Journal of Engineering Sciences

Vol 13, Issue 03, MARCH /2022

ISSN NO:0377-9254

CRIMINAL IDENTIFICATION SYSTEM USING FACE DETECTION AND RECOGNITION

Rohit Alex Badana
rohitalex4.b@gmail.com

Lohith Morishetty
lohith.moris@gmail.com

Mounika Chekuri
chekurimounika544@gmail.com

Vijaya Bharathi Manjeti
vmanjeti@gitam.edu

Abhiraj Singh Chauhan
abhirajsingh1704@gmail.com

ABSTRACT: Identifying and Recognizing a criminal is a time-consuming and challenging task. According to the survey of NCRB (National Crime Records Bureau) , 80%of the same criminals do the same crimes repetitively. Criminals are becoming smarter by not leaving any biological evidence or fingerprint impressions at the crime site. The face is a unique and crucial aspect of the human body structure that recognizes a person. This Face recognition from an image may be used to identify criminals or a video frame captured by the cameras that are installed in multiple regions. As a result, we may utilize it to track down a criminal's identification. Face recognition is a biometric-based technique that mathematically maps an individual's facial traits and retains the data as a face print. It generates a unique pattern for each face and compares it to other images that are included in the collection. If a match is identified for the input face, the details linked with the relevant image will be displayed. This approach will reduce crime and protect public safety.

Keywords- *Criminal Identification; facial recognition; Haar classifier; OpenCV*

1. INTRODUCTION

The face is crucial for human identity. It is the feature which best distinguishes a person. Face recognition is an interesting and challenging problem and impacts important applications in many areas such as identification for law enforcement, authentication for banking and security system access [8], and personal identification among others. Face recognition is an easy task for humans but it's entirely different task for a computer. A very little is known about human recognition to date on How do we analyze an image and how does the brain encode it and Are inner features (eyes, nose, mouth) or outer features (head shape, hairline) used for a successful face recognition? Neurophysiologist David Hubel and Torsten Wiesel has shown that our brain has specialized nerve cells responding to specific local features of a scene, such as lines, edges, angles or movement. Since we don't see the world as scattered pieces, our visual cortex must somehow combine the different sources of information into useful patterns. Automatic face recognition is all about extracting those meaningful features from an image, putting them into a useful representation and performing some classifications on them. Face recognition based on the geometric features of a face is probably the most instinctive approach for Human identification. The whole process can be divided in three major steps where the first step is to find a good database of faces with multiple images for each individual. The next step is to detect faces in the database images and use them to train the face recognizer and the final step is to test the face recognizer to recognize faces it was trained for.

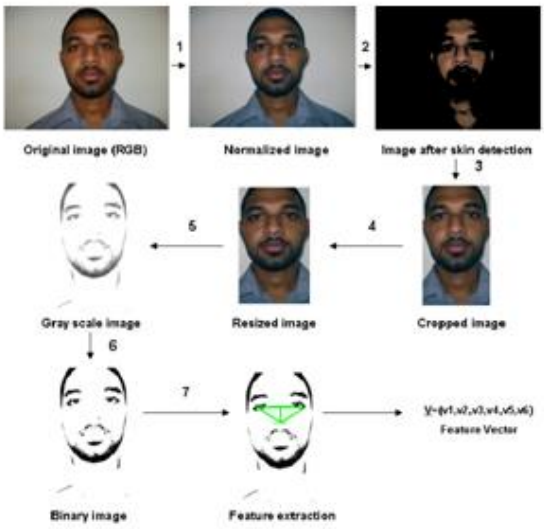

Fig.1: Example figure

Nowadays, face detection is used in many places especially the websites hosting images like Picassa, Photobucket and Facebook. The automatically tagging feature adds a new dimension to sharing pictures among the people who are in the picture and also gives the idea to other people about who the person is in the image. In our project, we have studied and implemented a pretty simple but very effective face detection algorithm which takes human skin color into account. Our aim, which we believe we have reached, was to develop a system that can be used by police or investigation department to recognize criminal from their faces. The method of face recognition used is fast, robust, reasonably simple and accurate with a relatively simple and easy to understand algorithms and technique.

