A
Project Report on
CROWD DETECTION SYSTEM

By

MANISH SINGH PADYAL

DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING

Model Institute of Engineering & Technology

A
Project Report on

CROWD DETECTION SYSTEM

In partial fulfillment of requirements for the degree of

Master of Computer Applications

SUBMITTED BY:

MANISH SINGH PADYAL

Under the Guidance of

Prof. Ankur Gupta
Asst. Prof. Rishi Gupta
Asst. Prof. Purnendu Prabhat
Dr. Sahil Sawhney

DEPARTMENT
OF
COMPUTER SCIENCE & ENGINEERING
Model Institute of Engineering & Technology – Jammu

CERTIFICATE

Certified that major project work entitled “CROWD DETECTION SYSTEM” is a bonafide work carried out in the 2nd semester by “MANISH SINGH PADYAL” in partial fulfillment for the award of Master of Computer Applications from Model Institute of Engineering & Technology during the academic year 2023-2024.

Project Guide:
Prof. Ankur Gupta
Asst. Prof. Purnendu Prabhat
Asst. Prof. Rishi Gupta
Dr. Sahil Sawhney

ACKNOWLEDGEMENT

The accomplishment of this Major Project report on “Crowd Detection System” benefits of the guidance, moral support and encouragement bestowed on me throughout my work period. For this, I wish to express my sincere gratitude and appreciation to all those who gave me the possibility to complete this report.

I would like to acknowledge with much appreciation, the crucial role of the faculty of MCA department and CRIE of MIET, Jammu, who gave me the permission to use all the resources and material required to complete my report. I cannot express enough thanks to Prof. Ankur Gupta, Asst. Prof. Purnendu Prabhat, Asst. Prof.
Rishi Gupta and Dr. Sahil Sawhney for their continued support and encouragement to work on my own and discover new things in this field.

Lastly, I must offer my sincere appreciation for the receipt of affectionate care and opportunities to Model Institute of Engineering and Technology (MIET) for providing me with such a wonderful work environment.

My accomplishment of this report could not have been possible without the support of my classmates and my family, who provided me with love, support and encouragement when the times got rough.

MANISH SINGH PADYAL
MCA, MIET

INDEX PAGE

Topic Page **Page No.**

FIGURE INDEX

Topic Page **Page No.**

ABSTRACT

In the present environment, where technology and artificial intelligence is an emerging edge of evolution for the world that human live in. There is a wide research going on in the field of crowd behavior analysis, through which many problems that result in loss to human security and property are being dealt with.

With the expanding population and several problems arising due to crowded situations, the necessity of crowd detection is also at a raise. It includes assessing the number of individuals in the group and in addition the appropriation of the crowd density in different regions of the group. Estimation of such crowd density can be done from the image of the crowded scene. In CNN approach, deep learning technique is used to estimate the crowd density. Since, crowd behavior analysis is an emerging field in the scope of artificial intelligence, the problem as discussed can be relieved to a certain level by precognition of stampede at the region of interest, i.e., pilgrim places.

Pilgrimage in a country like India, which ranks 2nd after China according to the report statistics of World Population Prospects as in 1st July, 2019, is a session gathering event which attracts large crowd due to the importance given to the religious thoughts. The system so proposed is based on the concepts and approaches of computer vision, image analysis and deep learning. The crowd is analyzed and tracked by the neural model are compared in context to the density limits of the defined area, which if exceed are marked as threat and an alert signal is displayed on the video feed as received by the authorities.

In the current situation where COVID19 virus or CORONA VIRUS is declared as a pandemic disease due to which human life is facing dire consequences. The main focus of mostly all the countries and states is to create an atmosphere of social distancing which is a crucial step in order to restrict this disease from spreading at a fierce pace.

In such a situation where, social distancing is to be monitored by the officials personally, thereby causing a risk to their own self and their related family members, a solution such as Crowd Detection System can come in handy to help out the officials.

CHAPTER 1: INTRODUCTION

1.1. Computer Vision

Computer vision tasks include methods for acquiring, processing, analysing and understanding digital images, and extraction of high-dimensional data from the real world in order to produce numerical or symbolic information, e.g. in the forms of decisions. Understanding in this context means the transformation of visual images (the input of the retina) into descriptions of the world that can interface with other thought processes and elicit appropriate action. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory.

In surveillance applications, the video streams captured by an increasing number of cam-eras monitoring public spaces must be watched by a limited number of human observers, and computer vision algorithms could be used to detect anomalous events and warn the observers. Despite the potential of crowd analysis applications using computer vision algorithms, most of the existing work for detection/identification of people, groups of people, or estimating motion have been focused on noncrowded situations.

Although one may think that a straightforward extension of techniques designed for noncrowded scenes could be suitable for dealing with crowded situations, that is not true. First of all, a crowd is something beyond a simple sum of individuals. The crowd can assume different and complex behaviours as those expected by their individuals. The behaviour of crowds is widely understood to have collective characteristics that can be described in general terms.

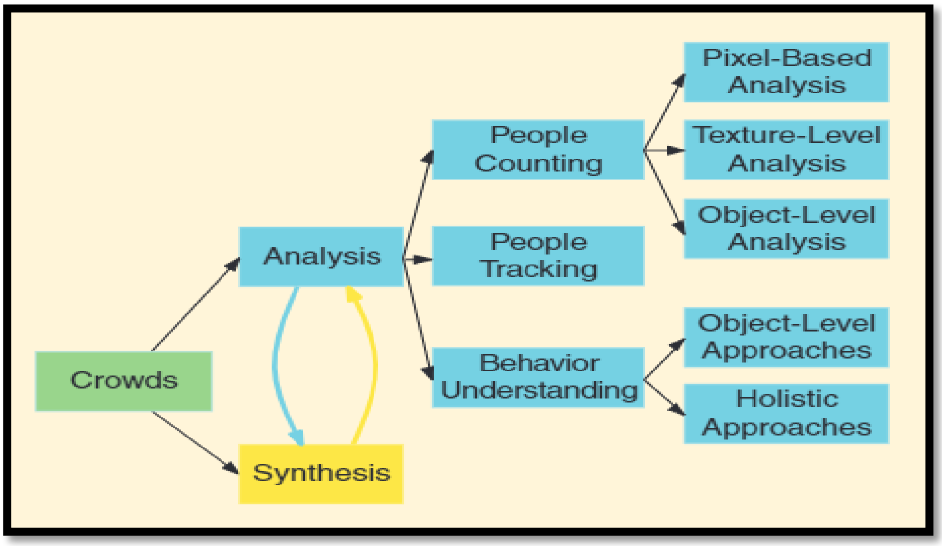


Figure 1:Crowd Dynamics Schema

1.2. Person (Object) Detection and Recognition

Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and pedestrian detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance. Computer vision is an interdisciplinary scientific field that deals with how computers can be made to gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to automate tasks that the human visual system can do.

Object recognition – technology in the field of computer vision for finding and identifying objects in an image or video sequence. Humans recognize a multitude of objects in images with little effort, despite the fact that the image of the objects may vary somewhat in different viewpoints, in many different sizes and scales or even when they are translated or rotated. Objects can even be recognized when they are partially obstructed from view. This task is still a challenge for computer vision systems. Many approaches to the task have been implemented over multiple decades.

Moving object detection is to recognize the physical movement of an object in a given place or region. By acting segmentation among moving objects and stationary area or region, the moving objects motion could be tracked and thus could be analysed later.

The detection process generally occurs in two steps: object detection and object classification. Object detection could be performed by background subtraction, optical flow and spatio-temporal filtering. Background subtraction is a popular method for object detection where it attempts to detect moving objects from the difference between the current frame and a background frame in a pixel-by-pixel or block-by-block fashion. There are few available approaches to perform background subtraction. The most common ones are adaptive Gaussian mixture, non-parametric background, temporal differencing, warping background and hierarchical background models. The optical flow-based object detection technique uses characteristics of flow vectors of moving objects over time to detect moving regions in an image sequence. Apart from their vulnerability to image noise, colour and non-uniform lighting, most of the flow computation methods have large computational requirements and are sensitive to motion discontinuities. For motion detection based on the spatial-temporal filter methods, the motion is characterized via the entire three-dimensional (3D) spatial-temporal data volume spanned by the moving person in the image sequence.

Their advantages include low computational complexity and a simple implementation process. However, they are susceptible to noise and variations of the timings of movements.

The object classification methods could be divided into three categories: shape-based, motion-based and texture-based. Shape-based approaches first describe the shape information of moving regions such as points, boxes and blobs. Then, it is commonly considered as a standard template-matching issue. However, the articulation of the human body and the differences in observed viewpoints lead to a large number of possible appearances of the body, making it difficult to accurately distinguish a moving human from other moving objects using the shape-based approach. This challenge could be overcome by applying part-based template matching. Texture-based methods such as histograms of oriented gradient (HOG) use high dimensional features based on edges and use support vector machine (SVM) to detect human regions.

