FACE DETECTION AND RECOGNITION TECHNIQUES

Sitaram Shetkar[1], Brijesh Yeldi[2], Vishal Vasam[3], Kaushik Vishwakarma[4]
[1234] Department of Information Technology
P.V.P.P. College of Engineering
Mumbai, Maharashtra India.
[1]shetkars508@gmail.com, [2]brijesh.yeldi@gmail.com, [3]vasamvishal12@gmail.com, [4]vishwakarmakhs@gmail.com

Abstract: The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Computer Vision has been a pioneer in making things more automated and better for humans. This paper presents a study based approach for detecting human faces using the Viola Jones algorithm and recognizing the faces using Linear Binary Pattern Histogram algorithm. We train our system to automatically identify the human face on real time basic. Based on the experimental results we have discussed about the Viola -Jones algorithm and recognizing the faces using Linear Binary Pattern Histogram algorithm.

I. INTRODUCTION

Automatic human detection and recognition is an important and challenging field of research and has many application areas. The increasing use of computer vision in surveillance, replacing human beings, has initiated the research in the field of face detection. In this paper, we proposed a more efficient algorithm that consists of three intermediate steps, first is the development of a new image representation called “integral image”, which allows feature selection to be easy and rapid. Second step deals with the construction of classifiers that helps us to segregate desired features from the set of large number of features using a technique called “AdaBoost”. Third step deals with the cascading of different classifiers which was introduced in step 2 for further detailed selection of features and thereby narrowing down our search and increasing speed of face detection and recognition.

This detecting algorithm will bring practical applications like Smart captcha, Webcam based energy/power saver,

Time tracking service, Outdoor surveillance camera service.

In this paper, Section 2 describes the algorithm for face detection. Section 3 describes the algorithm for human face recognition. Section 4 and 5 describes the simulation results and conclusion respectively.

II. FACE DETECTION

Viola Jones

Viola Jones is a framework for detecting faces proposed by Paul Viola and Michael Jones in 2001. It achieves high detection rate while rapidly processing images. It gives a rate of 15 frames per second.[2]The algorithm is implemented in OpenCV.
It has four stages

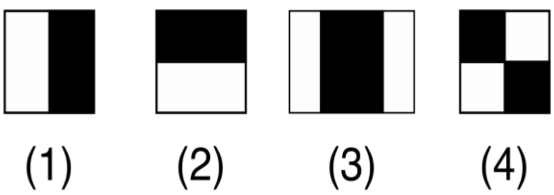
1. Haar feature Selection
2. Creating an integral image
3. AdaBoost training
4. Cascading classifiers

1. Haar Features:

Every human face share some common properties, these properties can be matched using Haar like features. Few common features of human faces are:

- Eye region is darker than the nose bridge region.
- Upper cheek region is brighter than the eye region.
- Location of eyes, mouth, nose-bridge etc.
- Value = Oriented gradients of pixel intensities.

These three features are calculated by the algorithm and are then searched in the image. Viola Jones algorithm uses a 24*24 window. It starts with one pixel per feature and matches it with the entire window. Value of each feature is calculated by subtracting the white region from the black region. Each feature gives one value. After that, two pixels are taken for each feature and is matched across whole window, it again gives one value. Same step is followed for other features also. This gives rise to about 16,000+ features each window. To optimize this number, Viola and Jones devised a method which is discussed in next section called integral image.



2. Integral image:

Integral image allows very fast feature calculation. It was introduced so that the features can be calculated very rapidly on a large set of scales. It is used to represent images, and it can be evaluated using few operations per pixel. The integral image at location at x, y contains the sum of pixels above and left to x and y inclusive. I,I Where I,i(x,y) is the integral image and (x,y) is the original image.

1	1	1
1	1	1
1	1	1

Input image

1	2	3
2	4	6
3	6	9

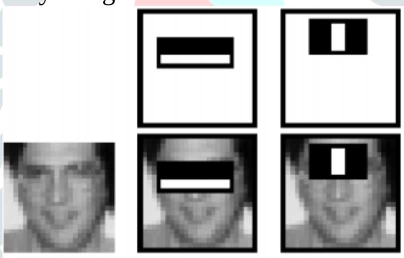
Integral image

3. AdaBoost training:

A classifier is created for classifying the image using AdaBoost. This classifier is created using a small number of features from a large set of classifier using AdaBoost. All of the features calculated are not necessarily part of the face, so the unnecessary features are discarded using this classifier. All of the 160,000+ features are reduced to few hundreds by this classifier.[1]Hence it's of prime importance that this classifier is efficient. It can be viewed as the feature selection. AdaBoost provides an effective learning algorithm and strong bound on generalization performance.

