CRIMINAL DETECTION USING FACE RECOGNITION SYSTEM

Dr.Syed Abdul Sattar[1], Mohd Khaleel Ahmed[2]

Principa l&Professor[1], Asst.Prof[2]

Department of CSE

NAWAB SHAH ALAM KHAN COLLEGE OF ENGINEERING & TECHNOLOGY

NEW MALAKPET, HYDERABAD-500024

ABSTRACT

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 and personal identification among others. Face recognition is an easy task for humans but it's an entirely different and difficult task for a computer. 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 into 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, if it recognizes the faces it was trained with. There is an abnormal increase in the crime rate and also the number of criminals. This leads towards a great concern about the security issues. Crime prevention and criminal identification are the primary issues that police personnel face. With the advent of security technology, cameras especially CCTV have been installed in many public and private areas to provide surveillance activities. The CCTV footage can be used to identify suspects on the scene. The model will be able to recognize criminals, whose pictures the model is initially trained with, using Convolution Neural Networks, Artificial Neural Networks and OpenCV and further send a message to the cops about the location and other details of the criminal.

Keywords: Criminal Detection, Face Recognition, Artificial Neural Networks, Convolution Neural Networks, OpenCV

1. INTRODUCTION

In recent years, we've seen that there has been a marked and sustained growth in the use of Closed Circuit Television (CCTV) surveillance cameras in order to prevent crimes in public places. With the ever growing installation of advanced CCTV infrastructure, almost entire cities can now be monitored, through the major purpose served by the same is purely evidential. It would only be natural to expect an alert or warning system for ongoing (or about to happen) mishaps and crimes, where timely action can be the difference between life and death. Such scenarios are expected to be monitored and identified by personnel viewing live footage. But as the number of CCTVs per unit is keeping rising, this approach is becoming increasingly impractical. Thus what we require is a surveillance unit capable of thriving in these situations with negligible human input. We shall define a "situation of interest" or a "critical situation" as any sensitive situation that could possibly lead to the afore-mentioned predicaments. Consider the idea of a smart surveillance which would be triggered 'active' only when the statistical chances of the situation being of "interest" are high. The video feed would be recorded only under a "situation of interest" in case it needs to be documented for a legal investigation. In the response to the above trigger could be an alert to be issued to the appropriate authorities along with certain alarms which could help in preventing the situation from escalating further. So, this validates the requirement for a system which could provide smart surveillance, while ensuring privacy and confidentiality.

The surveillance camera activated for recording only when there is a situation of interest. The camera is inactive or it is not recording the video when there is no human presence. The human presence is checked using motion detection algorithm. When a crime is about to be committed, then the human is notified and an alarm system connected to the main system will be activated.

2. Literature review

Choi Woo Chul and Na JoonYeop: The orders of priority about the intelligent crime prevention technologies & system based on spatial information (e.g. Positioning System, CCTV Technology, and Integrated Management System) are constructed for integrated management in Testbed (Crime-Zero Zone) of Smart City.

[2] Yuchae Jung and Yongik Yoon: In this paper, we propose an abnormal behavioral tracking model for prediction of abnormal situation by using Expectation Maximization (EM) algorithm combined with Viterbi algorithm. The tracking model will detect objects from CCTV image in dynamic environment for the prediction of dangerous situation. This tracking system has five main steps. (1) The detection of object and their environment, (2) Feature extraction from objects and situations such as human body posture, weather, and time (3) Location information such as object trajectory and area safety level (4) knowledge update and decision making (5) prediction of abnormal situation and maximized risk rates.

[3] QianHui, Claire Chen, Li Fang and Zhang Meng: Motion detection surveillance technology came about as a relief for the generally time-consuming reviewing process that a normal video surveillance system offers. It has gained a lot of interests over the past few years. In this paper, we propose a motion detection surveillance system, consisting of its Graphic User Interface (GUI) and its method for motion detection, through the study and evaluation of currently available products and methods. The proposed system is efficient and convenient for both office and home uses.

Existing System

The existing system of CCTV monitoring is done by humans and also the automated smart CCTV monitoring systems are not fully capable of making decisions and triggering actions. The traditional passive video surveillance system proves ineffective as the number of cameras exceeds the capability of human operators to observe them. In addition to placing cameras in the 2 place of human eyes, the goal of visual surveillance is to accomplish, as automatically as possible, the surveillance task.

3. Proposed System

The proposed system deals with the current trend of security system i.e. inclusion of privacy concern, cost effectiveness, efficient alarm system. Surveillance in dynamic scenes attempts to recognize, detect and track certain objects from image sequences, and more generally to understand object behaviors. The objective is to develop an intelligent visual surveillance system for replacing the traditional passive video surveillance system.

