Research Article

Semantic-Based Facial Image-Retrieval System with Aid of Adaptive Particle Swarm Optimization and Squared Euclidian Distance

Manikandan Kalimuthu and Ilango Krishnamurthi

Sri Krishna College of Engineering & Technology, Coimbatore 641008, India

Correspondence should be addressed to Manikandan Kalimuthu; manikandank.prof@gmail.com

The semantic-based facial image-retrieval system is concerned with the process of retrieving facial images based on the semantic information of query images and database images. The image-retrieval systems discussed in the literature have some drawbacks that degrade the performance of facial image retrieval. To reduce the drawbacks in the existing techniques, we propose an efficient semantic-based facial image-retrieval (SFIR) system using APSO and squared Euclidian distance (SED). The proposed technique consists of three stages: feature extraction, optimization, and image retrieval. Initially, the features are extracted from the database images. Low-level features (shape, color, and texture) and high-level features (face, mouth, nose, left eye, and right eye) are the two features used in the feature-extraction process. In the second stage, a semantic gap between these features is reduced by a well-known adaptive particle swarm optimization (APSO) technique. Afterward, a squared Euclidian distance (SED) measure will be utilized to retrieve the face images that have less distance with the query image. The proposed semantic-based facial image-retrieval (SFIR) system with APSO-SED will be implemented in working platform of MATLAB, and the performance will be analyzed.

1. Introduction

Due to the popularity of digital devices and the rise of social network/photo sharing services, the availability of consumer photos is increasing [1, 2]. A large percentage of those photos contain human faces [3]. The importance and the sheer amount of human face photos make the manipulation (e.g., search and mining) of large-scale human face images an important research problem and enable many real-world applications [4]. Facial images have gained importance amongst digital images due to their use in various aspects of life, such as in airports, law enforcement applications, security systems, and automated surveillance applications [5]. Law enforcement and criminal investigation agencies typically maintain large image databases of human faces [6]. Such databases consist of faces of individuals who have either committed crimes or are suspected of having been involved in criminal activities in the past. Composite drawings are used in identifying a potential suspect from an image database [7]. Retrieval from these databases is performed in the context of the following activities: the matching of composite drawings, Bann file searching, and the ranking of a photo lineup [1].

The face-retrieval problem is concerned with retrieving facial images that are relevant to users' requests from a collection of images. Retrieving is based on the visual contents and/or information associated with this facial image [8]. The retrieval system is expected to display the images in the database that match the sketch. Bann file searching is performed when a (suspected) criminal at hand does not disclose his/her legitimate identification information to enable a law enforcement/criminal investigator to retrieve the criminal's past history [9]. Under such circumstances, the investigator visually scans the criminal's face to extract certain features and uses them in performing the retrieval. In a ranking of a photo lineup, the person performing the retrieval provides a vague and often uncertain set of features of a face and expects the system to provide a ranking of faces in the database that match the feature descriptions [10]. The victim or an eye witness of a crime describes the facial features of the perpetrator of the crime to a forensic composite technician

[11]. The human face is the most significant component of the human body that people use for face identification and verification and the recognition of others; therefore, facial images are the most common biometric characteristic used for human verification and identification [10].

The five major aspects of face study are the representation of faces, the detection of faces, the identification of faces, the analysis of facial expressions, and the classification of faces based on physical features [5]. Human attributes (e.g., gender, race, hairstyle, and hair color) are high-level semantic descriptions of a person. Using these human attributes, many researchers have achieved capable results in different applications such as face verification, identification, and keyword-based face-image retrieval [12, 13]. The retrieval task contains two types of target images. One is face images with the same identity as the query face [14]. The other is face images that have an appearance similar to the query face [7]. The description of a face given by people is almost always semantic in nature, using verbal terms such as "long face," "thick lipped," or "blonde haired." Semantic face retrieval involves the retrieval of face images based on not the raw image content but the semantics of the facial features, such as the description of a person's nose, eyes, lips, or chin. The semantic face-retrieval system is based on the semantic description of the face and then an image match of the composed image with the images in the database [15]. Face-recognition systems are in high demand in airports and other public places for automated surveillance applications [16]. Most of these abovementioned applications use the face as a hard biometric for the verification or identification of a person and consist primarily of the task of matching the actual image of a face with those stored in a database of face images [2]. However, apart from their use as a hard biometric, the "soft" traits of face modality are used to group people instead of uniquely identifying a person by his/her face. Face images have been used to identify a person's ethnicity, gender, and age [6, 17]. A more interesting application that views the face as a soft biometric is in face-retrieval systems. In many law enforcement applications, the soft biometric [12] traits of the face have to be matched to retrieve the target face image from a dataset [18].

The outline of this paper is organized as follows. Section 2 provides brief descriptions of recent related research. Section 3 reveals the motivation of the work. Section 4 presents our proposed semantic-based facial image retrieval using APSO-SED. Section 5 shows evaluations of our experimental results and discussion, and Section 6 presents the paper's conclusions.

2. Recent Related Research

Park and Jain [19] proposed a method to utilize demographic information and facial marks to improve face-image matching and retrieval performance. They developed an automatic facial mark-detection method that uses (1) the active appearance model for locating primary facial features (e.g., eyes, nose, and mouth), (2) Laplacian-of-Gaussian blob detection, and (3) morphological operators. The experimental results based on the FERET database (426 images of 213 subjects)

and two mug-shot databases from the forensic domain (1225 images of 671 subjects and 10 000 images of 10 000 subjects, resp.) showed that the use of soft biometric traits was able to improve the face-recognition performance of a state-of-the-art commercial matcher. This method does not capture the semantic aspects of a face. Humans by nature tend to use the verbal description of the semantic features (high-level features) to describe what they are looking for, and they encounter difficulty using the language of low-level features. In our system, however, we propose a new face-image-retrieval technique with semantic face aspects.

