ReFace: Face description and retrieval

Oshin Choudhary
Department of Computer Engineering
Army Institute of Technology
Pune, Maharashtra

Geet Bhardwaj
Department of Computer Engineering
Army Institute of Technology
Pune, Maharashtra

Pooja Bhayal
Department of Computer Engineering
Army Institute of Technology
Pune, Maharashtra

Kavita Kumari
Department of Computer Engineering
Army Institute of Technology
Pune, Maharashtra

Prof. N.K. Bansode
Department of Computer Engineering
Army Institute of Technology
Pune, Maharashtra

Abstract - The ReFace system described in this paper, presents a novel face retrieval approach based on textual description with the stored facial components extracted from different face databases. The system has two types of databases:

a. Full Face Database consisting of frontal face images collected from different face databases

b. Facial Component Database consisting of facial components extracted from the faces of Full face database.

Both the databases also contain the textual description of the face images/facial component images. Whenever user enters textual description of a face, the ReFace system will search all the similar images in the database and display them. Now user can select the best image.

I. INTRODUCTION

Face is the most important visual identity of a person and on meeting an unknown person it is the face that draws our attention most. We often describe a person in terms of the characteristic features of important face com-ponents like eyes, eyebrows, nose, and lips together with the overall shape of the face. The criminal investigation agencies employ artists to sketch human faces from the description given by an eye-witness about the criminals appearance which is then searched in the face database of known criminals, for identification. But the problems with these existing systems are that (a) they do not produce real face images and cannot be matched with a real face image using the existing automated face recognition techniques, and (b) a human expert is always needed for these systems. To overcome these problems, we have designed a new technique which displays real faces of people from the description of different eye-witnesses and does not require a human expert. This system will retrieve different face images from the database as per the description given by the eye-witness, thereby accelerating the process of identifying the criminal. The concept of generation of a new face image from textual description of facial components is new one. The first and the most important step in facial component detection is to track the position of the eyes. Thereafter, the symmetry property of the face with respect to the eyes is used for tracking rest of the components like eyebrows, lips, and nose. Splitting face into two halves eases the process further. All the faces and annotated face components are stored in the database and are retrieved according to the user given queries.

II. LITERATURE SURVEY

Describing faces textually involves a lot of steps like detection of a face in a picture, extraction of features and finally classifying them according to their attributes. Some work that is already done in all the three stages is summarized below:

A. Face Detection and Feature Extraction

Face Detection is a computer technology that deter-mines the locations and sizes of human faces in digital images. Face detection algorithms can be categorized into four parts:

a. Knowledge-based Methods - It depends on using the rules about human facial feature. It is easy to come up with simple rules to describe the features of a face and their relationships.
b. Feature-invariant Method - Algorithms that try to find invariant features of a face despite its angle or position. This approach depends on extraction of facial features that are not affected by variations in lighting conditions, pose, and other factors. These methods are classified according to the extracted features.
c. Template-matching Methods - These algorithms com-pare input images with stored patterns of faces or features. These methods try to define a face as a function.
d. Appearance-based Methods - A template matching method whose pattern database is learnt from a set of training images.

Face Feature Extraction is the extraction of facial features from the detected face like left eyebrow, right eyebrow, left eye, right eye, nose and lips. Object Detection using Haar feature-based cascade classifiers is a machine learn-ing based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. In an image, most of the image region is non-face region. So it is a better idea to have a simple method to check if a window is not a face region. If it is not, discard it in a single shot.

B. Face Feature Description

For describing extracted face features we will be refer-ring Halder and Bhattacherjees paper Fuzzy Classification of Facial Components for Face Construction[3][4]. They presented a novel type-2 Fuzzy Logic System [4] to define the shape of a facial component with crisp output. They considered 6 facial components to describe a face which were right eye, right eyebrow, left eye, left eyebrow, nose and lip. They took ratio of the facial components width and height and used those ratios to categories those features.