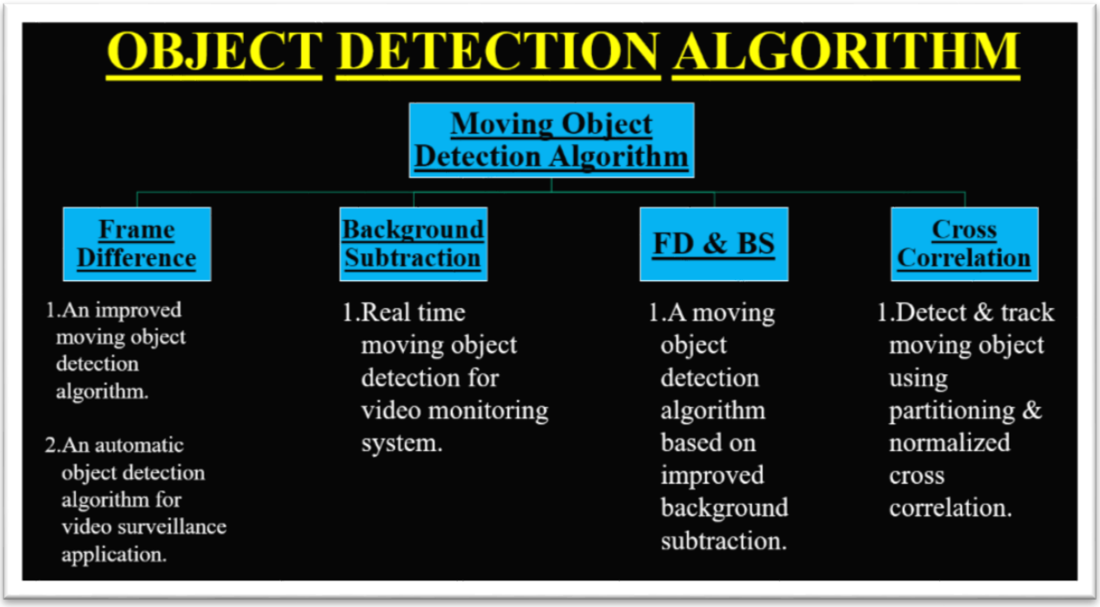

OBJECT DETECTION ALGORITHM
Moving Object Detection Algorithm
Frame Difference
Background Subtraction
FD & BS
Cross Correlation
1.An improved moving object detection algorithm.
2.An automatic object detection algorithm for video surveillance application.
1.Real time moving object detection for video monitoring system.
1.A moving object detection algorithm based on improved background subtraction.
1.Detect & track moving object using partitioning & normalized cross correlation.

Figure 2:Object Detection Algorithms Classified

1.3. Person (Object) Tracking

Object tracking is the process of locating a moving object (or multiple objects) over time using a camera. It has a variety of uses, some of which are: human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging and video editing. Video tracking can be a time-consuming process due to the amount of data that is contained in video. Adding further to the complexity is the possible need to use object recognition techniques for tracking, a challenging problem in its own right. The proliferation of high-powered computers, the availability of high quality and inexpensive video cameras, and the interesting need for automated video analysis has generated a great deal of interest in object tracking.

In its simplest form, tracking can be defined as a method of following an object through successive image frames to determine its relative movement with respect to other objects. In other words, a tracker assigns consistent labels to the tracked objects in different frames of video.

Tracking is an important problem in the domain of computer vision. It involves tracking an object through a sequence of frames. An ID is assigned to an object the first time it appears, and then this ID is carried forward in subsequent frames. The use of tracking is exploding with applications in retail stores, self-driving cars, security and surveillance, motion capture systems, and many more. Tracking is a challenging problem for several reasons — tracked items can get fully or partially occluded, tracked items may look similar to each other (causing ID switching), and an object may disappear completely only to reappear later. Though these challenges cannot be completely removed, they can be reduced by wisely choosing tracking and detection algorithms and combining them with human intelligence.

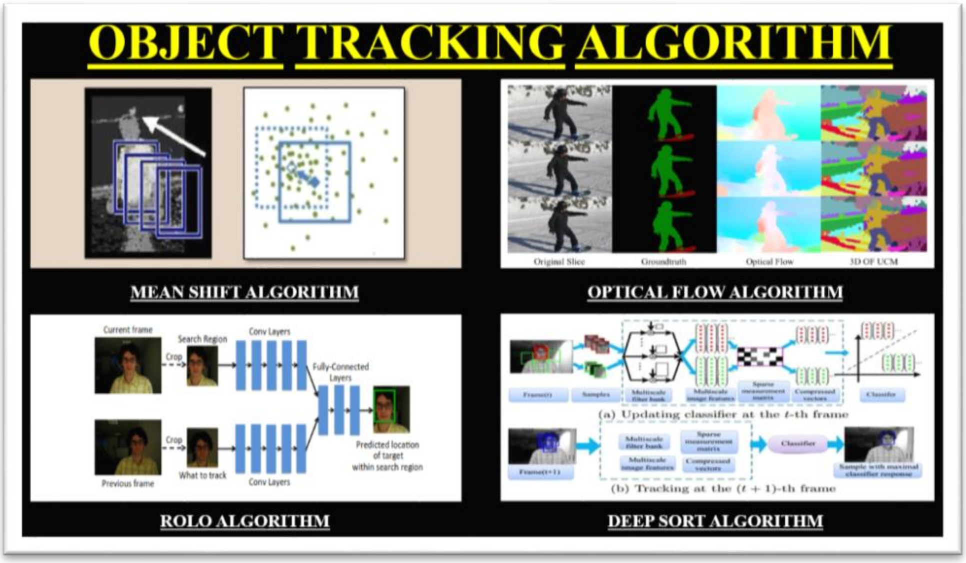


Figure 3:Object Tracking Algorithm Classified

1.4. Artificial Intelligence and Deep Learning

Artificial Intelligence is a theory that is used to make expert systems that can perform activities with the same intelligence as of humans. In the modern world of digitalization, the need to develop expert system is growing tremendously. Most of the expert systems perceive environment as image, and for reading the components of image there are various techniques which require a system to have vision. One of the important fields of Artificial Intelligence is Computer Vision. Computer Vision is the science of computers and software systems that can recognize and understand images and scenes. Computer Vision is also composed of various aspects such as image recognition, object detection, image generation, image super-resolution and more.

Deep Learning is an emerging field of Artificial Intelligence. Neural system of a human can be compared to neural models in deep learning. Neurons here, are defined as the functions to be run again and again in order to achieve precision called convolution. Once trained, the model can be used to implement the type of task, i.e., either classification or recognition based on the interest and domain of problem statement.

CHAPTER 2: TECHNICAL DETAILS

In this project, we have used DNN (deep neural network) to implement detection along with Mobile-net SSD (single shot detectors) to improve efficiency. Along with these model architectures, we have used libraries and modules like OpenCV, Dlib, Imutils, and NumPy that are explained below in detail with their working in project. All these libraries and neural models are easily available on the internet for installing and using them for project/product/research usage.

2.1. Deep Neural Networks (DNN)

A neural network, in general, is a technology built to simulate the activity of the human brain – specifically, pattern recognition and the passage of input through various layers of simulated neural connections. Many experts define deep neural networks as networks that have an input layer, an output layer and at least one hidden layer in between. Each layer performs specific types of sorting and ordering in a process that some refer to as “feature hierarchy.”

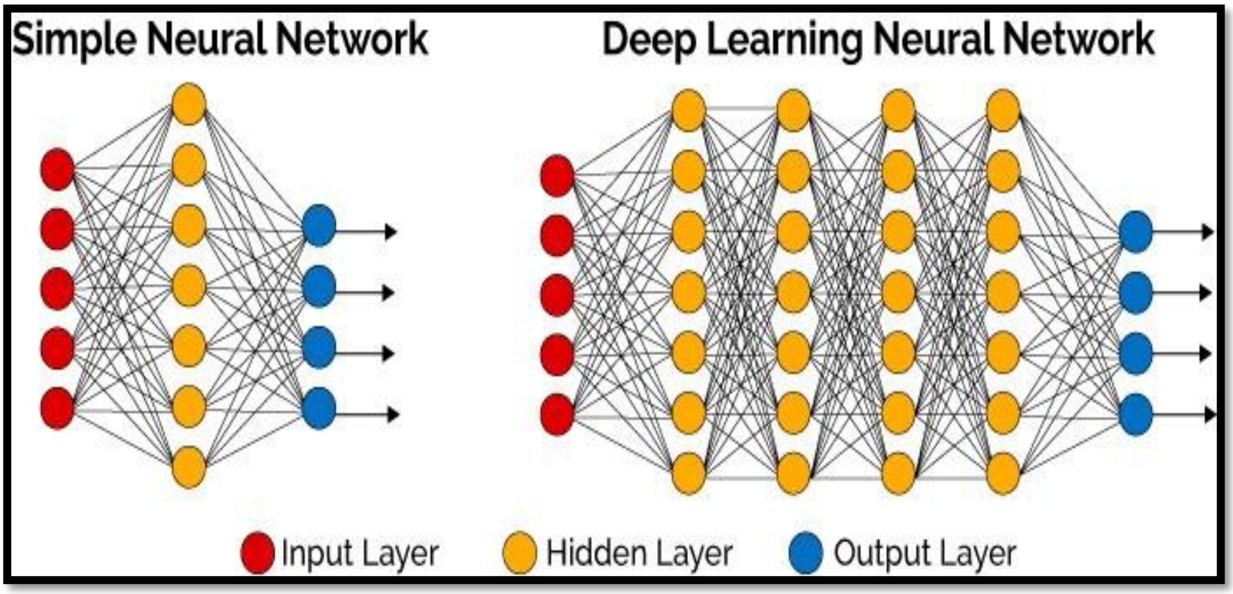


Figure 4:DEEP NURAL NETWORKS

One of the key uses of these sophisticated neural networks is dealing with unlabelled or unstructured data.