Thang et al. [6] proposed a CBIR that involves the development of a content-based facial image-retrieval system based on the constrained independent component analysis. Originating from independent component analysis CICA was a source-separation technique that uses a priori constraints to extract the desired independent components (ICs) from data. By providing query images as the constraints to the CICA, the ICs that share similar probabilistic features with the queries from the database can be extracted. Then, these extracted ICs were used to evaluate the rank of each image according to the query. The experimental results of their CBIR system tested with different facial databases showed that their system improved the retrieval performance by using a compound query, although the retrieved images had lower accuracy. The accuracy of the image retrieval will be improved in our proposed system.

Iqbal et al. [20] proposed an image-retrieval technique for biometric security purposes, which was based on color, texture, and shape features controlled by fuzzy heuristics. The proposed approach was based on the three well-known algorithms: color histogram, texture, and moment invariants. The color histogram was used to extract the color features of an image using four components: red, green, blue, and intensity. The Gabor filter was used to extract the texture features, and the Hu moment invariant was used to extract the shape features of an image. The evaluation was carried out using the standard precision and recall measures, and the results were compared with the existing approaches. The presented results showed that the proposed approach produced better results than previous methods. The study used the color and shape features in the image retrieval, but the accuracy was low. The accuracy of the image retrieval will be improved in our proposed APSO system.

Alattab and Kareem [21] proposed a method for semantic feature selection using natural language concepts. The research strategized to bridge the semantic gap between low-level image features and high-level semantic concepts. The semantic features were also integrated directly with the Eigen faces and color histogram features for facial image searching and retrieval to enhance retrieval accuracy for the user. The Euclidean distance was used for feature classes' integration and classification purposes. The proposed human facial image retrieval has been evaluated through several experiments using precision and recall methods. The results have indicated high accuracy, which was considered a significant improvement over low-level feature-based facial image-retrieval techniques. Although semantic features are used to retrieve images, there is a semantic gap in this technique. To

fill this semantic gap in the existing technique, we propose an innovative method known as adaptive particle swarm optimization (APSO).

Wang et al. [22] proposed an effective Weak Label Regularized Local Coordinate Coding (WLRLCC) technique, which exploits the principle of local coordinate coding by learning sparse features and employs the idea of graph-based weak label regularization to enhance the weak labels of similar facial images. The authors proposed an efficient optimization algorithm to solve the WLRLCC problem. They conducted extensive empirical studies on several facial image web databases to evaluate the proposed WLRLCC algorithm from different aspects. The experimental results validated its efficacy. The authors used local coordinate coding, which is a complicated optimization technique. Hence, we propose an adaptive particle swarm optimization algorithm.

3. Problem Definition

The ability to search through images is becoming increasingly important as the number of available images rises drastically. However, it is impractical for humans to label every image available and dictate which images are similar to which other images. Content-based image retrieval (CBIR) is an image-retrieval technique that uses the visual content of an image to find and retrieve the required images from databases. Normally in CBIR, facial images are different from other images because facial images are complex, multidimensional, and similar in their overall configuration. Many techniques have been proposed in the field of CBIR with facial images from face databases. In the existing methods, image-retrieval systems primarily use low-level visual features such as color, texture, and shape. Visual features are extracted automatically using image-processing methods to represent the raw content of the image. Image retrieval based on color usually yields images with similar colors, and image retrieval based on shape yields images that clearly have the same shape [23]. Therefore, systems that are used for the general purpose of image retrieval using low-level features are not effective with facial images, particularly when the user's query is a verbal description. Such descriptions do not capture the semantic aspects of a face. Humans by nature tend to use the verbal description of semantic features (high-level features) to describe what they are looking for, and they encounter difficulty using the language of low-level features. One recently developed CBIR system [19] for biometric security is applied on facial images with low-level features. This method acquired approximately 95% accurate face-image retrieval, but this method does not apply to real-time data.

Moreover, the CBIR based on low-level features does not produce accurate results in face images with the same people at different ages. Hence, the existing systems present some drawbacks in facial image retrieval when the facial images are from different ages; that is, they are not considered in the semantic features during retrieval.

4. The Proposed Semantic-Based Facial Image Retrieval Using APSO and Squared Euclidian Distance (SED)

We propose an efficient semantic-based facial image-retrieval (SFIR) system using APSO and squared Euclidian distance (SED). This technique retrieves more related images by using the APSO algorithm for the query images. The architecture of our proposed method is shown in Figure 1.

Our proposed semantic-based facial image retrieval is comprised of three stages:

(i) Feature extraction:

   (a) Low-level feature:

      (1) Shape (canny edge detection).
      (2) Texture (local binary pattern).
      (3) Color (color histogram equalization).

   (b) High-level feature:

      (1) Semantic features (face, mouth, nose, and left and right eyes).

(ii) Optimization:

   (a) Adaptive particle swarm optimization.

(iii) Image retrieval:

   (a) Squared Euclidian distance.