III. PROPOSED SYSTEM DESIGN AND IMPLEMENTA-TION

The proposed system helps the investigators by gener-ating different face images as per accounts of different eye witnesses, thereby accelerating the process of identifying the criminal. The first and the most important step in facial component detection is to track the position of the eyes. Thereafter, the symmetry property of the face with respect to the eyes is used for tracking rest of the components like eyebrows, lips, and nose. Splitting face into two halves eases the process further. All the faces and annotated face components are stored in the database and are retrieved according to the user given queries It retrieves these face images from textual description given by a witness.

The ReFace system has the following features:

1. Extraction of the components from a face image,
2. analysis of the extracted components and storage of them along with the corresponding textual labels into database for later use,
3. Handling of face queries using textual description of the face collected from different individuals.
4. Searching the required face in the database.

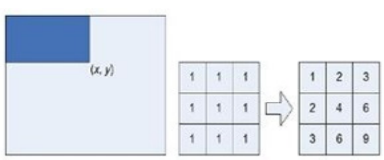

Fig. 1. Integral image generation.

The shaded region represents the sum of the pixels up to position (x, y) of the image. It shows a 33 image and its integral image representation.

A. Preprocessing

The image size is normalized and in case of blurred images, we try to reduce the noise. We will minimize the variation due to head tilt, shift, rotation, scaling, and light effect as much as possible.

IV. Face Detection and Feature Extraction

The face detection algorithm proposed by Viola and Jones is used as the basis of our design. The face detection algorithm looks for specific Haar features of a human face. When one of these features is found, the algorithm allows the face candidate to pass to the next stage of detection. A face candidate is a rectangular section of the original image called a sub-window. Generally these sub-windows have a fixed size (typically 2424 pixels). This sub-window is often scaled in order to obtain a variety of different size faces. The algorithm scans the entire image with this window and denotes each respective section a face candidate [2].

A. Integral Image: The integral image is defined as the summation of the pixel values of the original image. The value at any location (x, y) of the integral image is the sum of the images pixels above and to the left of location (x, y). Fig. 1 illustrates the integral image generation.

B. Haar Features: Haar features are composed of either two or three rectangles. Face candidates are scanned and searched for Haar features of the current stage. The weight and size of each feature and the features themselves are generated using a machine learning algorithm from AdaBoost [2].

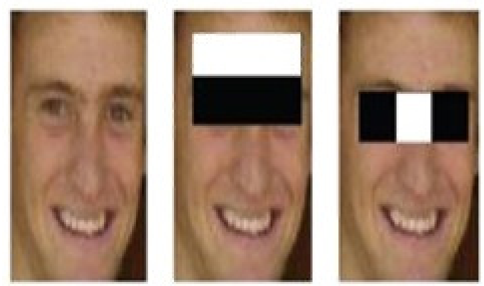

Fig. 2. Examples of Haar features. Areas of white and black regions are multiplied by their respective weights and then summed in order to get the Haar feature value.

The weights are constants generated by the learning algorithm. There are a variety of forms of features. Each Haar feature has a value that is calculated by taking the area of each rectangle, multiplying each by their respective weights, and then summing the results. The area of each rectangle is easily found using the integral image. The coordinate of the any corner of a rectangle can be used to get the sum of all the pixels above and to the left of that location using the integral image. By using each corner of a rectangle, the area can be computed quickly as denoted by figure. Since L1 is subtracted off twice it must be added back on to get the correct area of the rectangle. The area of the rectangle R, denoted as the rectangle integral, can be computed as follows using the locations of the integral image: L4-L3-L2+L1.

C. Haar Feature Classifier: A Haar feature classifier uses the rectangle integral to calculate the value of a feature. The Haar feature classifier multiplies the weight of each rectangle by its area and the results are added together. Several Haar feature classifiers compose a stage. A stage comparator sums all the Haar feature classifier results in a stage and compares this summation with a stage threshold. The threshold is also a constant obtained from the AdaBoost algorithm[2]. Each stage does not have a set number of Haar features. Depending on the parameters of the training data individual stages can have a varying number of Haar features.