While simple neural networks have a shorter epoch (training convolution), a deep neural network has a wider and deep set of training convolutions in order to improve its efficiency, so that it can detect the desired object with higher precision than a simple neural network. Above given figure shows the working of a deep neural network in contrast to a simple neural network.

The number of architectures and algorithms that are used in deep learning is wide and varied. We here discuss the CNN as it is used in Crowd Detection System. A CNN is a multilayer neural network that was biologically inspired by the animal visual cortex. The architecture is particularly useful in image-processing applications. The first CNN was created by Yann LeCun; at the time, the architecture focused on handwritten character recognition, such as postal code interpretation. As a deep network, early layers recognize features (such as edges), and later layers recombine these features into higher-level attributes of the input.

CNN architecture is made up of several layers that implement feature extraction, and then classification (see the following image). The image is divided into receptive fields that feed into a convolutional layer, which then extracts features from the input image. The next step is pooling, which reduces the dimensionality of the extracted features (through down-sampling) while retaining the most important information (typically through max pooling). Another convolution and pooling step is then performed that feeds into a fully connected multilayer perceptron. The final output layer of this network is a set of nodes that identify features of the image (in this case, a node per identified number). You train the network by using back-propagation.

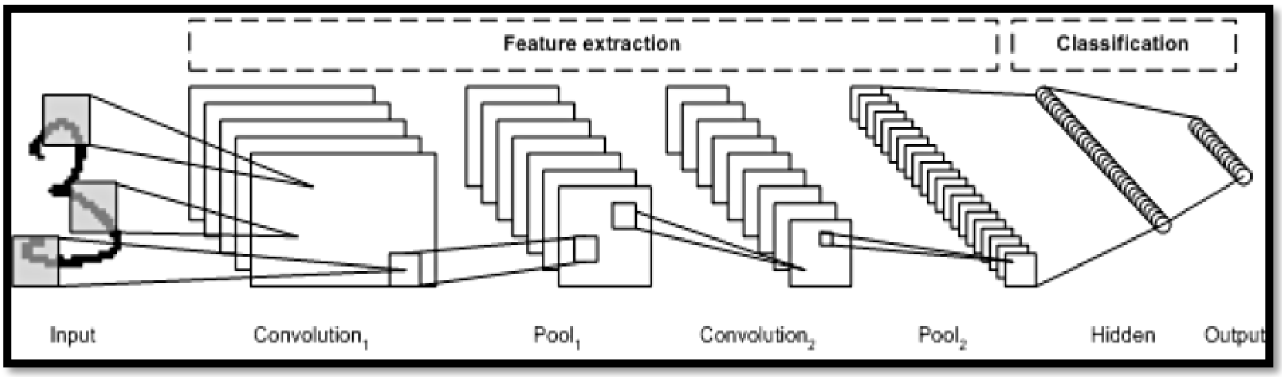


Figure 5:Deep CNN Architecture

The use of deep layers of processing, convolutions, pooling, and a fully connected classification layer opened the door to various new applications of deep learning neural networks. In addition to image processing, the CNN has been successfully applied to video recognition and various tasks within natural language processing.

Recent applications of CNNs and LSTMs produced image and video captioning systems in which an image or video is summarized in natural language. The CNN implements the image or video processing, and the LSTM is trained to convert the CNN output into natural language.

2.2. MOBILENET SSD

SSD (Single Shot Multi-Box Detector) is a popular algorithm in object detection. It’s generally faster than Faster RCNN. SSD is designed for object detection in real-time. Faster R-CNN uses a region proposal network to create boundary boxes and utilizes those boxes to classify objects. While it is considered the start-of-the-art in accuracy, the whole process runs at 7 frames per second. Far below what a real-time processing need. SSD speeds up the process by eliminating the need of the region proposal network.

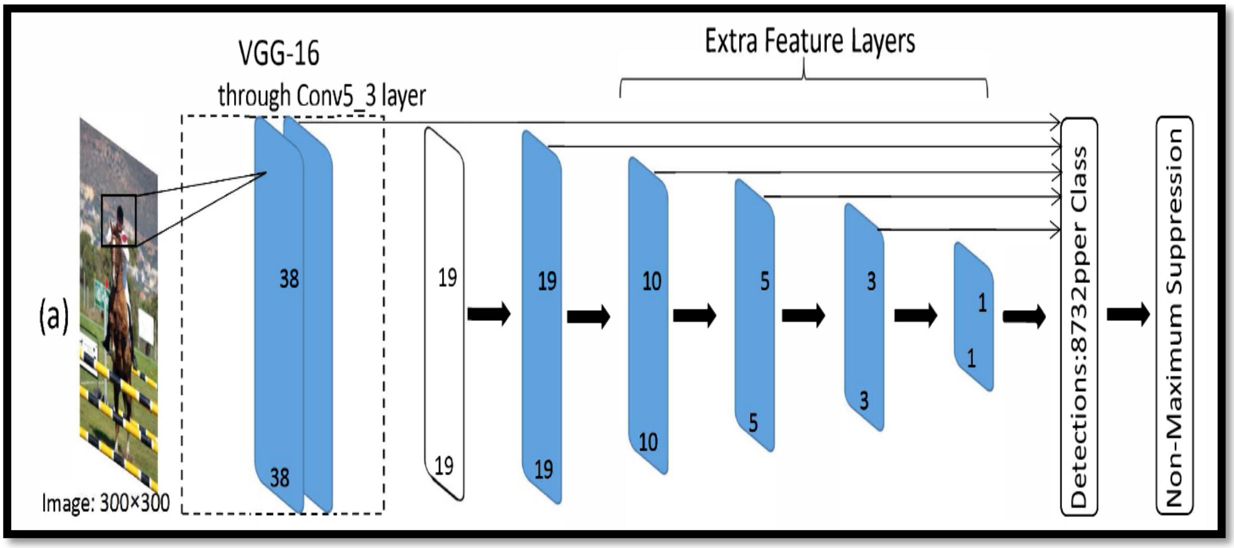


Figure 6:Mobilenet SSD Architecture

To recover the drop-in accuracy, SSD applies a few improvements including multi-scale features and default boxes. These improvements allow SSD to match the Faster R-CNN’s accuracy using lower resolution images, which further pushes the speed higher. According to the following comparison, it achieves the real-time processing speed and even beats the accuracy of the Faster R-CNN.

The paper about SSD: Single Shot MultiBox Detector (by C. Szegedy et al.) was released at the end of November 2016 and reached new records in terms of performance and precision for object detection tasks, scoring over 74% mAP (mean Average Precision) at 59 frames per second on standard datasets such as PascalVOC and COCO.

To better understand SSD, let’s start by explaining where the name of this architecture comes from:

- Single Shot: this means that the tasks of object localization and classification are done in a single forward pass of the network
- MultiBox: this is the name of a technique for bounding box regression developed by Szegedy et al. (we will briefly cover it shortly)
- Detector: The network is an object detector that also classifies those detected objects

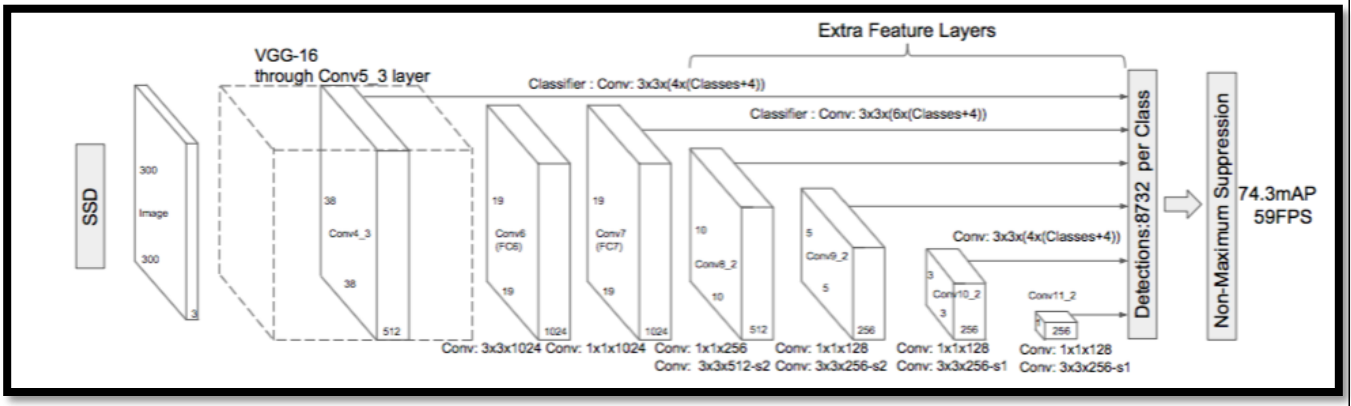


Figure 7:Detailed SSD Working

SSD’s architecture builds on the venerable VGG-16 architecture, but discards the fully connected layers. The reason VGG-16 was used as the *base network* is because of its strong performance in high quality image classification tasks and its popularity for problems where *transfer learning* helps in improving results. Instead of the original VGG fully connected layers, a set of *auxiliary* convolutional layers (from *conv6* onwards) were added, thus enabling to extract features at multiple scales and progressively decrease the size of the input to each subsequent layer.

