/j.jvcir.2021.103393

27. Anand R, Shanthi T, Nithish M, Lakshman S (2020) Face Recognition and Classification Using GoogleNET Architecture. https://doi.org/10.1007/978-981-15-0035-0_20

28. Yee JLS, Sheikh UU, Mokji MM, Rahman SA (2020) Face Recognition and Machine Learning at the Edge. IOP Conference Series: Materials Science and Engineering 884: https://doi.org/10.1088/1757-899X/884/1/012084

29. Santoso K, Kusuma GP (2018) Face Recognition Using Modified OpenFace. Procedia Comput Sci 135:510–517. https://doi.org/10.1016/j.procs.2018.08.203

30. Ivan G, Miroslav H, Miloš Z, Alexey K (2017) Facing Face Recognition with ResNet: Round One. 67–74. https://doi.org/10.1007/978-3-319-66471-2_8

31. Luo Y, Yang Y, Shen F et al (2018) Robust discrete code modeling for supervised hashing. Pattern Recogn 75:128–135. https://doi.org/10.1016/j.patcog.2017.02.034

32. Liang X, Tang Z, Huang Z, Zhang X and Zhang S, (2023). Efficient Hashing Method Using 2D-2D PCA for Image Copy Detection. IEEE Trans Knowl Data Eng 3765–3778. https://doi.org/10.1109/TKDE.2021.3131188

33. Huang Z, Tang Z, Zhang X, Ruan L and Zhang X. Perceptual Image Hashing With Locality Preserving Projection for Copy Detection. (2023). IEEE Trans Dependable Secure Comput 463–477. https://doi.org/10.1109/TDSC.2021.3136163

34. Paulevé L, Jégou H, Amsaleg L (2010) Locality sensitive hashing: A comparison of hash function types and querying mechanisms. Pattern Recogn Lett 31:1348–1358. https://doi.org/10.1016/j.patrec.2010.04.004

35. Dehghani M, Moeini A, Kamandi A (2019) Experimental Evaluation of Local Sensitive Hashing Functions for Face Recognition. 2019 5th International Conference on Web Research, ICWR 2019 184–195. https://doi.org/10.1109/ICWR.2019.8765276

36. Dos Santos CE, Kijak E, Gravier G, Schwartz WR (2016) Partial least squares for face hashing. Neurocomputing 213:34–47. https://doi.org/10.1016/j.neucom.2016.02.083

37. Dai Q, Li J, Wang J et al (2016) A Bayesian Hashing approach and its application to face recognition. Neurocomputing 213:5–13. https://doi.org/10.1016/j.neucom.2016.05.097

38. Tang Y, Zhang X, Hu X et al (2021) Facial Expression Recognition Using Frequency Neural Network. IEEE Trans Image Process 30:444–457. https://doi.org/10.1109/TIP.2020.3037467

39. Tang Z, Yang F, Huang L, Zhang X (2014) Robust image hashing with dominant DCT coefficients. Optik 125:5102–5107. https://doi.org/10.1016/j.ijleo.2014.05.015

40. Chen J, Zu Y (2020) Local Feature Hashing with Binary Auto-Encoder for Face Recognition. IEEE Access 8:37526–37540. https://doi.org/10.1109/ACCESS.2020.2973472