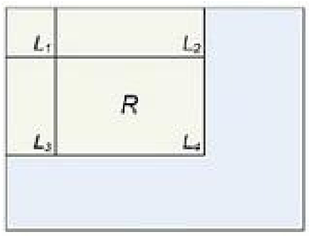

Fig. 3. Calculating the area of a rectangle R is done using the corner of the rectangle: L4-L3-L2+L1.

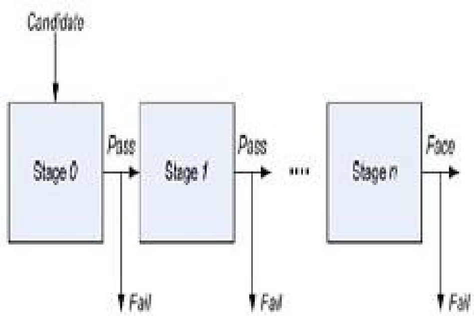

Fig. 4. Cascade of stages. Candidate must pass all stages in the cascade to be concluded as a face.

D. Cascade: The face detection algorithm eliminates face candidates quickly using a cascade of stages. The cascade eliminates candidates by making stricter requirements in each stage with later stages being much more difficult for a candidate to pass. Candidates exit the cascade if they pass all stages or fail any stage. A face is detected if a candidate passes all stages. This process is shown in Fig. 4.

V. *Face Classification*

A. Introduction

One of the main features of the ReFace system is to extract the facial components from a face image and store them in the database along with their characteristics. Suppose the Shape of a facial component FC-1 can be of three types: T1, T2 and T3. The present version of ReFace system describes a facial component FC-1 as either T1 or T2 or T3. But sometimes it is difficult to conclude that FC-1 belongs to only the set of T1 or T2 or T3. Rather, in this situation, this is more relevant to say that FC-1 is X% of T1 and Y% of T2 and Z% of T3. Moreover such type of characterization also gives more accurate classification of a component.

As a solution in this direction, we are trying to design a Fuzzy Rule Based Classifier for the analysis of the facial components after extracting them from a face image.The Fuzzy model, takes crisp value of width and height of extracted facial components and produces the crisp value of Shape for different facial components. These crisp values are stored into the database for later use.

B. Face Components And Derived Parameters

At initial level we will consider six facial components to describe a face. They are Right Eye, Right Eyebrow, Left Eye, Left Eyebrow, Nose and Lip. Each facial component has its corresponding width (W) and height (H) [1].

Components	Shape Parameters
Left Eye and Right Eye	Very Large, Large, Normal, Wide, Very Wide
Left Eyebrow and Right Eyebrow	Very Round, Round, Wavy, Flat, Very Flat
Nose	Very Narrow, Narrow, Normal, Wide, Very Wide
Lip	Very Linear, Linear, Low Linear, Wavy, Very Wavy

Fig. 5. Shape Description for Different Facial Components.

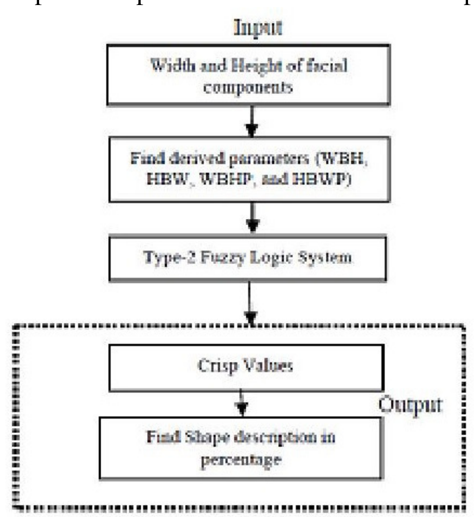

Fig. 6. The Fuzzy model

We will then compute the ratio H/W (labeled as HBW) for eyebrow, eye and lip and also calculates the value of W/H (labeled as WBH) and calculate the respective percentages. These crisp values are fed into the fuzzy system which outputs the shape description of respective facial component.