4. Cascading classifiers:

Those sub-windows which are not rejected by the initial classifier are processed by a sequence of classifiers, each slightly more complex than the last. If any classifier rejects the sub-window, no further processing is performed. The structure of the cascaded detection process is essentially that of a degenerate decision tree, and as such is related to the work of Fleuret and Geman (2001) and Amit and Geman (1999). The complete face detection cascade has 38 classifiers, which total over 80,000 operations. Nevertheless the cascade structure results in extremely rapid average detection times. On a difficult dataset, containing 507 faces and 75 million sub-windows, faces are detected using an average of 270 microprocessor instructions per sub-window. In comparison, this system is about 15 times faster than an implementation of the detection system constructed by Rowley et al. (1998). The reason for considering this algorithm first was because this algorithm was the first to process real time images, also it rejects false positive detection in very early stages. The computation time required for calculating the features is also low. Hence it provided a great means for applications extensively using face detection



$$f(x,y) = \sum_i p_b(i) - \sum_i p_w(i)$$

III. FACE REOCGNIZATION

Face recognization is an important part for the successful recognization of the images

LBHP: Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. Using the LBP combined with histograms we can represent the face images with a simple data vector.

Algorithm

Step-by-Step

Now that we know a little more about face recognition and the LBPH, let's go further and see the steps of the algorithm:

1. Parameters: the LBPH uses 4 parameters:

- Radius: the radius is used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1
- Neighbors: the number of sample points to build the circular local binary pattern. Keep in mind: the more sample points you include, the higher the computational cost. It is usually set to 8.
- Grid X: the number of cells in the horizontal direction.
- Grid Y: the number of cells in the vertical direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

2. Training the Algorithm:

First, train the algorithm. For this, we need a dataset with the facial images of the people we want to recognize. We need to also set an ID (it may be a number or the name of the person) for each image, so the algorithm will use this information to recognize an input image and give you an output. Images of the same person must have the same ID. With the training set already constructed, let's see the LBPH computational steps.

3. Applying the LBP operation:

The first computational step of the LBPH is to create an intermediate image that describes the original image in a better way, by highlighting the facial characteristics. To do so, the algorithm uses a concept of a sliding window, based on the parameters radius and neighbors.

Based on the image above, let's break it into several small steps so we can understand it easily:

- Suppose we have a facial image in grayscale.
- We can get part of this image as a window of 3x3 pixels
- It can also be represented as a 3x3 matrix containing the intensity of each pixel (0~255).
- Then, we need to take the central value of the matrix to be used as the threshold.
- This value will be used to define the new values from the 8 neighbors.
- For each neighbor of the central value (threshold), we set a new binary value.
- Now, the matrix will contain only binary values (ignoring the central value).
- Then, we convert this binary value to a decimal value and set it to the central value of the matrix, which is actually a pixel from the original image.
- At the end of this procedure (LBP procedure), we have a new image which represents better the characteristics of the original image.
- Note: The LBP procedure was expanded to use a different number of radius and neighbors, it is called Circular LBP. It can be done by using bilinear interpolation

4. Extracting the Histograms:

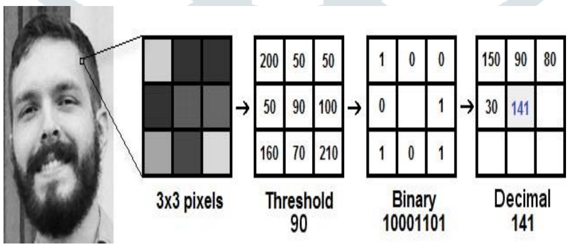
Now, using the image generated in the last step, we can use the Grid X and Grid Y parameters to divide the image into multiple grids, as can be seen in the following image: Based on the image above, we can extract the histogram of each region as follows:

- As we have an image in grayscale, each histogram (from each grid) will contain only 256 positions (0~255) representing the occurrences of each pixel intensity.
- Then, we need to concatenate each histogram to create a new and bigger histogram. The final histogram represents the characteristics of the image original image.

5. Performing the face recognition:

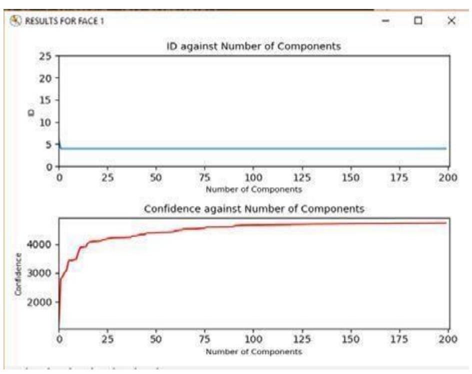
In this step, the algorithm is already trained. Each histogram created is used to represent each image from the training dataset. So, given an input image, we perform the steps again for this new image and creates a histogram which represents the image.