# 3.1 System requirements

Hardware requirements

• Computer System

• CCTV cameras

• Appropriate wiring (siamese and BNC cables)

• Power cords

Software requirements:

• Windows 10 Operating System

• TensorFlow Object Detection API

• Human Motion Database

• Python 3.5

3.2 DESIGN METHODOLOGY

In this paper, we use classification method for analyzing the video and to detect anomalies. The user should provide a live webcam feed. The fed video is then converted into frames, and it is analyzed for checking abnormalities. We first check whether there are some human actions present in the frames. This checking is done with the help of the trained models. These trained models are compared with the actions present in the frame, in order to detect human presence in that frame. For this purpose of training, we require a dataset containing a set of images, and the dataset used here is HMDB51 (Human Motion Database 51). Next, we annotate these frames using a tool labeling. Since there are quite a large number of videos, all these videos can be annotated only if it is converted into frames. This is done using a program, and the code is written in OpenCV using Python. Then, after saving all the frames as xml files, the train and test directories are separated. The usual procedure for separating train and test directories is as follows: 90% of the total xmls comprises the train directory 10% of the total xmls comprises the test directory The training data is used for creating the model, and while the training data is being trained, each model from the test directory is taken and tested with each model of the test directory. From this we can get the percentage loss in accuracy. At a particular point when the loss percentage graph becomes steady, the training may be stopped.
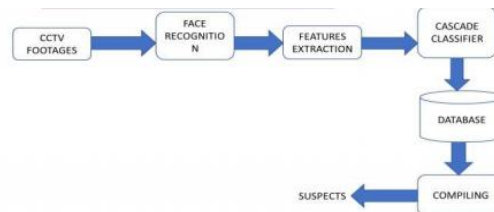

3.2 Architecture diagram of image capture and analysis
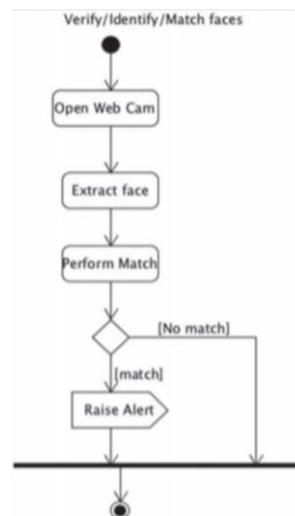
Activity diagram for criminal detection using face detection system

3.3 Haar Classifier Algorithm

The face detection algorithm looks for specific Haar features and not pixels 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 which is called as a subwindow. Generally, these sub windows have a fixed size (typically 24×24 pixels). This subwindow 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. 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 image's pixels above and to the left of location (X, Y). "Figure 1" illustrates the integral image generation.
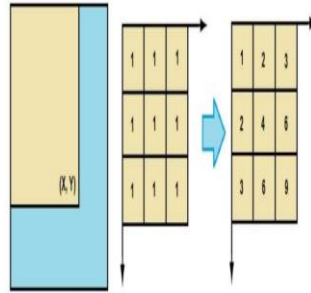
3.3 Integral image generation

IMPLEMENTATION

4.1 **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.

4.2 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. haarcascadefrontal face default.xml is used for detecting the face. Cascade is loaded using the cv2 Cascade Classifier 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.

4.3Create 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. 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. 18

4.4 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 lists, one containing the detected faces and the other containing the corresponding label for that face. For example, if the it index in the list of faces represents the 4th individual in the database, then the corresponding it 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.

4.5 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.

5. RESULTS

5.1 Interface

When the code is run, a pop-up window named "tk" appears on the screen. This window displays tabs -Input video, Add Criminal Details, Image Surveillance and Video Surveillance .the live video captured by the camera and runs analysis on this video in order to detect criminals, in any present.
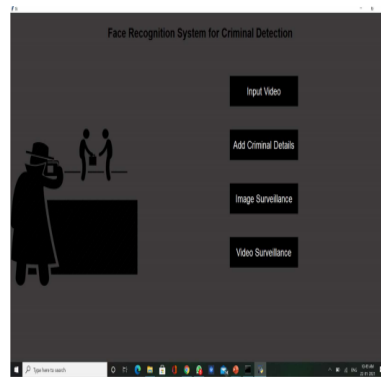

Figure 5.1: Interface

5.2 Face Detection

Based on the live video fed to the camera, it tries to detect the number of faces present in the frame and displays the count on the screen. It also recognizes any faces that it was earlier trained with, as criminals. It displays a message on the screen, within the pop-up window, along with the name of the person. The message displayed will read out that the detected person is a criminal.
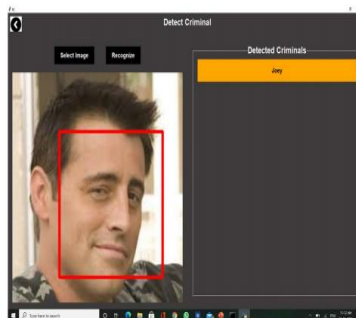

Figure 5.2.1: Model after identification of a person's face (Example 1)

The above figure 5.2.1 shows the identification of a person's face by checking the criminal records in the given dataset
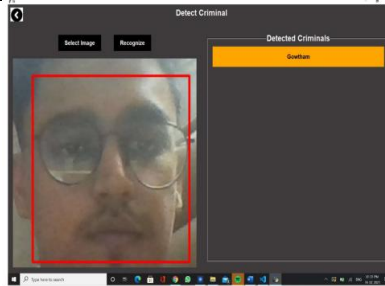
Figure 5.2.2: Model after identification of a person's face (Example 2)

The above figure 5.2.2 shows another example for the identification of a person's face by checking the criminal records in the given dataset

6. CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

N 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. 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.

6.2 Future Scope

As for the future work, a lot more testing and debugging is needed as this system was developed in a very limited time and resources. However, since it is open source software, developer can easily add new function and improve the default function. Additionally, the system can feature an image processing where the input image can be made less blurry so the system can detect face on lower quality images. Other than that, the system can use a database which contains the personal info of the person in the database, so whenever FRCI recognize a face, it will display the details about the person.

BIBLIOGRAPHY

[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.html.

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

[7] 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".

[8] *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.*

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