C. Fuzzy Model

The Fuzzy model takes the widths and heights of the different facial components as the input. Then it calculates the derived parameters (for eye, eyebrow, lip and WBH, WBHP for nose) for the facial components. The five values of H/W percentage for eye, eyebrow, lip and one W/H percentage value for nose are fed into the Type-2 Fuzzy Logic System which produces the crisp values of Shape. Then the model finds the Degree of Membership (DOM) values for the different parameters of Shape for the each facial component and multiplies the DOM values with 100 to find the corresponding percentage value as the Shape description of the facial components.

D. Fuzzy Logic

Type-1 fuzzy sets have certain and crisp value for their membership. While type-2 fuzzy sets do not have any certain membership and their membership is a fuzzy member. As Type-2 fuzzy system is one of the most successful techniques to deal with high uncertainty, sensitive and non-linear problem[4]. The design of novel type-2 fuzzy logic system for the classification of shape parameters for different facial components is consist of four steps, i.e. Fuzzification, Rulebase, Inference engine and Output process.

1) *Fuzzification:* The process of describing crisp val-ues into linguistic terms (fuzzy set) is called Fuzzification. Firstly the input for each facial component is fuzzified into Very Low, Low, Normal, High and Very High for the different facial components.
2) *Rulebase:* Fuzzy rules are formulated using the information gathered from experiments to map input with output. Each rule has two parts; IF part which is called antecedent and THEN part which is called consequent.

The extracted Fuzzy rules for eye are given below: Rule1: IF HBWP is Very Low Then SHAPE is Very Large.

Rule2: IF HBWP is Low Then SHAPE is Large. Rule3: IF HBWP is Normal Then SHAPE is Normal. Rule4: IF HBWP is High Then SHAPE is Wide. Rule5: IF HBWP is Very High Then SHAPE is Very Wide.

Similarly, rules are formulated for other facial features too.

3) *Inference Engine:* The fuzzy inference engine ap-plies the fuzzy rules on truth value of input variables in order to determine the corresponding output. The most commonly used methods are minimum t-norm and product t-norm. The product t-norm inference implication method is used to scale the output membership function Shape by the truth values of H/W Percentage/W/H Per-centage for each facial components.
4) *Output Process:* Firstly, the defuzzifier uses the centroid method to find the crisp value of Shape.

Secondly based on the crisp value, the system cal-culates the degree of membership values for all the parameters of each Shape for each facial component and multiplies the DOM values by 100 to find the correspond-ing percentage value.

VI. FACE RETRIEVAL

Face retrieval means retrieval of face images from the database according to the textual description given by the user. This module will use an algorithm to find similarities between the textual description entered by user and the images in the database. The various algorithm that can be used are:

A. Nearest-Neighbor Classifiers : A very simple classifier can be based on a nearest-neighbor approach. In this method, one simply finds in the N-dimensional feature space the closest object from the training set to an object being classified. Since the neighbor is nearby, it is likely to be similar to the object being classified and so is likely to be the same class as that object. Nearest neighbor methods have the advantage that they are easy to implement. They can also give quite good results if the features are chosen carefully (and if they are weighted carefully in the computation of the distance.)

There are several serious disadvantages of the nearest-neighbor methods. First, they (like the neural networks) do not simplify the distribution of objects in parameter space to a comprehensible set of parameters. Instead, the training set is retained in its entirety as a description of the object distribution. (There are some thinning methods that can be used on the training set, but the result still does not usually constitute a compact description of the object distribution.) The method is also rather slow if the training set has many examples. The most serious shortcoming of nearest neighbor methods is that they are very sensitive to the presence of irrelevant parameters.