We considered image database I_D, where n is the total number of images in database I_D. Similarly, the query image database is defined as Q_D. m represents the total number of query images in query database Q_D. Hence,

$$I_D = \{i_1, i_2, \ldots, i_n\},$$
$$Q_D = \{q_1, q_2, \ldots, q_m\}. \tag{1}$$

Initially, the different-age face images from a larger number of persons will be collected and stored in the face database. Let $I_D = \{i_1, i_2, \ldots, i_n\}$ be a face database containing face images and let $Q_D = \{q_1, q_2, \ldots, q_m\}$ be query database images. After the face images are collected, the initial stage is the feature extraction. In the feature-extraction stage, low-level features such as the shape, texture, and color and high-level features such as the face, mouth, nose, left eye, and right eye will be extracted from the database images and query database images. Afterward, based on the features obtained, the different-age face images will be classified by generating rules, and semantic labeling will be created for those face images. To obtain better image-retrieval results, a semantic gap between these database image features and query database image features will be reduced by a well-known optimization method known as adaptive particle swarm optimization (APSO). Thus, we obtain a reduced set of features. Afterward, a squared Euclidian distance (SED) measure will be utilized to retrieve the face images that have a smaller distance from the query image.
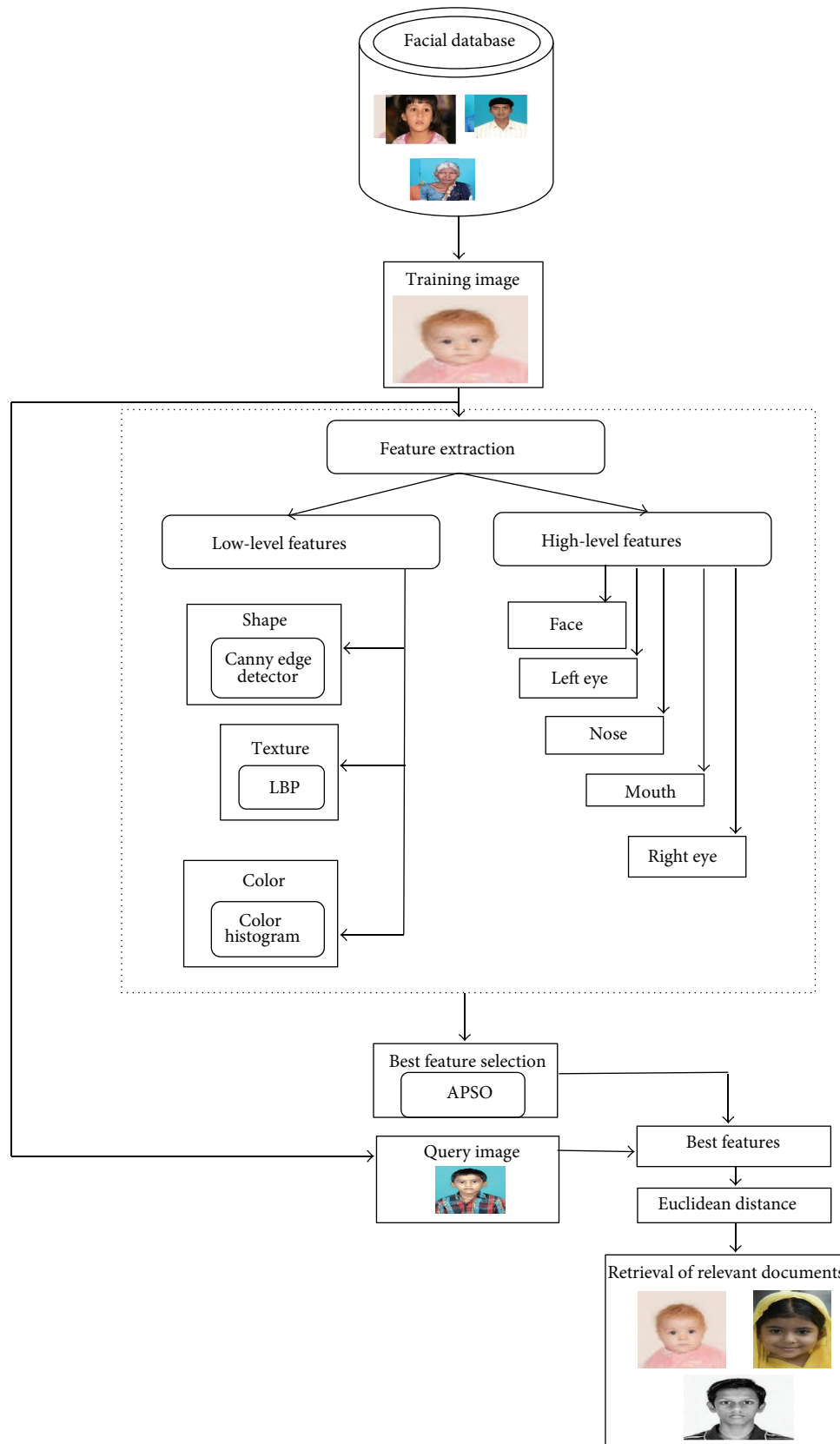
FIGURE 1: Architecture of our proposed semantic-based facial image retrieval.

4.1. Feature Extraction

(i) Low-Level Features:

      (1) Shape (canny edge detection).

      (2) Texture (local binary pattern).

      (3) Color (color histogram equalization).

4.1.1. Shape Feature Extraction Using Canny Edge-Detection Technique. This method effectively detects the edges of the given input face image (i_n) and the query image (q_m). The edges of the face images are detected using the canny edge-detection algorithm. The process of the edge-detection algorithm is given below in detail.