2. LITERATURE REVIEW

2.1 Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection:

We develop a face recognition algorithm which is insensitive to large variation in lighting direction and facial expression. Taking a pattern classification approach, we consider each pixel in an image as a coordinate in a high-dimensional space. We take advantage of the observation that the images of a particular face, under varying illumination but fixed pose, lie in a 3D linear subspace of the high dimensional image space-if the face is a Lambertian surface without shadowing. However, since faces are not truly Lambertian surfaces and do indeed produce self-shadowing, images will deviate from this linear subspace. Rather than explicitly modeling this deviation, we linearly project the image into a subspace in a manner which discounts those regions of the face with large deviation. Our projection method is based on Fisher's linear discriminant and produces well separated classes in a low-dimensional subspace, even under severe variation in lighting and facial expressions. The eigenface technique, another method based on linearly projecting the image space to a low dimensional subspace, has similar computational requirements. Yet, extensive experimental results demonstrate that the proposed "Fisherface" method has error rates that are lower than those of the eigenface technique for tests on the Harvard and Yale face databases.

2.2 Face Recognition: Features versus templates:

Two new algorithms for computer recognition of human faces, one based on the computation of a set of geometrical features, such as nose width and length, mouth position, and chin shape, and the second based on almost-gray-level template matching, are presented. The results obtained for the testing sets show about 90% correct recognition using geometrical features and perfect recognition using template matching.

2.3 Rapid object detection using boosted cascade of simple features:

This paper describes a machine learning approach for visual object detection which is capable of processing images extremely rapidly and achieving high detection rates. This work is distinguished by three key contributions. The first is the introduction of a new image representation called the "integral image" which allows the features used by our detector to be computed very quickly. The second is a learning algorithm, based on AdaBoost, which selects a small number of critical visual features from a larger set and yields extremely efficient classifiers. The third contribution is a method for combining increasingly more complex classifiers in a "cascade" which allows background regions of the image to be quickly discarded while spending more computation on promising object-like regions. The cascade can be viewed as an object specific focus-of-attention mechanism which unlike previous approaches provides statistical guarantees that discarded regions are unlikely to contain the object of interest. In the domain of face detection the system yields detection rates comparable to the best previous systems. Used in real-time applications, the detector runs at 15 frames per second without resorting to image differencing or skin color detection.

2.4 Robust Real-time Object Detection:

This paper describes a visual object detection framework that is capable of processing images extremely rapidly while achieving high detection rates. There are three key contributions. The first is the introduction of a new image representation called the "Integral Image" which allows the features used by our detector to be computed very quickly. The second is a learning algorithm, based on AdaBoost, which selects a small number of critical visual features and yields extremely efficient classifiers. The third contribution is a method for combining classifiers in a "cascade" which allows background regions of the image to be quickly discarded while spending more computation on promising object-like regions. A set of experiments in the domain of face detection are presented. The system yields face detection performace comparable to the best previous systems. Implemented on a conventional desktop, face detection proceeds at 15 frames per second.

2.5 Fast Adaboost Training Algorithm by Dynamic Weight Trimming:

This paper presents a novel fast Adaboost training algorithm by dynamic weight trimming, which increases the training speed greatly when dealing with large datasets. At each iteration, the algorithm discards most of the samples with small weight and keeps only the samples with large weight to train the weak classifier. Then it checks the performance of the weak classifier on all the samples, if the weighted error is above 0.5, it will increase the number of training samples and retrain the weak classifier. During training, only a small portion of the samples are used to train the weak classifier, so the speed is increased greatly.

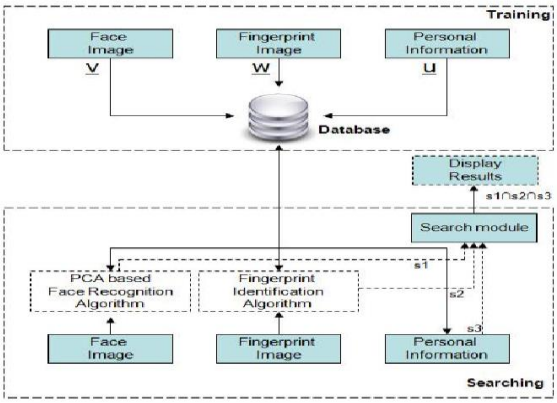

Fig.2: Face recognition system.

2.6 Feature-Based Face Recognition Using Mixture Distance:

We consider the problem of feature-based face recognition in the setting where only a single example of each face is available for training. The mixture-distance technique we introduce achieves a recognition rate of 95% on a database of 685 people in which each face is represented by 30 measured distances. This is currently the best recorded recognition rate for a feature-based system applied to a database of this size. By comparison, nearest neighbor search using Euclidean distance yields 84%. In our work a novel distance function is constructed based on local second order statistics as estimated by modeling the training data as a mixture of normal densities. We report on the results from mixtures of several sizes. We demonstrate that a flat mixture of mixtures performs as well as the best model and therefore represents an effective solution to the model selection problem. A mixture perspective is also taken for individual Gaussians to choose between first order (variance) and second order (covariance) models. Here an approximation to flat combination is proposed and seen to perform well in practice. Our results demonstrate that even in the absence of multiple training examples for each class, it is sometimes possible to infer from a statistical model of training data, a significantly improved distance function for use in pattern recognition.