2.3. OPEN CV

OpenCV (Open source computer vision) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itseez (which was later acquired by Intel[2]). The library is cross-platform and free for use under the open-source BSD license. It has many applications ranging from 2D &3D Feature Toolkits and Facial Recognition to Motion Tracking, HCI (Human Computer Interactions), Mobile Robotics, Emotion Recognition and so on. OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. There are bindings for Python, Java, and MATLAB/Octave. The API for these interfaces can be found in the online documentation.

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001. 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.

Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier. Then we need to extract features from it. For this, Haar features shown in the below image are used. Each feature is a single value obtained by subtracting sum of pixels under the white rectangle from sum of pixels under the black rectangle.

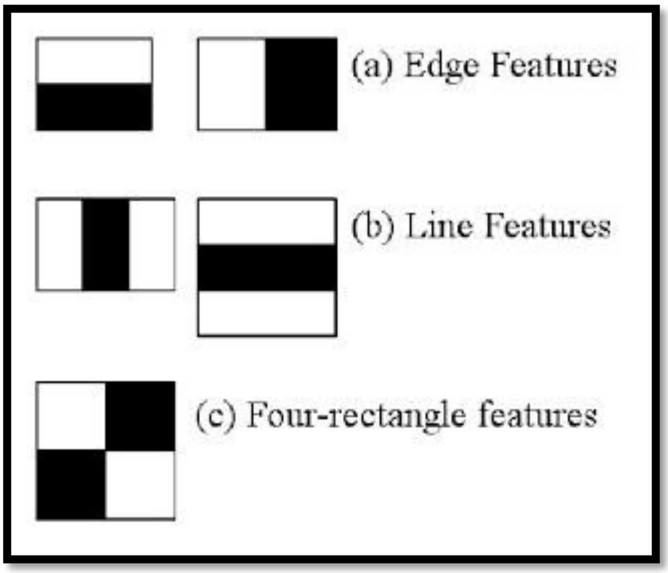


Figure 8:Feature Maps of Haar Classifiers in OpenCV

2.4. NUMPY

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, which in this project is used to keep track of multiple lists created as a result of tracking of people in the video frame. It also provides tools and functions to work with these arrays and is fundamental package for scientific computing in python. Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data (features of human body parts in this project case). Arbitrary data-types can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

Numpy array is a powerful N-dimensional array object which is in the form of rows and columns. We can initialize numpy arrays from nested Python lists and access it elements. To install Python NumPy, go to your command prompt and type “pip install numpy”. Once the installation is completed, go to your IDE (For example: PyCharm) and simply import it by typing: “import numpy as np”

We use python numpy array instead of a list because of the below three reasons:

1. Less Memory
2. Fast
3. Convenient

The very first reason to choose python numpy array is that it occupies less memory as compared to list. Then, it is pretty fast in terms of execution and at the same time it is very convenient to work with numpy. So these are the major advantages that python numpy array has over list.

2.5. Dlib and Imutils

Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems. It is used in both industry and academia in a wide range of domains including robotics, embedded devices, mobile phones, and large high-performance computing environments. Dlib's open source licensing allows you to use it in any application, free of charge.

Imutils however is a series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, and displaying Matplotlib images easier with OpenCV and Python.

CHAPTER 3: IMPLEMENTATION

3.1. IDEATION

Idea-development or ideation of any project is considered to be the rawest yet exciting phase of the development cycle. In this phase an idea is pondered over through sessions as brainstorming, worst possible idea, brainwriting, prototyping, sketching and other helpful ideation techniques. The aim here is to generate a bucket full of ideas and refine them into a singular best possible idea that has a definite scope and prospective to work it into a real-time project.

Initially, the ideation phase in my project's development carried a vast approach to find out the idea fit for development. In this phase, my ideas were rejected and reformed a number of times, after which final idea for the project, i.e., "Crowd Detection System" was framed.

Figure 9: Centre of Research Innovation and Entrepreneurship

The idea of crowd detection system started off as a mere density tracker, however running a lot of review runs on the selected idea with a review committee including my mentors Prof. Ankur Gupta, Asst. Prof. Purnendu Prabhat, Asst. Prof. Rishi Gupta and Dr. Sahil Sawhney, the ideation phase refined the idea to a vast crowd behaviour analysis aspect. And this had me the opportunity to work on a real-life problem.

3.2. DEFINITION

When we're about to venture into stormy territory, we know we need to prepare for a bumpy ride and take extra provisions so that we come out on the other side and arrive at our destination unscathed. Defining an idea into broader terms is also a similar brainstorming process. When this project idea was defined, the scope of the project was constrained to pilgrim places. However, once the challenges of feasibility of the concept to real world scenario came up, the scope was refined and cut-short through the previous prospects to a short-ranged area covered by the surveillance cameras at the pilgrimage.

3.3. RESEARCH AND REQUIREMENTS

Once the idea is refined with a clear path of definition, the next step is to understand the previously designed projects and study carefully the research prevailing in the same domain as that of your proposed idea. While surfing through internet, one easily comes across different approaches and proposals regarding the idea that one is working upon. Some of the approaches seem different while some have same thought process involved as the project developer.

In my case, the project was a stub of various approaches leading to a similar outcome. Many researches and developments that were related to the same idea were already at the pinnacle of success, yet struggled to find the best solution possible that may be efficient to create an output with lower error rate.

I went through different research papers, projects and books in order to choose the best fit algorithm, model architecture and methods for the project development. When you are going through various options available to choose the right fit for your project, you may select a wrong choice, but then you should have the courage to take risks and work upon it in a dignified manner and make your choice right.

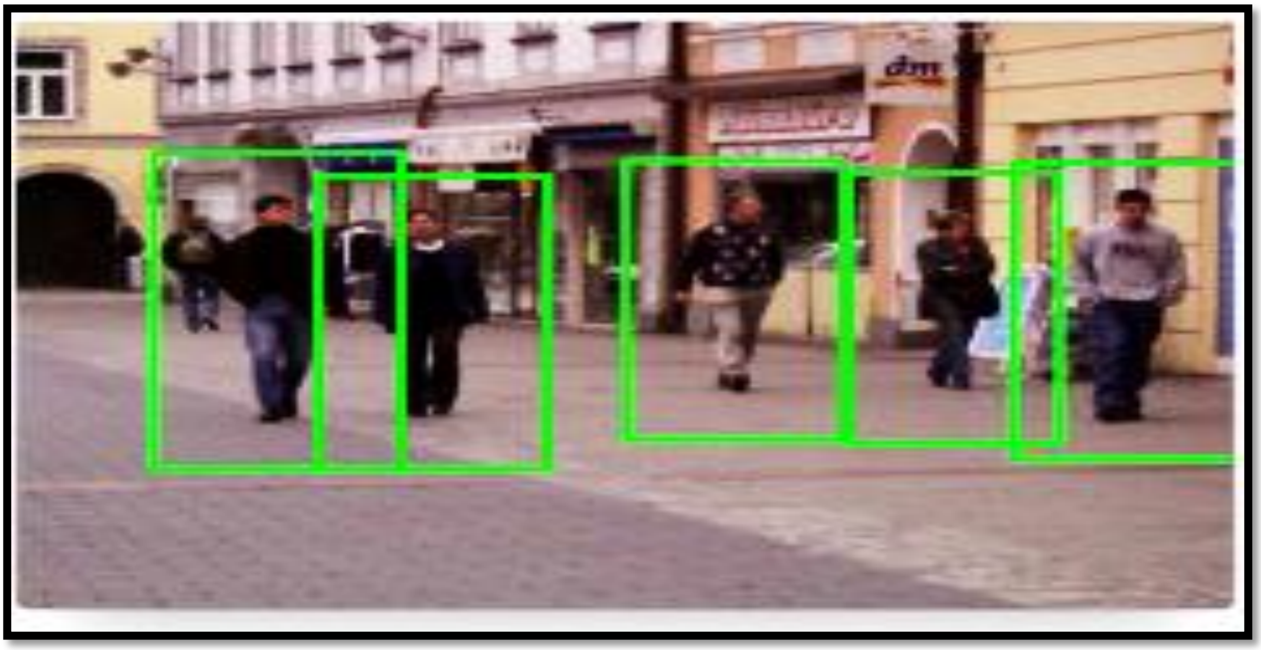


Figure 9: Neural Network Mobile Net SSD Chosen for Detection and Tracking

3.4. PROTOTYPE DEVELOPMENT

The prototype building phase is the most crucial phase in the project development cycle because it is the phase where various phases of the project development conclude in a single outcome that may or may not be desired.