Step 1 (smoothing). Any noises present in the image can be filtered out by the smoothing process with the help of a Gaussian filter. The Gaussian mask is slid over the image to manipulate one square of pixels at a time and is smaller than the actual image size.

Step 2 (finding gradients). From the smoothed image, the gradients are found using the Sobel operator. In the Sobel operator, a pair of 3×3 convolution masks is utilized to find the gradients in x- and y-axis directions. The gradients in the x-axis direction are given in

$$G_X = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}. \tag{2}$$

The gradients in y-axis direction are given in

$$G_Y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{3}$$

The magnitude of the gradient is approximated using the following equation with the use of (2) and (3), which is also called the edge strength of the gradient:

$$|G| = |G_X| + |G_Y|. \tag{4}$$

The edges are difficult to find; their location and the directions of the edges can be found using

$$\theta = \arctan\left(\frac{|G_X|}{|G_Y|}\right). \tag{5}$$

Step 3 (nonmaximum suppression). To obtain the sharp edges from the blurred edges, this step is performed by considering only the local maxima of the gradients. The gradient direction θ is to be rounded nearest to $45°$ by its corresponding 8-connected neighborhood. The magnitude of the current pixel is compared with the magnitude of the pixel in the positive and negative gradient directions. If the magnitude of the current pixel is the greatest value means, then only that pixel magnitude's value is preserved; otherwise, the particular pixel magnitude value is suppressed.
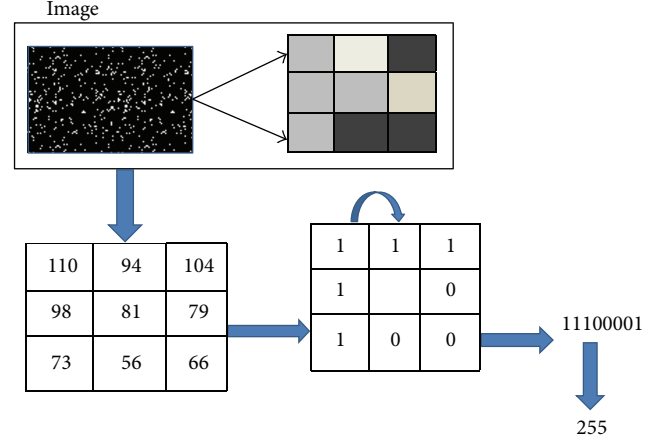
FIGURE 2: Architecture of local binary pattern.

Step 4 (thresholding). The edge pixel that remains after the nonmaximum suppression process is subjected to the thresholding process by choosing the thresholds to find the only true edges in the image. The edge pixels that are stronger than the high threshold are considered strong edge pixels, and the edge pixels that are weaker than the low threshold are suppressed. The edge pixels between these two thresholds are taken as weak edge pixels.

Step 5 (edge tracking). Edge pixels are divided into connected BLOBs (Binary Large Objects) using an 8-connected neighborhood. The BLOBs that have at least one of the strong edge pixels are preserved, and the other BLOBs without strong edge pixels are suppressed.

Thus, from the face image (i_n), we obtain the shape features of an image $\text{sh}(i_n)$, $\text{sh}(q_m)$.

4.1.2. Texture Feature Extraction Using LBP

(1) Local Binary Pattern. The LBP originally appeared as a generic descriptor. The operator assigns a label to each pixel of an image by thresholding a 3×3 neighborhood with the center pixel value and considering the result as a binary number. The resulting binary values are read clockwise starting from the top-left neighbor, as shown in Figure 2.

Given a pixel position (x_c, y_c), LBP is defined as an ordered set of binary comparisons of pixel intensities between the central pixel and its surrounding pixels. The resulting 8 bits can be expressed as follows:

$$\text{LBP}(x_c, y_c) = \sum_{n=0}^{7} s(l_n - l_c) 2^n, \tag{6}$$

where l_c corresponds to the grey value of the central pixel (x_c, y_c) and l_n corresponds to the grey value of the surrounding pixels.

Thus, using the local binary pattern texture feature $\text{tx}(i_n)$, $\text{tx}(q_m)$ has been extracted from the face image (i_n).

4.1.3. Color Feature Extraction Using Histogram

(1) Color Features. Color is one of the most widely used visual features in image retrieval [24]. Here, color features $c(i_n)$ and $c(q_m)$ are extracted by applying a histogram on the query image (q_m) and database image (i_n). A color histogram is a representation of the distribution of colors in an image. For digital images, a color histogram represents the number of pixels that have colors in each fixed list of color ranges that span the image's color space in the set of all possible colors. The histogram provides compact summarization of the distribution of data in an image. The color histogram is a statistic that can be viewed as an approximation of an underlying continuous distribution of color values.

(2) Color Histogram. Color histogram $h((i_n),(q_m)) = (h_k((i_n)),(q_m))\ k = 1,\ldots,K)$ is a K-dimensional vector such that each component $h_k((i_n),(q_m))$ represents the relative number of pixels of color C_k in the image, that is, the fraction of pixels that are most similar to the corresponding color. To build the color histogram, the image colors should be transformed to an appropriate color space and quantized according to a particular codebook of size K:

$$p_{(x)}i = \frac{n_i}{n}, \quad 0 \le i < G. \tag{7}$$

In (7), G being the total number of gray levels in the image, n_i is the number of occurrences of gray level i. n is the total number of pixels in the image, and $p_{(x)}i$ is the image histogram for pixel value i.