2.7: Automatically Detecting Deceptive Criminal Identities:

The uncovering patterns of criminal identity deception based on actual criminal records and algorithmic approach to reveal deceptive identities are discussed. The testing results shows that no false positive errors occurs which shows the effectiveness of the algorithm. The errors occurs in the false negative category in which unrelated suspects are recognized as being related. The threshold value is set to capture maximum possible true similar records. Adaptive threshold is required for making an automated process in the future research.

2.8: Facial feature detection using AdaBoost with shape constraints:

Recently a fast and efficient face detection method has been devised [11], which relies on the AdaBoost algorithm and a set of Haar Wavelet like features. A natural extension of this approach is to use the same technique to locate individual features within the face region. However, we find that there is insufficient local structure to reliably locate each feature in every

image, and thus local models can give many false positive responses. We demonstrate that the performance of such feature detectors can be significantly improved by using global shape constraints. We describe an algorithm capable of accurately and reliably detecting facial features and present quantitative results on both high and low resolution image sets.

3. IMPLEMENTATION

The face detection algorithm proposed by Viola and Jones is used as the basis of our design [4]. The face detection algorithm looks for specific Haar features and not pixels of a human face [5]. 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 which is called as a sub-window. Generally, these sub windows have a fixed size (typically 24×24 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 [4].

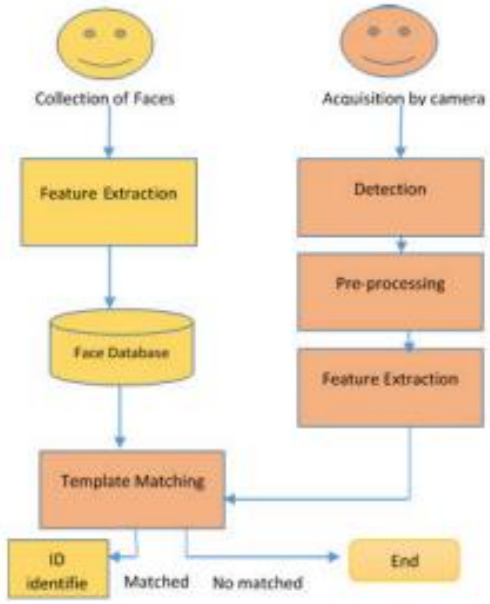

Fig.3: Architecture diagram

Import the required modules:

The Modules required to perform the facial recognition are cv2, os, image module and numpy. cv2 is the OpenCV module and contains the functions for face detection and recognition. OS will be used to maneuver with image and directory names. First, we use this module to extract the image names in the database directory and then from these names individual number is extracted, which is used as a label for the face in that image. Since, the dataset images are in gif format and as of now, OpenCV does not support gif format, Image module from PIL is used to read the image in grayscale format. Numpy arrays are used to store the images.

Load the face detection Cascade:

To Load the face detection cascade the first step is to detect the face in each image. Once we get the region of interest containing the face in the image, we use it for training the recognizer. For the purpose of face detection, we will use the Haar Cascade provided by OpenCV. The haar cascades that come with OpenCV are located in the directory of OpenCV installation. haarcascade frontalface default.xml is used for detecting the face. Cascade is loaded using the cv2 CascadeClassifier function which takes the path to the cascade xml file. if the xml file is in the current working directory, then relative path is used.

Create the Face Recognizer Object:

The next step involves creating the face recognizer object. The face recognizer object has functions like FaceRecognizer.train() to train the recognizer and FaceRecognizer.predict() to recognize a face [13]. OpenCV currently provides Eigenface Recognizer, Fisherface Recognizer and Local Binary Patterns Histograms(LBPH) Face Recognizer. We have used LBPH recognizer because Real life isn't perfect. We simply can't guarantee perfect light settings in your images or 10 different images of a person. LBPH focus on extracting local features from images. The idea is to not look at the whole image as a high-dimensional vector but describe only local features of an object. The basic idea of Local Binary Patterns is to summarize the local structure in an image by comparing each pixel with its neighbourhood. LBP operator is robust against monotonic gray scale transformations.

Prepare the training set and Perform the training:

To create the function to prepare the training set, we will define a function that takes the absolute path to the image database as input argument and returns tuple of 2 list, one containing the detected faces and the other containing the corresponding label for that face. For example, if the ith index in the list of faces represents the 4th individual in the database, then the corresponding ith location in the list of labels has value equal to 4. Now to perform the training using

the Face Recognizer. Train function. It requires 2 arguments, the features which in this case are the images of faces and the corresponding labels assigned to these faces which in this case are the individual number that we extracted from the image names.