In the prototype development for my project, I collected various pictures and videos from internet first. When the system worked on demonstratable pictures and videos, I considered to test the system for real world scenario. For that, we first constructed all possible test cases and then tested our system on them one-by-one.

When you are implementing the system prototype in real-world, a lot of difficulties may occur to which you have to provide the solution to. Some pictures taken from my test phase are given below as figure 5.

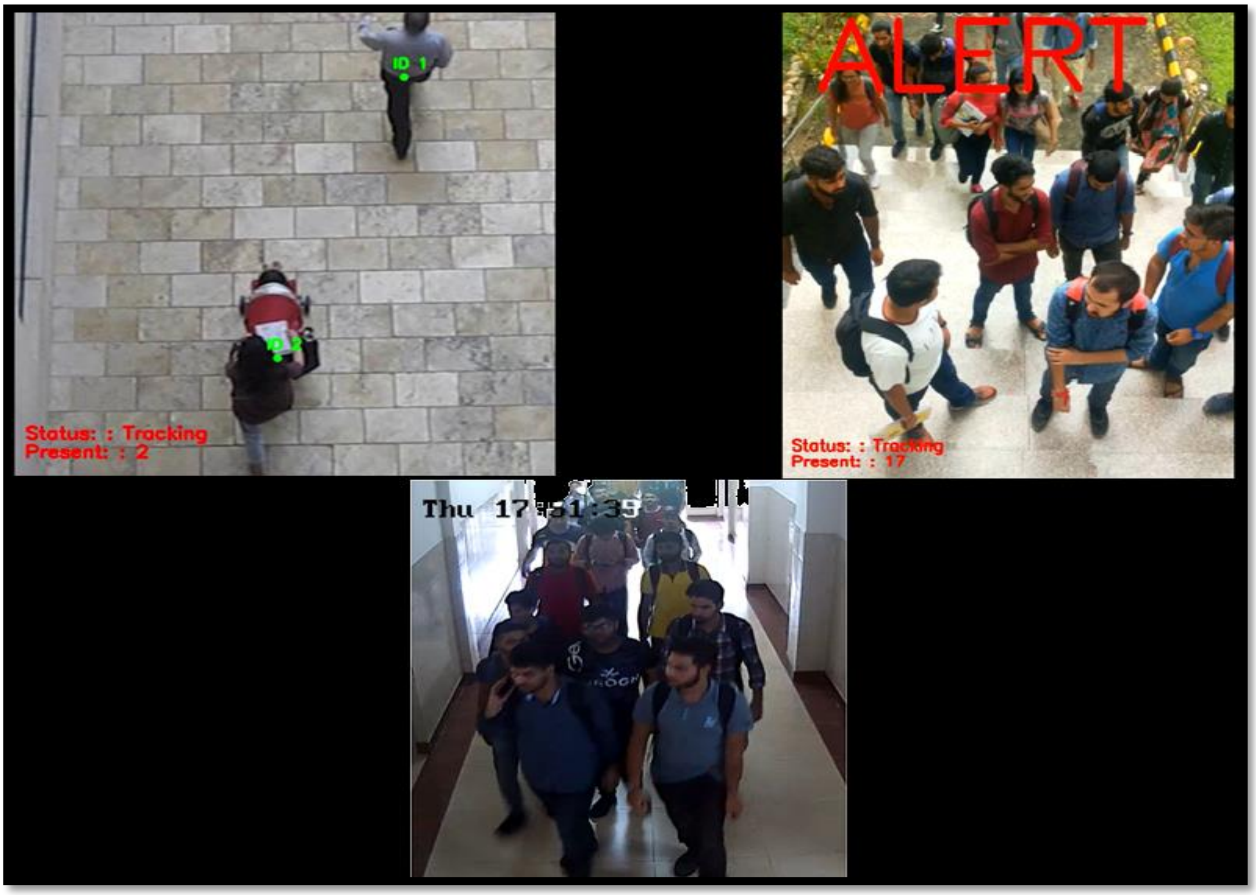


Figure 10: Test Cases

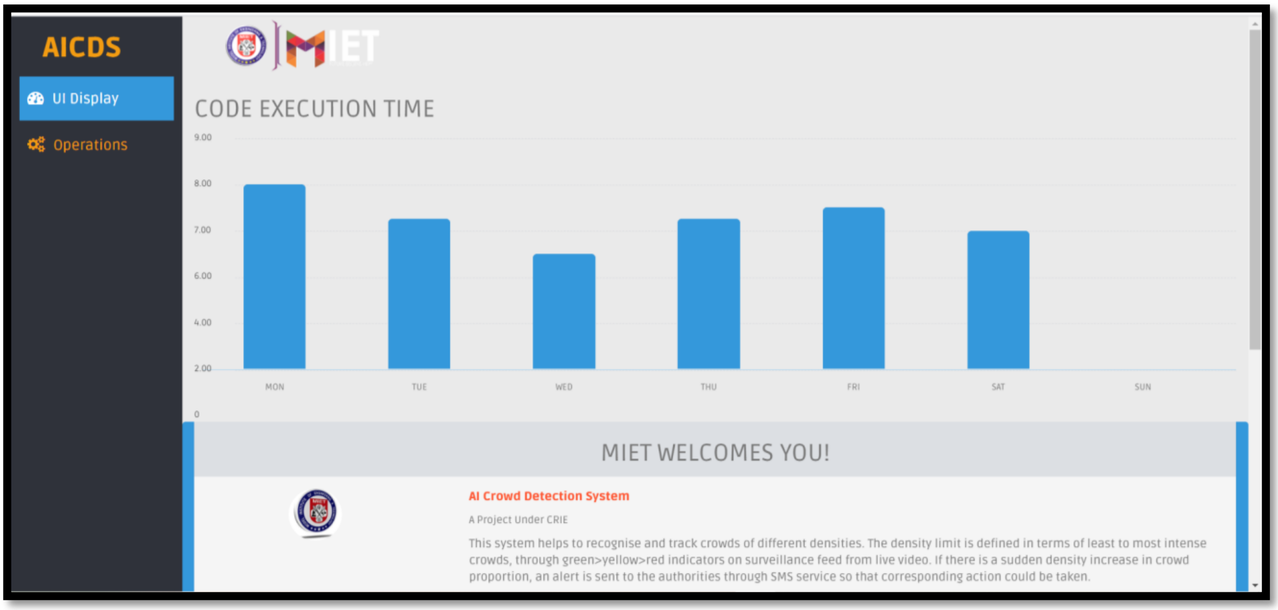
3.5. DASHBOARD OR FRONTEND ADMIN PANEL

Once the system is working properly on multiple test cases, the next step is to create a dashboard or an admin panel for helping the consumers or buyers to easily interact with the system without having the need to know the underlying details about how the system adapts the code or program that is written in the scripted text running in the terminal stack or kernel.

A data dashboard is an information management tool that visually tracks, analyzes and displays key performance indicators (KPI), metrics and key data points to monitor the health of a business, department or specific process. They are customizable to meet the specific needs of a department and company. Behind the scenes, a dashboard connects to your files, attachments, services and API's, but on the surface displays all this data in the form of tables, line charts, bar charts and gauges. A data dashboard is the most efficient way to track multiple data sources because it provides a central location for businesses to monitor and analyze performance. Real-time monitoring reduces the hours of analyzing and long line of communication that previously challenged businesses.

Firstly, users need to know that dashboard definition is dependent on the role it plays within an organization. Everyone uses data dashboards differently. Not all business dashboards serve the same purpose, which is why it is important users understand what KPIs to track and why. The business questions a dashboard answers depends on industry, department, process and position. Analytical dashboards are typically designed to help decision makers, executives and senior leaders, establish targets, set goals and understand what and why something happened with the same information they can use to implement appropriate changes. An analytical dashboard does this based on insights from data collected over a period of time determined by the user (i.e. last month, quarter or year).