Computationally, the color histogram is formed by discrediting the colors within an image and counting the number of pixels of each color. The color descriptors of video can be global and local. Global descriptors specify the overall color content of the image but contain no information about the spatial distribution of these colors. Local descriptors relate to particular image regions and, in conjunction with the geometric properties of the latter, also describe the spatial arrangement of the colors. By comparing the histogram signatures of two images and matching the color content of one image with another, the color features $c(i_n)$ and $c(q_m)$ can be extracted.

(ii) High-Level Features. The most natural way people describe a face is by semantically describing facial features. Semantic face retrieval refers to retrieval of face images based on not the raw image content but the semantics of the facial features, such as the description features of the face, mouth, nose, and left and right eyes. Here, the high-level semantic features are extracted based on the pixel intensity value, which ranges according to the image taken. Here, we use the Viola-Jones algorithm for the extraction of high-level features.

The Viola-Jones algorithm utilizes a multiscale, multistage classifier that functions on image intensity information. Generally, this approach scrolls a window across the image and assigns a binary classifier that discerns between features of the face, mouth, nose, and left and right eyes and the background. This classifier is disciplined with a boosting machine learning meta-algorithm. It is credited as the fastest
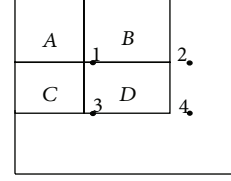

FIGURE 3: The sum of the pixels within rectangle D can be computed with four array references.

and most accurate method for faces in monocular grey-level images. Viola and Jones developed a real-time face detector comprised of a cascade of classifiers trained by AdaBoost. Each classifier exercises an integral image filter, which is reminiscent of Haar basis functions and can be processed very fast at any location and scale. This is essential to speed up the detector. At every level in the cascade, a subset of features is chosen using a feature selection procedure based on AdaBoost.

The method operates on so-called integral images: Each image element contains the sum of all pixels values to its upper left, allowing for the constant-time summation of arbitrary rectangular areas.

Integral Image. For the original image $i(x, y)$, the integral image is defined as follows:

$$ii(x, y) = \sum_{x' \le x} \sum_{y' \le y} i(x', y'). \tag{8}$$

Using the following pair recurrence,

$$s(x, y) = s(x, y - 1) + i(x, y),$$
$$ii(x, y) = ii(x - 1, y) + s(x, y) \tag{9}$$

(where $s(x, y)$ is the cumulative row sum, $s(x, -1) = 0$ plus $ii(-1, y) = 0$), the integral image can be processed in one pass over the original image. Using the integral image, any rectangular sum can be computed in four array references (see Figure 3).

The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is $A + B$, at location 3 is $A + C$, and at location 4 is $A + B + C + D$. The sum within D can be computed as $4 + 1 - (2 + 3)$. Viola-Jones' modified AdaBoost algorithm is presented in pseudocode [22] below.

Viola-Jones' algorithm is as follows:

(i) Given sample images $(x_1, y_1),\ldots,(x_n, y_n)$, where $y_1 = 0, 1$ for negative and positive examples, carry out the following.

(ii) Initialize weights $w_{1,i} = 1/2m,\ 1/2l$ for $y_1 = 0, 1$, where m and l are the numbers of positive and negative examples.

(iii) For $t = 1,\ldots,T$,

(1) normalize the weights, $w_{t,i} \leftarrow w_{t,i}/\sum_{j=1}^{n} w_{t,j}$;

(2) choose the best weak classifier in regard to the weighted error

$$\varepsilon_t = \min_{f,p,\theta} \sum_i w_i \left| h\left(x_i, f, p, \theta\right) - y_i \right|; \qquad (10)$$

(3) define $h_t(x) = h(x, f_t, p_t, \theta_t)$, where f_t, p_t, and θ_t are the minimizers of ε_i;

(4) update the weights:

$$w_{t+1,i} = w_{t,i}\beta^{1-e_i}, \qquad (11)$$

where $e_i = 0$ if example x_i is classified correctly and $e_i = 1$ otherwise and $\beta_t = \varepsilon_t/(1 - \varepsilon_t)$.

(iv) The final strong classifier is

$$C(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) \geq \dfrac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise,} \end{cases} \qquad (12)$$

where $\alpha_t = \log(1/\beta_t)$.

Thus, the semantic features of the database face images $f(i_n)$, $m(i_n)$, $n(i_n)$, $l(i_n)$, and $r(i_n)$ and query images $f(q_m)$, $m(q_m)$, $n(q_m)$, $l(q_m)$, and $r(q_m)$ are extracted based on the pixel intensity value.

4.2. Optimization. To obtain better image-retrieval results, the feature set of the query dataset will be reduced by the well-known optimization method of the computer-vision platform, adaptive particle swarm optimization (APSO). Particle swarm optimization (PSO) is a population-based search algorithm. It is created to imitate the behavior of birds in search for food on a cornfield or a fish school. The method can efficiently find optimal or near-optimal solutions in large search spaces.