Testing:

For testing the Face Recognizer, we check if the recognition was correct by seeing the predicted label when we bring the trained face in front of camera. The label is extracted using the os module and the string operations from the name of the sample images folder. Lower is the confidence score better is the prediction.

4. TECHNIQUES

Haar Features:

A simple rectangular Haar-like feature can be defined as the difference of the sum of pixels of areas inside the rectangle, which can be at any position and scale within the original image. This modified feature set is called 2- rectangle feature. Viola and Jones also defined 3-rectangle features and 4-rectangle features. Faces 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 [4][8]. The weights are constants generated by the learning algorithm. There are a variety of forms of features as seen below:

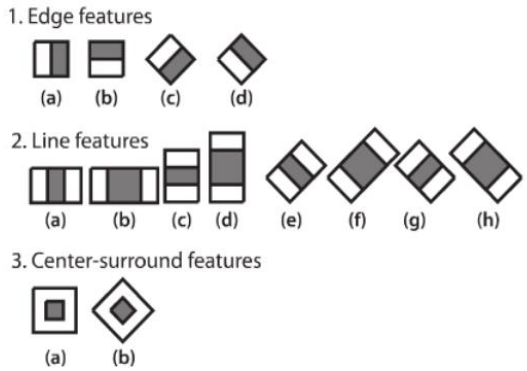

Fig.4: Common haar features

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 [3]. 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 5". Since A 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: C + A - B - D.

Fig.5: Examples of Haar features. Areas of white and black regions are multiplied by their respective weights and then summed up to get the Haar feature value.

5. EXPERIMENTAL RESULTS

Fig.6: Home screen.

Fig.7: Criminal Registration Page

Fig.8: Detect criminal page

Fig.9: Criminal Profile Page

Fig.10: Detect Criminals in Real time video

5. CONCLUSION

In this project, we are able to detect and recognize faces of the criminals in an image and in a video stream obtained from a camera in real time and raise a alarm when detected. We have used Haar feature-based cascade classifiers in OpenCV approach for face detection. It is a machine learning 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. Also, we have used Local Binary Patterns Histograms(LBPH) for face recognition. Several advantages of this algorithm are: Efficient feature selection, Scale and location invariant detector, instead of scaling the image itself, we scale the features Such a generic detection scheme can be trained for detection of other types of objects (e.g. cars, sign boards, number plates etc). LBPH recognizer can recognize faces in different lighting conditions with high accuracy. Also, LBPH can recognize efficiently even if single training image is used for each person. Our application has some disadvantages like: Detector is most effective only on frontal images of faces, it can hardly cope with 45° face rotation both around the vertical and horizontal axis.

REFERENCES

[1] Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 19, pp. 711-720. IEEE Computer Society

[2] Bornet, O. (2005, May 19). Learning Based Computer Vision with Intel's Open Source Computer Vision Library. Retrieved April 2007, 2007, from Intel.com Website: http://www.intel.com/technology/itj/2005/volume09issue02/art03_learning_vision/p04_face_dete ction.htm

[3] Brunelli, R., & Poggio, T. (1993). Face Recognition: Features versus templates. IEEE Transaction on Pattern Analysis and Machine Intelligence , 15 (10), 1042-1052.

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

[5] P. Viola and M. Jones. Robust Real-time Object Detection. International Journal of Computer Vision, 57(2):137–154,2002.

[6] https://en.wikipedia.org/wiki/Cascading_classifiers

[7] Open Computer Vision Library Reference Manual. Intel Corporation, USA, 2001.

[8] Hui-Xing, J., Yu-Jin, Z.: Fast Adaboost Training Algorithm by Dynamic Weight Trimming. Chinese Journal of Computers (2009)

[9] Bureau of Justice Statistics, U.S. Department of Justice, April 1990), pp. 43-66; SEARCH Group, “Legal and Policy Issues Relating to Biometric Identification Technologies”.

[10] J. Cox, J. Ghosn, and P.N. Yianilos, "Feature-Based Face Recognition Using Mixture Distance," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1996, pp. 209-216.

[11] G. Wang, H. Chen, and H. Atabakhsh, "Automatically Detecting Deceptive Criminal Identities," Comm. ACM, Mar. 2004, pp. 70-76.

[12] Cristinacce, D. and Cootes, T. Facial feature detection using AdaBoost with shape constraints. British Machine Vision Conference, 2003.

[13] https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html#local-binary-patterns-histograms

[14] http://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_detection.html

[15] https://aip.scitation.org/doi/pdf/10.1063/1.5005335