Data is visualized on a dashboard as tables, line charts, bar charts and gauges so that users can track the health of their business against benchmarks and goals. Data dashboards surface the necessary data to understand, monitor and improve your business through visual representations. Depending on how you decide to design your dashboard, even straightforward numerical data can be visually informative by utilizing intuitive symbols, such as a red triangle facing downward to indicate a drop in revenue or a green triangle facing up to indicate an increase in website traffic.



CODE EXECUTION TIME

MIET WELCOMES YOU!

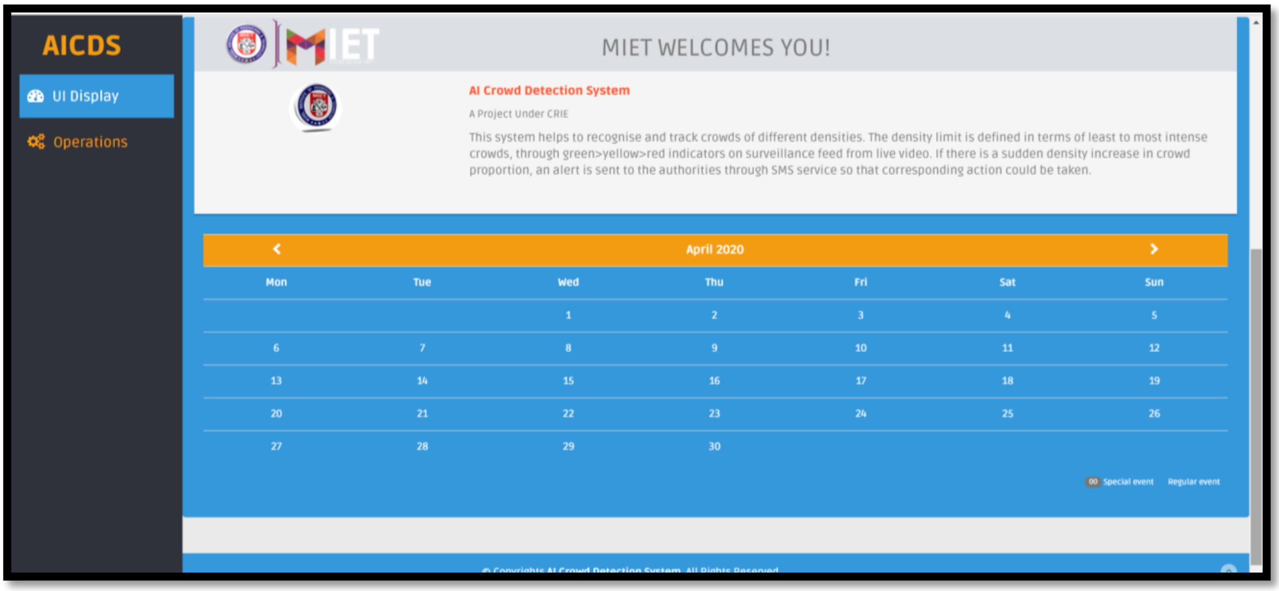
AI Crowd Detection System

A Project Under CRIE

This system helps to recognise and track crowds of different densities. The density limit is defined in terms of least to most intense crowds, through green>yellow>red indicators on surveillance feed from live video. If there is a sudden density increase in crowd proportion, an alert is sent to the authorities through SMS service so that corresponding action could be taken.

Figure 11:Dashboard of Crowd Detection System

In case of Crowd Detection System, as you can see in this above dashboard view a lot of data analytics are embedded in order to get the desired results of crowd behaviour analytics proof of concept. The first open-up page on the website includes of multiple function elements as Bar graph for execution hour readings in order to get track of the time for which the product was used by the consumer end. A message display of information about the product to tell the viewers or consumers about the product on a brief account. It also includes a calendar for event monitoring perspective for the viewer or user of the product as you can see in the figure given below.



MIET WELCOMES YOU!

AI Crowd Detection System

A Project Under CRIE

This system helps to recognise and track crowds of different densities. The density limit is defined in terms of least to most intense crowds, through green>yellow>red indicators on surveillance feed from live video. If there is a sudden density increase in crowd proportion, an alert is sent to the authorities through SMS service so that corresponding action could be taken.

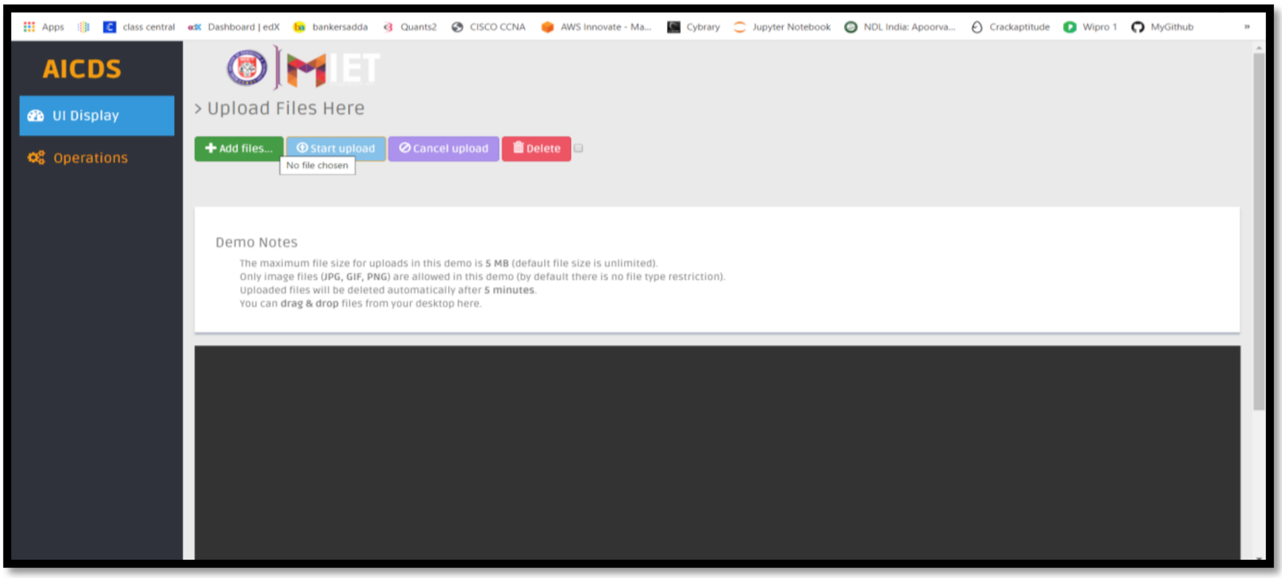
‹ April 2020 ›

Mon	Tue	Wed	Thu	Fri	Sat	Sun
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

00 Special event Regular event

© Copyrights AI Crowd Detection System. All Rights Reserved.

Figure 12: Dashboard View



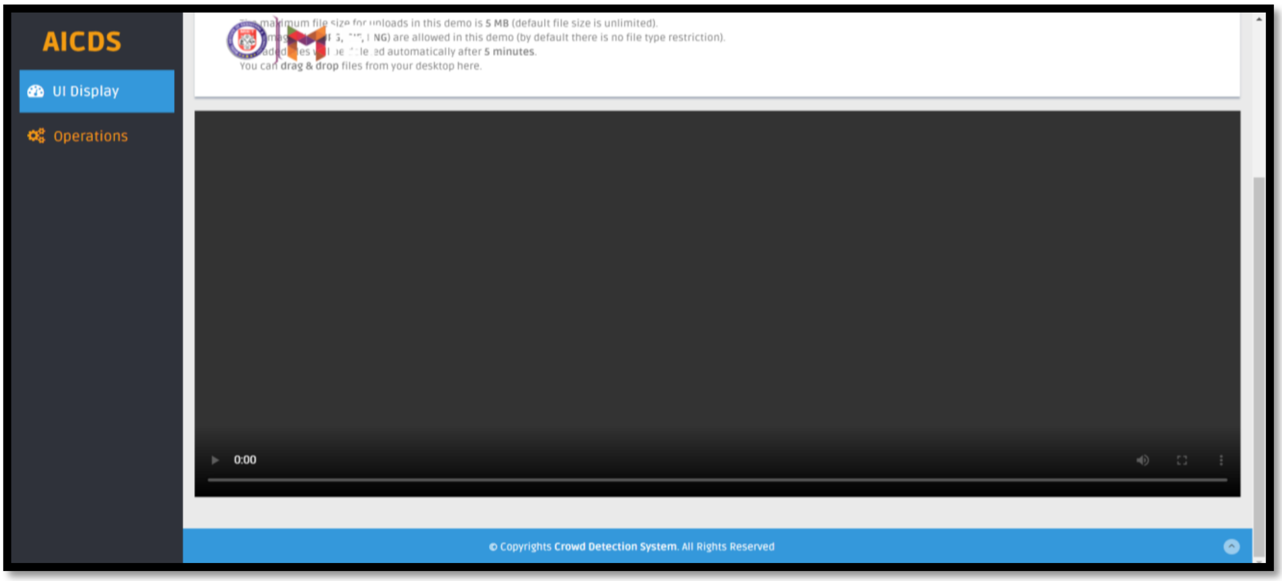
Demo Notes

The maximum file size for uploads in this demo is **5 MB** (default file size is unlimited).
Only image files (**JPG, GIF, PNG**) are allowed in this demo (by default there is no file type restriction).
Uploaded files will be deleted automatically after **5 minutes**.
You can **drag & drop** files from your desktop here.