41. Tang J, Li Z, Zhu X (2018) Supervised deep hashing for scalable face image retrieval. Pattern Recogn 75:1339–1351. https://doi.org/10.1016/j.patcog.2017.03.028
42. Tian L, Fan C, Ming Y (2017) Learning spherical hashing based binary codes for face recognition. Multimedia Tools Appl 76:13271–13299. https://doi.org/10.1007/s11042-016-3708-4
43. Tuncer T, Dogan S, Abdar M, Pławiak P (2020) A novel facial image recognition method based on perceptual hash using quintet triple binary pattern. Multimedia Tools Appl 79:29573–29593. https://doi.org/10.1007/s11042-020-09439-8
44. Kong J, Chen M, Jiang M et al (2018) Face Recognition Based on CSGF(2D)2PCANet. IEEE Access 6:45153–45165. https://doi.org/10.1109/ACCESS.2018.2865425
45. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-Up Robust Features (SURF). Comput Vis Image Underst 110:346–359. https://doi.org/10.1016/j.cviu.2007.09.014
46. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60:91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94
47. Indyk P, Motwani R (1998) Approximate nearest neighbors. Proceedings of the thirtieth annual ACM symposium on Theory of computing 604–613. https://doi.org/10.1145/276698.276876
48. Gionis A, Indyk P, Motwani R (1999) Similarity Search in High Dimensions via Hashing. Proceedings of the 25th International Conference on Very Large Data Bases 518–529
49. Datar M, Immorlica N, Indyk P (2004) Locality-Sensitive Hashing Scheme Based on p-Stable Distributions. Proceedings of the twentieth annual symposium on Computational geometry 253–262. https://doi.org/10.1145/997817.997857
50. Qin C, Chen X, Dong J, Zhang X (2016) Perceptual image hashing with selective sampling for salient structure features. Displays 45:26–37. https://doi.org/10.1016/j.displa.2016.09.003
51. Tang Z, Wang S, Zhang X et al (2011) Lexicographical framework for image hashing with implementation based on DCT and NMF. Multimedia Tools Appl 52:325–345. https://doi.org/10.1007/s11042-009-0437-y
52. Devi, S.K. (2017). Image Compression Using Discrete Cosine Transform (DCT) & Discrete Wavelet Transform (DWT) Techniques. Int J Res Appl Sci Eng Technol 1689–1696. https://doi.org/10.22214/ijraset.2017.10246
53. Zhang K, Zhang Z, Li Z et al (2016) (MTCNN) Multi-task Cascaded Convolutional Networks. IEEE Signal Process Lett 23:1499–1503. https://doi.org/10.48550/arXiv.1604.02878
54. Cureton EE, D'Agostino RB (2019) Face Recognition by Independent Component Analysis. Factor Anal 13:296–338. https://doi.org/10.4324/9781315799476-12
55. Zeng J, Zhao X, Gan J, et al (2018) Deep Convolutional Neural Network Used in Single Sample per Person Face Recognition. Comput Int Neurosci 2018: https://doi.org/10.1155/2018/3803627
56. Min R, Xu ZCS (2019) Single-Sample Face Recognition Based on Feature Expansion. IEEE Access 7:45219–45229. https://doi.org/10.1109/ACCESS.2019.2909039
57. Bakhshi M, Hassanpour H (2018) Spatial-frequency features extracting for facial image retrieval from a big image database. Tabriz J Electr Eng 48(2):509–517
58. Nikan F, Hassanpour H (2020) Face recognition using non-negative matrix factorization with a single sample per person in a large database. Multimedia Tools Appl. https://doi.org/10.1007/s11042-020-09394-4
59. Chen X, Song L, Qiu C (2018) Face recognition by feature extraction and classification. Proceedings of the International Conference on Anti-Counterfeiting, Security and Identification, ASID 2018-Novem:43–46. https://doi.org/10.1109/ICASID.2018.8693198
60. Truong H, Kim Y (2018) Enhanced Line Local Binary Patterns (EL-LBP) : An Efficient Image Representation for Face Recognition. Advanced Concepts for Intelligent Vision Systems. 19th International Conference, ACIVS 2018. 285–296 (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-030-01449-0_24
61. Khayam KN, Mehmood Z, Chaudhry HN et al (2022) Local-tetra-patterns for face recognition encoded on spatial pyramid matching. Comput Mater Continua 70:5039–5058. https://doi.org/10.32604/cmc.2022.019975
62. Kagawade VC, Angadi SA (2021) Savitzky-Golay filter energy features-based approach to face recognition using symbolic modeling. Pattern Anal Appl 24:1451–1473. https://doi.org/10.1007/s10044-021-00991-z
63. Serengil S, Ozpinar A. (2021) HyperExtended LightFace: A Facial Attribute Analysis Framework. International Conference on Engineering and Emerging Technologies (ICEET). 1–4, https://doi.org/10.1109/ICEET53442.2021.9659697
64. Ahmed SB, Ali SF, Ahmad J et al (2020) On the frontiers of pose invariant face recognition: a review. Artif Intell Rev 53:2571–2634. https://doi.org/10.1007/s10462-019-09742-3

65. Ghasemi M, Hassanpour H (2021) A Three-stage Filtering Approach for Face Recognition using Image Hashing. Int J Eng 34(8):1856–1864. https://doi.org/10.5829/ije.2021.34.08b.06
66. Shi Q, Li H, Shen C (2010) Rapid face recognition using hashing. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2753–2760. https://doi.org/10.1109/CVPR.2010.5540001
67. Vadlamudi LN, Vaddella VDRPV (2016) A Review Of Robust Hashing Methods For Content Based Image Authentication. i-manager's J Image Process 3:8–45. https://doi.org/10.26634/jip.3.4.8304
68. Li Y, Lu R, Huang R, Zhang W (2021) Research on Face Recognition Algorithm Based on HOG Feature. J Phys: Conf Ser 1757. https://doi.org/10.1088/1742-6596/1757/1/012099
69. Raveendra K, Ravi J (2021) Performance evaluation of face recognition system by concatenation of spatial and transformation domain features. Int J Comput Netw Inf Secur 13:47–60. https://doi.org/10.5815/ijcnis.2021.01.05