Each individual particle i has a randomly initialized position $P_i = (p_i^1, p_i^2, \ldots, p_i^d)$, where p_i^d is its position in the dth dimension. The velocity $V_i = (v_i^1, v_i^2, \ldots, v_i^d)$, where v_i^d is the velocity in the dth dimension, $bp_{p_i} = (bp_i^1, bp_i^2, \ldots, bp_i^d)$, where bp_i^d is the best position in the dth dimension, and $gp = (gp^1, gp^2, \ldots, gp^d)$, where gp^d is the global best position in the dth dimension in the D-dimensional search space. Any particle can move in the direction of its personal best position to its best global position in the course of each generation. The moving process of a swarm particle in the search space is described as

$$V_i^d = V_i^d + c_1 \cdot r_1 \cdot \left(bp_i^d - p_i^d\right) + c_2 \cdot r_2 \cdot \left(gp^d - p_i^d\right), \quad (13)$$

$$p_i^d = p_i^d + \delta V_i^d. \qquad (14)$$

In (13),

c_1, c_2 are constants with the value of 2.0;

r_1, r_2 are independent random numbers generated in the range [0.1];
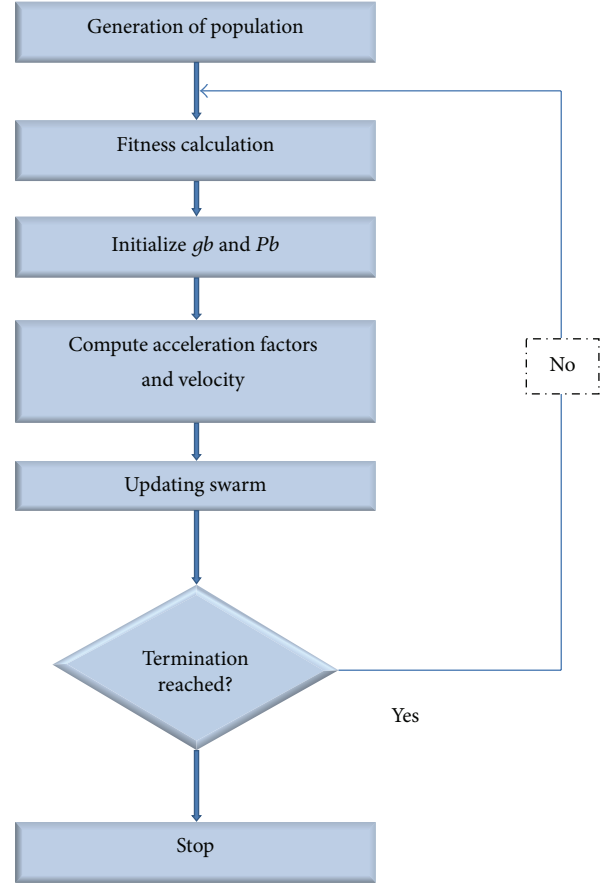
FIGURE 4: Structural design of adaptive particle swarm optimization.

V_i^d is the velocity of the ith particle;

p_i^d is the current position of particle i;

bp_i^d is the best fitness value of the particle at the current iteration;

gp^d is the best fitness value in the swarm.

The adaptive particle swarm optimization (APSO) method is used here. It provides more accurate results. The APSO acceleration coefficients are determined by

$$nc_1 = \frac{2}{3}\left(c_{1\max} - c_{1\min}\right)\left(\frac{f_{\min}}{f_{\text{avg}}} + \frac{f_{\min}}{2f_{\max}}\right) + c_{1\min}$$

$$nc_2 = \frac{2}{3}\left(c_{2\max} - c_{2\min}\right)\left(\frac{f_{\min}}{f_{\text{avg}}} + \frac{f_{\min}}{2f_{\max}}\right) + c_{2\min}, \qquad (15)$$

where $c_{1\max}$, $c_{1\min}$ are minimum and maximum values of c_1, $f_{\min}$, f_{avg}, $f_{\max}$ are minimum, average, and maximum fitness value of the particles, and $c_{2\max}$, $c_{2\min}$ are minimum and maximum values of c_2.

4.2.1. APSO in terms of Parameter Optimization Phases. Figure 4 shows the structural design of adaptive particle swarm optimization to identify the reduced set of features for

image retrieval. The steps for parameter optimization are as follows:

(i) *Generate the particle randomly*: for population size m, generate the particles randomly.

(ii) *Describe the fitness function*: choose the fitness function, which should be used for the constraints according to the current population. Here, (16) is used to calculate fitness function:

$$\text{fitness} = \min\left(pd\right)(c), \; \max\left(ls\right)(bw)\left(ps\right). \qquad (16)$$

(iii) *Initialize gp and bp*: initially, the fitness value is calculated for each particle and set as the Pbest value of each particle. The best Pbest value is selected as the gp value.

(iv) *Calculations of the acceleration factors*: the acceleration factors are computed using (15).

(v) *Velocity computation*: the new velocity is calculated using the equation below. Substituting nc_1 and nc_2 values in the velocity equation (8), the equation becomes

$$V_i^d = w^d V_i^d + nc_1 \cdot r_1 \cdot \left(bp_i^d - p_i^d\right) + nc_2 \cdot r_2$$
$$\cdot \left(gp^d - p_i^d\right). \qquad (17)$$

(vi) *Swarm update*: calculate the fitness function again and update bp and gp values. If the new value is better than the previous one, replace the old with the current one and select the best bp as gp.

(vii) *Criterion to stop*: continue until the solution is good enough or a maximum iteration is reached.

Thus, we obtained a reduced set of features from the APSO technique. This reduced set of features will be then subjected to image retrieval using the squared minimum distance calculation.