- So to find the image that matches the input image we just need to compare two histograms and return the image with the closest histogram.
- We can use various approaches to compare the histograms (calculate the distance between two histograms), for example: euclidean distance, chi- square, absolute value, etc. In this example, we can use the Euclidean distance (which is quite known) based on the following formula:
- So the algorithm output is the ID from the image with the closest histogram.
- We can then use a threshold and the 'confidence' to automatically estimate if the algorithm has correctly recognized the image.

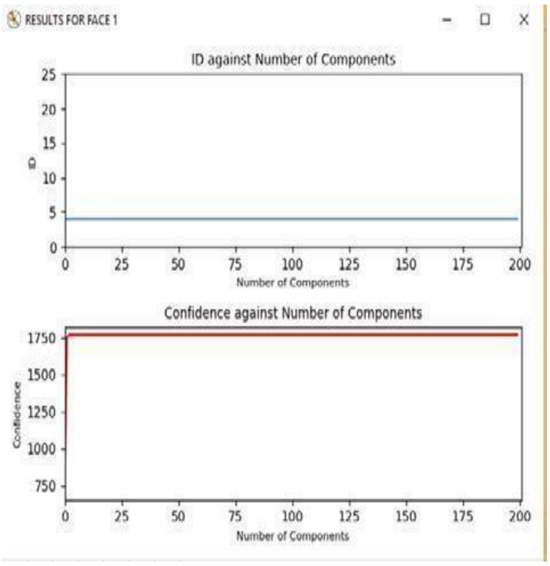


IV. RESULT

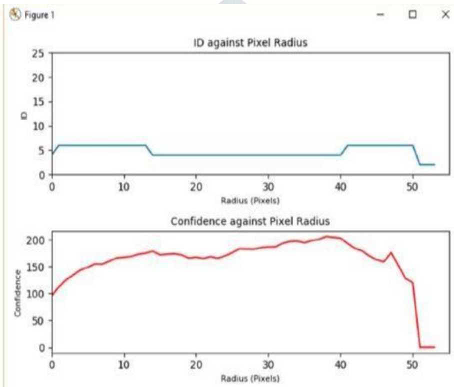
Eigen Face Recognizer



Fisher Face Recognizer



LBPH Face Recognizer



This can be seen from the table for the [3] efficiency and accuracy of the algorithms

CRITERIA	EIGEN FACE	FISHER FACE	LBPH
Confidence Factor(based on output)	2,000-3,000	100-400	2-5
Threshold	4,000	400	7
Principle of dataset generation	Component-Based	Component-Based	Pixel Based
Basic Principle	PCA	LDA	Histogram
Background Noise	Maximum	Medium	Minimum
Efficiency	Low	Higher than Eigenface	Highest

V. CONCLUSION

This paper brings altogether a new algorithm, representations and insights which are generic and may well have broader application in computer region and image processing. Finally, this paper presents a set of detailed experiments on difficult face detection and tracking data set which has been widely studied. This data set includes faces under a wide range of conditions including: illumination, scale, pose, camera, figure variation. Nevertheless the system which work under this algorithm are subjected to same set of conditions and but the algorithm is flexible enough to adjust according to the changing conditions.

REFERENCES

[1] Jatin Chatrath, Pankaj Gupta,Puneet Ahuja, Aryan Goel, Shaifali M.Arora,"REAL TIME HUMAN FACE DETECTION AND TRACKING",In proc. Of the 2014 International Conference on Signal Processing and Integrated Networks (SPIN).

[2] Ankit Srivastava, Suraj Mane, Aaditya Shah, Nirmit Shrivastava, Prof. Bhushan Thakare"A survey of face detection algorithms"In proc. Of the International Conference on Inventive Systems and Control (ICISC- 2017).

[3] SudhaNarang, Kriti Jain, MeghaSaxena, AashnaArora,"Comparison of Face Recognition Algorithms Using Opencv for Attendance System",In proc of.International Journal of Scientific and Research Publications, Volume 8, Issue 2, February 2018 268 ISSN 2250-3153

[4] 2014 International Conference on Signal Processing and Integrated Networks (SPIN) REAL TIME HUMAN FACE DETECTION AND TRACKING 978-1-4799-2866-8/14/$31.00 ©2014 IEEE.

[5] 2013 Annual IEEE India Conference (INDICON) Real- Time Human Detection and Tracking 978-1-4799-2275-8/13/$31.00 ©2013 IEEE.

[6] International Journal of Scientific and Research Publications, Volume 8, Issue 2, February 2018 268 ISSN 2250-3153 Comparison of Face Recognition Algorithms Using Opencv for Attendance System.