Fig. 7. Face. Eye, Eyebrow, Nose and Lip Seperated.

Facial Components	W	H	HBWP/ WBHP	Shape
Eye-1	48	24	50.00	Wide : 00.05 % Very Wide : 99.95%
Eye-2	38	12	31.57	Very Large : 65.31% Large : 34.69%
Eye-3	27	10	37.03	Large : 57.55% Normal : 42.45%
Eye-4	39	20	51.28	Wide : 00.06% Very Wide : 99.94%

Fig. 8. Table Created According To Type-2 Fuzzy Logic

B. Decision Tree : A decision tree is a flowchart-like structure in which internal node represents a ”test” on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represents classification rules. In decision analysis a decision tree and the closely related influence diagram are used as a visual and analytical decision support tool, where the expected values (or expected utility) of competing alternatives are calculated.A decision tree consists of 3 types of nodes:

1) Decision nodes - commonly represented by squares
2) Chance nodes - represented by circles
3) End nodes - represented by triangles

C. Support Vector Machine (SVM) : A support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training data point of any class (so-called functional mar-gin), since in general the larger the margin the lower the generalization error of the classifier. Whereas the original problem may be stated in a finite dimensional space, it often happens that the sets to discriminate are not linearly separable in that space. For this reason, it was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space. To keep the computational load reasonable, the mappings used by SVM schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, by defining them in terms of a kernel function k(x,y) selected to suit the problem. The hyperplanes in the higher dimensional space are defined as the set of points whose dot product with a vector in that space is constant.

D. Principal component analysis (PCA) : Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to con-vert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components[5]. The number of principal components is less than or equal to the number of original variables. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the con-straint that it is orthogonal to (i.e., uncorrelated with) the preceding components. The principal components are orthogonal because they are the eigen vectors of the covariance matrix, which is symmetric. PCA is sensitive to the relative scaling of the original variables.

In our project, we will use SVM along with PCA(due to the reduced dimensional features offered by PCA which allow only the useful key features to participate in the classification process.) as it is more accurate and precise and works best with PCA.[7]

VII. *References*

[1] Viola, P. and Jones, M. Rapid *Object detection using boosted cascade of simple features*, IEEE Conference on Computer Vision and Pattern Recognition, 2001.

[2] M. Gopi Krishna, A. Srinivasulu, Prof (Dr.) T.K.Basak *Face De-tection System on Ada boost Algorithm Using Haar Classifiers*, International Journal of Modern Engineering Research (IJMER) www.ijmer.com Vol.2, Issue.6, pp-3996-4000 ISSN: 2249-6645, Nov-Dec. 2012.

[3] Debotosh Bhattacharjee and Santanu Halder and Mita Nasipuri and Deepak Kumar Basu and Mahantapas Kundu *Face Synthesis(FASY) System for Generation of a Face Image from Human Descrip-tion*,IEEE Conference on Industrial and Information Systems, pp 429-447, 2003.

[4] S. Halder and D. Bhattacharjee and M. Nasipuri and D. K. Basu and M. Kundu *Fuzzy Classification of Facial Components for Face Construction*, International Journal of Recent Trends in Engineering, 2009.

[5] T.P.S Ruprah *Face Recognition based on PCA algorithms*, Special issue of International Journal of Computer Science and Informatics 221-225. 2008.

[6] Meenakshi Shruti Pal and Dr Sushil Kumar Garg*Image Retrieval : A Literature Review*, International Journal of Advance Research in Computer Engineering and Technology,pp 2077-2080. 2013.

[7] Tan Chue Poh, Nur Fateha Muhamad Lani, Lai Weng Kin *Multi-Dimensional Features Reduction of PCA on SVM Classifierfor Imaging Surveillance Application*, International Journal of Systems Applications, Engineering Development. 2007.