4.3. Image Retrieval

4.3.1. Distance Measurement. Distance measurement is an important stage in image retrieval. A query image is given to a system that retrieves a similar image from the image database. Our main objective is to retrieve images by measuring the distance between the reduced set of query image features that is obtained from the optimization process and the image features in the database. To retrieve the image, a squared minimum distance measure is computed. The process of the squared minimum distance measure is described below:

$$\text{dist}\left(x, y\right)(a, b) = \sqrt{\left(x - a\right)^2 + \left(y - b\right)^2}. \qquad (18)$$

Here, (x, y) are the database images and (a, b) are the query images.

By exploiting (18), the relevant images are extracted, which have the minimum value of SED. By following the aforementioned process, images similar to the query images are successfully retrieved from the image database.

5. Experimental Results

The proposed facial image-retrieval technique using APSO is implemented in the working platform of MATLAB (version 7.13) with a machine configuration as follows:

Processor: Intel Core i3.

OS: Windows 7/XP.

CPU speed: 3.20 GHz.

RAM: 4 GB.

The proposed facial image-retrieval technique with the aid of APSO is analyzed using the Adience dataset [25], the IMM face dataset [26], and the locally collected facial dataset. The facial database consists of a total of 961 images, which include children, middle-aged people, and old people. For training, 313 images were used; 641 images were used for the testing process.

5.1. Performance Analysis. The performance of the proposed technique is compared with PSO and GA algorithms to show the efficiency of the proposed technique. The performance is evaluated by three quantitative performance metrics:

(i) F-measure.

(ii) Precision.

(iii) Recall.

5.1.1. Precision and Recall Values. Precision is the fraction of retrieved images that are relevant to the findings, and recall in retrieval is the fraction of images relevant to the query that are successfully retrieved.

The precision and recall values are calculated using the following equations:

$$p = \frac{N_{rr}}{N_r},$$
$$R = \frac{N_{rr}}{N_d}. \qquad (19)$$

(i) N_{rr} denotes the number of relevant images retrieved.

(ii) N_r represents the total of retrieved images.

(iii) N_d is the total number of images in the database.

5.1.2. F-Measure. A measure that combines precision and recall is the traditional F-measure:

$$F = 2\left(\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}\right). \qquad (20)$$

Figures 5, 6, and 7 represent the sample input training images of children, old people, and middle-aged people, respectively.

From the input images, high-level features and low-level features are extracted, and the extracted features are given to APSO to select the optimal features. After that, the Euclidean distance is calculated between the features of the query image

FIGURE 5: Input images of children.
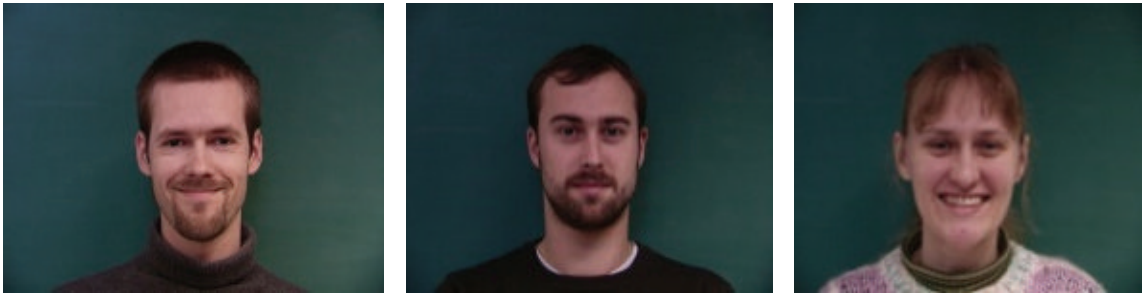

FIGURE 6: Input images of old people.

FIGURE 7: Input images of middle-aged people.

TABLE 1: Precision, recall, and F-measure values for children.

Measurements	Proposed technique	PSO	GA
Precision	0.9665	0.9402	0.9326
Recall	1	1	1
F-measure	0.4871	0.4811	0.480

and the best features. Finally, the images that are similar to the distance are retrieved.

Figure 8(a) shows the query images of children, and the corresponding retrieved images are given in Figure 8(b).

Table 1 shows the performance measures such as precision, recall, and F-measures, calculated using the statistical measures given in [27] for children.

Precision is one of the important performance measures in the image-retrieval process. Table 1 shows that the precision rate of the proposed technique is higher than that

of the other techniques. Similarly, the F-measure is higher than the other techniques. Although the difference is small, it indicates the improvement in the performance of the proposed technique.

Figure 9(a) shows the query images of middle-aged people, and the corresponding retrieved images are given in Figure 9(b). In Table 2, performance measures such as precision, recall, and F-measure are given for the proposed technique, PSO, and GA in middle-aged people image retrieval. The performance measures are higher than the other techniques, except for recall, which implies that the proposed technique performs better than the other techniques.

Figure 10(a) shows the query images of old people, and the corresponding retrieved images are given in Figure 10(b). In Table 3, the performance of the proposed technique is analyzed in old people image retrieval using precision, recall, and F-measure, and these performance measures are compared with other techniques such as PSO and GA. Compared to

(a)

(b)

FIGURE 8: (a) Query images of children and (b) retrieved images.