Figure 13: Dashboard Surveillance Window

An operations tab in the side panel is provided in the dashboard for surveillance of the feed from camera or on an existing video feed already captured and stored in the output store of the terminal coding end folder, i.e., admin database for the user view end. The operations tag includes buttons to upload, process, delete or stop uploading a file, may it be a video or a photograph to get hold of crowd density behaviour or crowd motion analytics.

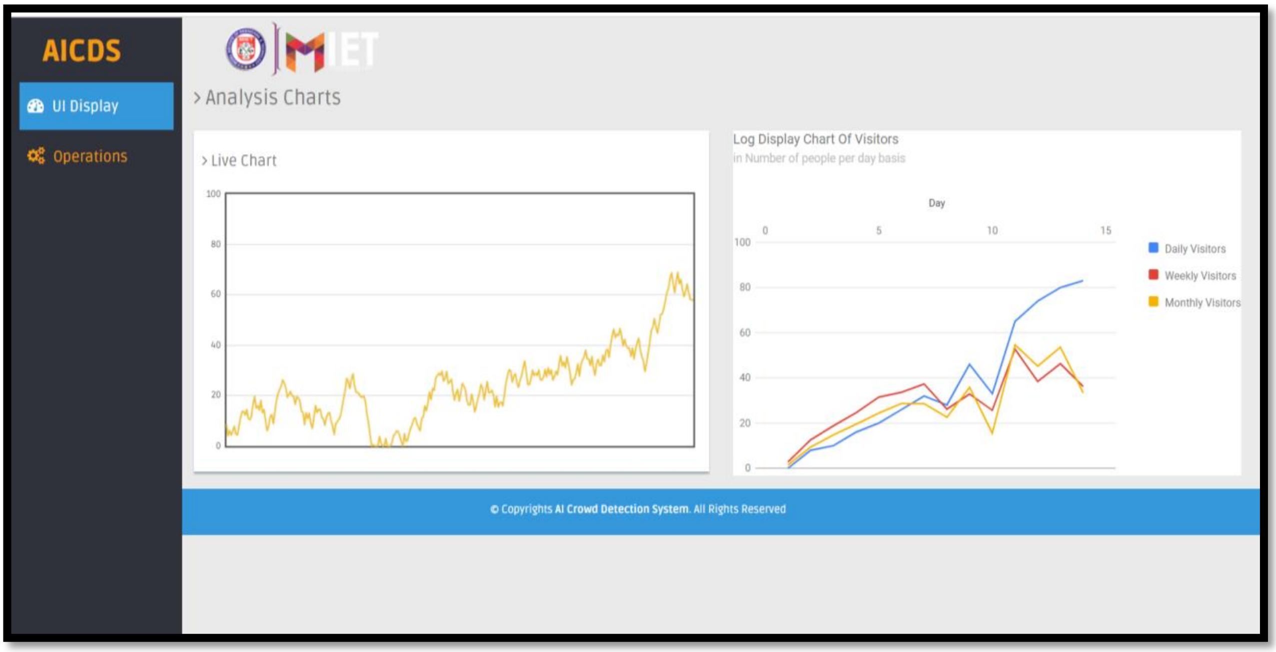
It also includes a live surveillance window included within it, through which a surveillant user can see what is happening around in the region he/she needs or wishes to survey upon. You can see the figure below for the surveillance window view of dashboard.



maximum file size for uploads in this demo is **5 MB** (default file size is unlimited).
NG) are allowed in this demo (by default there is no file type restriction).
ed automatically after **5 minutes**.
You can **drag & drop** files from your desktop here.

© Copyrights **Crowd Detection System**. All Rights Reserved

Figure 14: Live Surveillance Window



© Copyrights AI Crowd Detection System. All Rights Reserved

Figure 15: Analytics from Dynamic Data of Incurring Crowd

The Operations tag also consists of another option called as analytics. Under analytics tag, the user can view the peak hour analytics as well as the live feed analytics that are being processed by the system backend, or the server side.

In the live analytics, the data that gets stored on the database of Crowd Detection System gets transferred by making use of the GET and POST function requests of php servers on the internet server domain side. Thereby the client side can easily view the live feed dynamics through the live chart in the analytics section.

Also, the analytics section as shown in the diagram above includes the log of visitors displayed right by the side of live chart analytics. It includes the people count on daily, weekly and even monthly basis which can help improve the users of the system to efficiently manage crowds.

CONCLUSION

Testing phase is always considered as a scope for efficiency and further developments in the project. The testing phase is a risky phase where the developers or programmers have to test their project prototypes in an environment just like the one in which they desire to plant or establish their project.

Stimulating such an environment may pose many difficulties and obstacles because a real-world scenario is always a new scenario which can never be presumed with much accuracy.

In my scenario, since I had to analyse the group behaviour in pilgrimages, a crowd was required to create such an environment as pilgrimage that we targeted. Creating such an environment is a challenge as it requires the crowd of a given density within a defined area of surveillance. Yet I with the help of my institute, MIET, Jammu, was able to stimulate such an environment up to a bar so that it could help me in testing the prototype so developed.

Once your testing phase is successfully run once, then you are required to go back and test the prototype over different test cases which can help in modifying the project and help it to be useful for solving more problems of the same category.

Thereafter comes the business perspective of the project, which in my case is showing the customers how useful this project will be if a shopkeeper needs to keep a track of how many people come out on which day or which hour just in case to have a hold on the stocks and the supplies he has to keep in that situation where there is a large consumer base on a typical time of day or on a particular day of a week.

Also, this project comes in handy where there is a need of social distancing, like in the present situation in year 2020 where due to COVID19 outbreak social distancing is the only crucial step to prevent the disease from spreading at a large scale and thus preventing large audience to get affected by this pandemic.

Another most important aspect of this project is that, this project can be used where there is a need to keep a low-density crowd to reduce the chances of stampede at places like shrines, markets, protests, etc. These places are at a constant surveillance that includes personal human intervention. But with this project installed on the scene, personal human intervention can be reduced to a large extent and thus, it can help the authorities to manage the crowd.

So, with this I would like to conclude that making a project is worth, but what is really worth is the phases through which we go step-wise implementing the project. These phases help us to improve ourselves in every aspect, may it be technical, social or theoretical.

BIBLIOGRAPHY

[1] X H Fang1, W Xiong1, B J Hu1 and L T Wang2

1 The Institute of Command and Technology of Equipment, Beijing,

101416, China

2 Unit 92941, PLA, Huludao, Liaoning, 125000, China

[2] A novel approach to detect and track moving object using partitioning and NCC

Manoj S. Nagmode, Mrs. Madhuri A. Joshi, Ashok M. Sapkal College Of

Engineering, PUNE UNIVERSIT, INDIA

[3] INTERNATIONAL JOURNAL OF SYSTEMS APPLICATIONS, ENGINEERING &

DEVELOPMENT Issue 1, Volume 2, 2008 on Automatic detection, extraction &

Recognition of moving objects.

[4] Crowd Analysis Using Computer Vision Techniques.

Researchgate.net/publication/224172297

[5] Research on neural networks

Asp-eurasipjournals.springeropen.com/1687-6180-2013-176

[6] Link References:

