FACE DETECTION AND MONITORING IN MULTIPLE SURVEILLANCE FOOTAGE USING CONVOLUTIONAL NEURAL NETWORK (CNN)

Project Reference No.: *45S_BE_3469*

College ***:*** *Sapthagiri College of Engineering, Bengaluru*
Branch ***:*** *Department of Computer Science and Engineering*
Guide(s) ***:*** *Dr. Gururaj Murtugudde*
Student(S) ***:*** *Mr. Harsh P Kavatekar*
Mr. Aditya Sharma R
Mr. Aruna Kumar

Keywords: Open CV, convolutional neural network (CNN) , Gray-scale and binary histogram, Graph data structure.

Introduction:

This project describes the look and implementation of a framework that may track and determine multiple individuals in a very huddled scene captured by multiple cameras. Identifying and trailing individuals in a very huddled scene could be a complicated downside with several attainable applications. Those 2 issues will be approached severally. However once combined into one framework they will offer higher results by sharing info. Face detector is used to estimate the position of people. Those positions estimate square measure which is utilized by the face detector to prune the search area of attainable face locations and minimize the false positives. A face classifier is used to assign identities to the trajectories and to recognize the individuals within the scene. Intelligent surveillance are gaining popularity in the field of surveillance systems. Real-time detection of faces in a crowd is a key feature that would greatly improve intelligent surveillance systems. Using a high-resolution ip cameras for surveillance would greatly help in gaining the accuracy of the search algorithm. Running face recognition on multiple cameras can consume a lot of processing power and other resources. We have come up with implementation of face recognition algorithm using graphs and Process pool executor, where all the cameras are connected in graph data structure. Only the neighbouring cameras of the last found location will run the face detection process and search for the target face. This methodology of processing would not burden by consuming the high data processing rates required for multiple high-resolution video. Using this approach, we can track a person's mobility and use less processing and computing resources.

Objectives:

- To perform visual search in a highly complex environment which uses Realtime surveillance footages from multiple security closed-circuit television (CCTV) system.
- To locate the person and display the person’s current whereabouts using visual search feature on Realtime surveillance footages.
- To guide the search algorithm to perform its next visual search on its adjacent

surveillance footages based on the person's current location and the person's mobility.

- To keep live track of an individual and monitor him through surveillance.
- To find missing kids or persons.
- To make human tracking more efficient and accurate.
- To enhance Security system.
- To reduce the time taken by intelligent agencies to track down a person/ an object.

Methodology:

The Realtime surveillance footages from multiple security closed-circuit television (CCTV) systems are taken as an input to the search algorithm. The search algorithm processes these data using parallel processing. These are the Algorithms used for this project :-

In this project we present the various facial datasets that have been analyzed and evaluated to determine if the face matches the target. Obviously, we aim for the best accuracy but mathematical efficiency is important. To enhance the performance the project is still distributed within 4 modules.

1] Feature Extraction using Haar Cascades features

2] Open CV Framework

3] Process Pool Executor to run multiple cameras

4] Implementing Graph to track a person

OpenCV2 now comes with the very new FaceRecognizer magnificence for face recognition, so that you can begin experimenting with face recognition directly. It provides a library to carry out face recognition using OpenCV (with complete supply code listings) and offers you a creation into the algorithms behind. It will additionally display the way to create the visualizations you may discover in lots videos.

The currently available algorithms are:

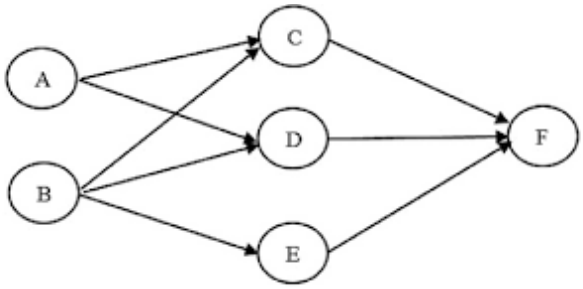
- Eigenfaces
- Fisher faces
- Local Binary Patterns Histograms

Process Pool Excecutor: It lets in parallelism of code and the Python language has approaches to perform this through multiprocessing module. From Python 3.2 onwards a brand new elegance referred to as ProcessPoolExecutor became delivered in python in concurrent. The futures module to effectively control and create Process.

Graph implementation to track a person:

Here we take the input of ip address of all cameras that are connected and create a graph data structure. We use a directed graph to map the cameras. The graph is created according to the location of the cameras and the connectivity of the place. The graph will depict the path of the detected person and will guide the search algorithm accordingly. At first the search algorithm will run in the last known location or in all the cameras of a particular place. On detecting the person, the search algorithm will be running only in that

particular surveillance. But if the algorithm is not able to find the person in that surveillance anymore. The algorithm will then look for the probable paths the person could have taken and it ensures to run the search algorithm in all these paths first. If the person is still not detected then the search algorithm is run in all the surveillance footages again. This method will make it easier to track down a person and optimise the search algorithm by searching in the most probable surveillance footage.



Results:

- Using this visual search technique on Realtime surveillance footages from multiple security closed-circuit television (CCTV) system the person is located and his current whereabouts are displayed.
- The person's current location and the person's mobility guides the search algorithm to perform its next visual search on its adjacent surveillance footages.
- The persons location can be live tracked and his activities can be monitored through surveillance.
- Any missing kids or individuals can be traced easily.
- Time taken by intelligent agencies to track down a person/ an object is reduced , hence the security system is enhanced .

Conclusions:

In the field of image processing, face recognition has always been a hot topic. Our project takes a video as an inclusion, calculates and compares the known faces for each face in the frame and as well as add face identification. Well-known content is boxed in the CNN paradigm. As a result people or crowds, this algorithm recognizes them and track the particular person using a guided search algorithm until a stop command is encountered. The whereabouts of the person is logged continuously into a file and is displayed constantly as he moves. We have introduced a real-time tracking and identification framework in multiple surveillance footages. Face models, based on Local Binary Pattern Histograms, of various people are taken offline and stored on the website. This model is built to detect queries with a wide range of face, shape and brightness. The approach seems promising and enhances performance in identifying and tracking of a person. The fact that facial recognition proves to be reliable is very important as in most cases having the actual identity (not just the label) of the person being tracked is important.

Scope for the future works:

In future we are planning to implement this search algorithm remotely on the ip camera. Using this enhancement we can completely reduce the processing power and implement on a larger scale(cities and localities). To implement this process we connect a cluster of cameras to a network and perform the search on the required cluster of cameras.

We also aim to enhance tracking of a person by locating his present co-ordinates in the footage and predict his future movements according to it. This can be implemented by taking down the co-ordinates and querying the specified co-ordinate in the neighboring camera graph to run the face detection model on the required cameras.

We aim to automate the graphing process of the cameras ,by looking through the past datasets .We can achieve this by using location co-ordinates of the persons movements and analyzing it with the data set of the cameras connected at the specified location.