(a)

(b)
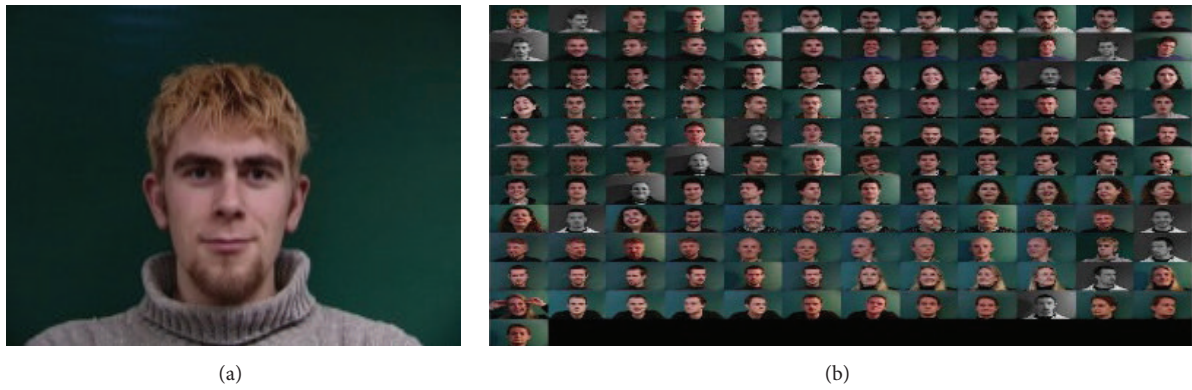
FIGURE 9: (a) Query images of middle-aged people and (b) retrieved images.

(a)

(b)
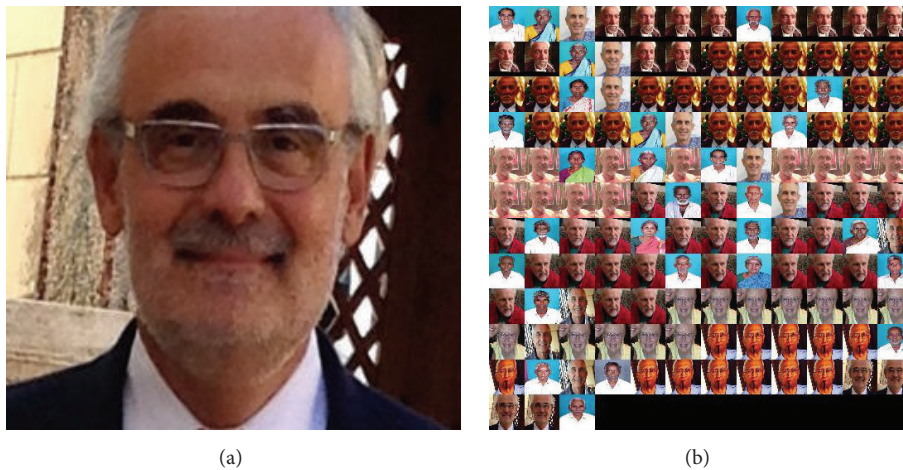
FIGURE 10: (a) Query images of old people and (b) retrieved images.

the other techniques (PSO and GA), the proposed technique has a higher rate of precision and F-measure.

Discussion. The average rate for all measures is calculated from Tables 1, 2, and 3, and a graph is drawn and given in Figure 11. The precision rate of the proposed technique is 0.9684, and the other techniques, such as PSO and GA, have precision rates of 0.9399 and 0.9248, respectively. The proposed technique has a higher precision rate than other techniques, such as PSO and GA. The variation in the

[15] S. Pange and S. Lokhande, "Image retrieval system by using CWT and support vector machines," *Signal & Image Processing*, vol. 3, no. 3, pp. 63–71, 2012.

[16] D. Depalov, T. Pappas, D. Li, and B. Gandhi, "A perceptual approach for semantic image retrieval," *American Journal of Scientific Research*, vol. 1, no. 11, pp. 6–19, 2010.

[17] V. Huddy, S. R. Schweinberger, I. Jentzsch, and A. M. Burton, "Matching faces for semantic information and names: an event-related brain potentials study," *Cognitive Brain Research*, vol. 17, no. 2, pp. 314–326, 2003.

[18] M. Balaganesh and N. Arthi, "Image retrieval using attribute enhanced sparse code words," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 3, no. 5, pp. 12580–12588, 2014.

[19] U. Park and A. K. Jain, "Face matching and retrieval using soft biometrics," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 3, pp. 406–415, 2010.

[20] K. Iqbal, M. O. Odetayo, and A. James, "Content-based image retrieval approach for biometric security using colour, texture and shape features controlled by fuzzy heuristics," *Journal of Computer and System Sciences*, vol. 78, no. 4, pp. 1258–1277, 2012.

[21] A. A. Alattab and S. A. Kareem, "Semantic features selection and representation for facial image retrieval system," in *Proceedings of the 4th International Conference on Intelligent Systems, Modelling and Simulation (ISMS '13)*, pp. 299–304, January 2013.

[22] D. Wang, S. Hoi, Y. He, J. Zhu, T. Mei, and J. Luo, "Retrieval-based face annotation by weak label regularized local coordinate coding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 550–563, 2014.

[23] G. Reddy, G. R. Babu, and P. V. Somasekhar, "Image retrieval by semantic indexing," *Journal of Theoretical and Applied Information Technology*, vol. 2, no. 1, pp. 745–750, 2010.

[24] M. Rehman, M. Iqbal, M. Sharif, and M. Raza, "Content based face image retrieval: survey," *World Applied Sciences Journal*, vol. 19, no. 3, pp. 404–412, 2012.

[25] http://www.openu.ac.il/home/hassner/Adience/data.html.

[26] http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=3160.

[27] https://en.wikipedia.org/wiki/Sensitivity_and_specificit.