link.springer.com

blog.pro-vigil.com

in.pycon.org

www.nanonotes.com

www.coursera.org

www.ijarcce.com

www. wikipedia.org

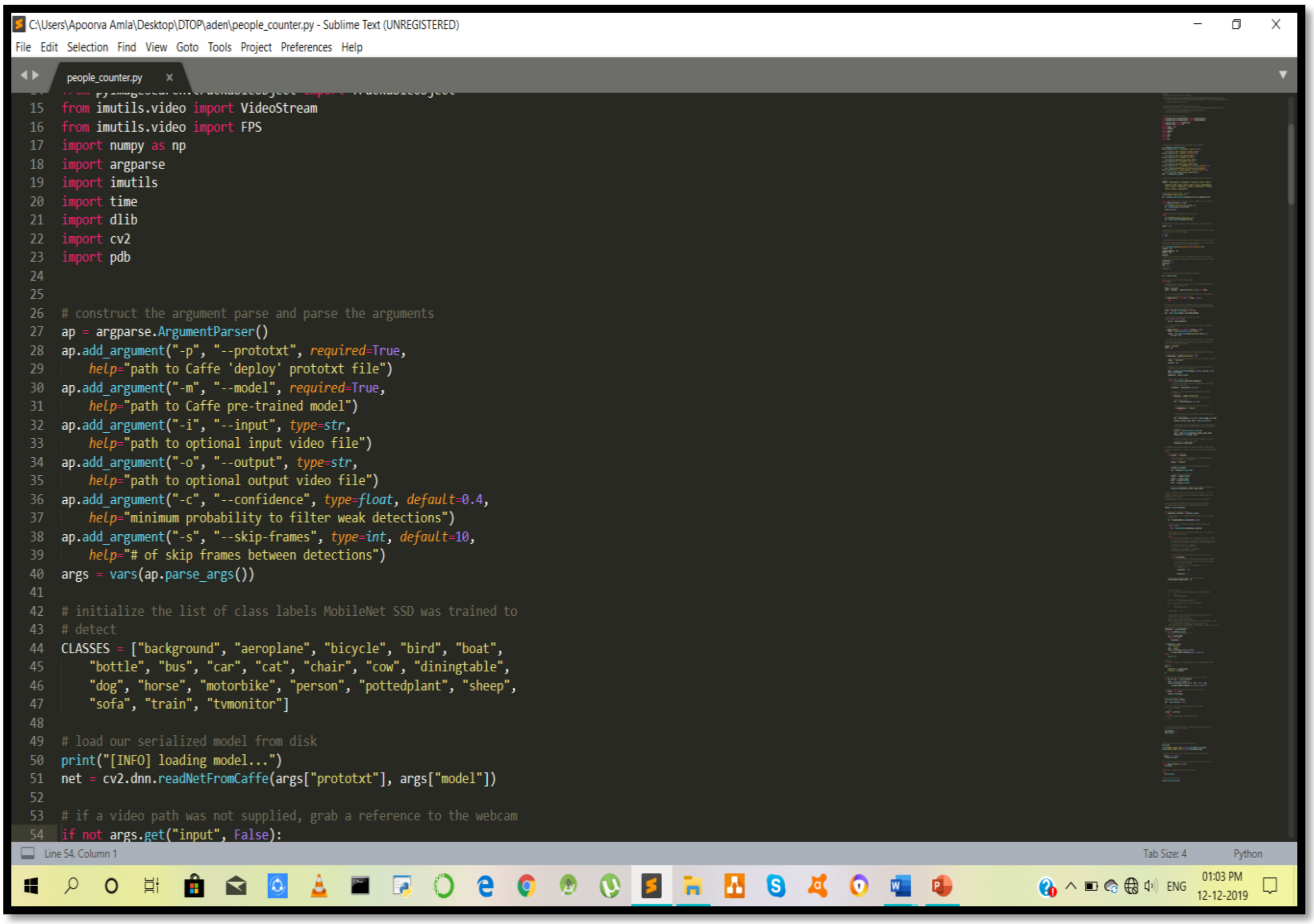
www.learnopencv.com

www.iopscience.iop.org

www.researchgate.net

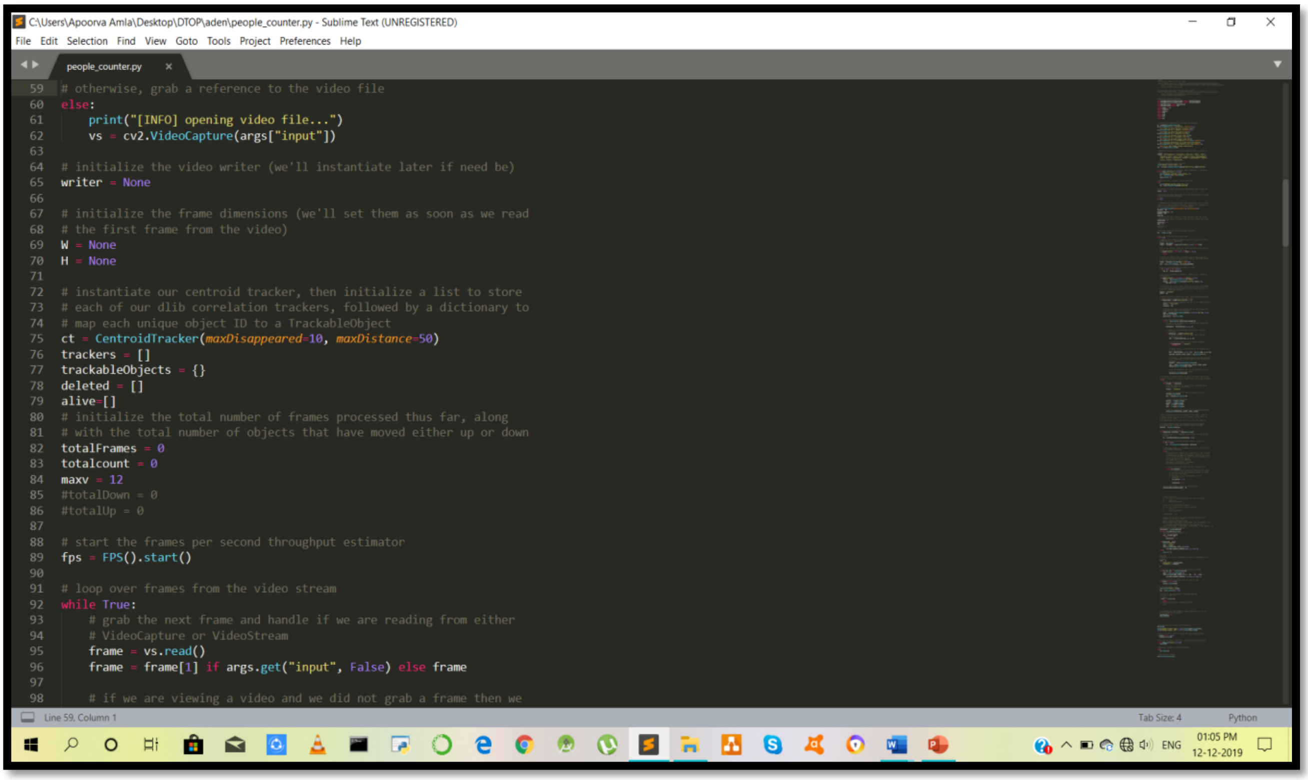
APPENDICES

CODE SNIPPET


C:\Users\Apoorva Amla\Desktop\DTOP\aden\people_counter.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
people_counter.py
Line 54, Column 1
Tab Size: 4
Python
ENG
01:03 PM
12-12-2019

```python
from pyimagesearch.trackableobject import TrackableObject
from imutils.video import VideoStream
from imutils.video import FPS
import numpy as np
import argparse
import imutils
import time
import dlib
import cv2
import pdb


# construct the argument parse and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-p", "--prototxt", required=True,
    help="path to Caffe 'deploy' prototxt file")
ap.add_argument("-m", "--model", required=True,
    help="path to Caffe pre-trained model")
ap.add_argument("-i", "--input", type=str,
    help="path to optional input video file")
ap.add_argument("-o", "--output", type=str,
    help="path to optional output video file")
ap.add_argument("-c", "--confidence", type=float, default=0.4,
    help="minimum probability to filter weak detections")
ap.add_argument("-s", "--skip-frames", type=int, default=10,
    help="# of skip frames between detections")
args = vars(ap.parse_args())

# initialize the list of class labels MobileNet SSD was trained to
# detect
CLASSES = ["background", "aeroplane", "bicycle", "bird", "boat",
    "bottle", "bus", "car", "cat", "chair", "cow", "diningtable",
    "dog", "horse", "motorbike", "person", "pottedplant", "sheep",
    "sofa", "train", "tvmonitor"]

# load our serialized model from disk
print("[INFO] loading model...")
net = cv2.dnn.readNetFromCaffe(args["prototxt"], args["model"])

# if a video path was not supplied, grab a reference to the webcam
if not args.get("input", False):
```

Figure 16: Code Importing required libraries and packages



C:\Users\Apoorva Amla\Desktop\DTOP\aden\people_counter.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
people_counter.py
Line 59, Column 1
Tab Size: 4
Python
ENG
01:05 PM
12-12-2019

```python
# otherwise, grab a reference to the video file
else:
    print("[INFO] opening video file...")
    vs = cv2.VideoCapture(args["input"])

# initialize the video writer (we'll instantiate later if need be)
writer = None

# initialize the frame dimensions (we'll set them as soon as we read
# the first frame from the video)
W = None
H = None

# instantiate our centroid tracker, then initialize a list to store
# each of our dlib correlation trackers, followed by a dictionary to
# map each unique object ID to a TrackableObject
ct = CentroidTracker(maxDisappeared=10, maxDistance=50)
trackers = []
trackableObjects = {}
deleted = []
alive=[]
# initialize the total number of frames processed thus far, along
# with the total number of objects that have moved either up or down
totalFrames = 0
totalcount = 0
maxv = 12
#totalDown = 0
#totalUp = 0

# start the frames per second throughput estimator
fps = FPS().start()

# loop over frames from the video stream
while True:
    # grab the next frame and handle if we are reading from either
    # VideoCapture or VideoStream
    frame = vs.read()
    frame = frame[1] if args.get("input", False) else frame

    # if we are viewing a video and we did not grab a frame then we
```

Figure 17: Initializing Variables

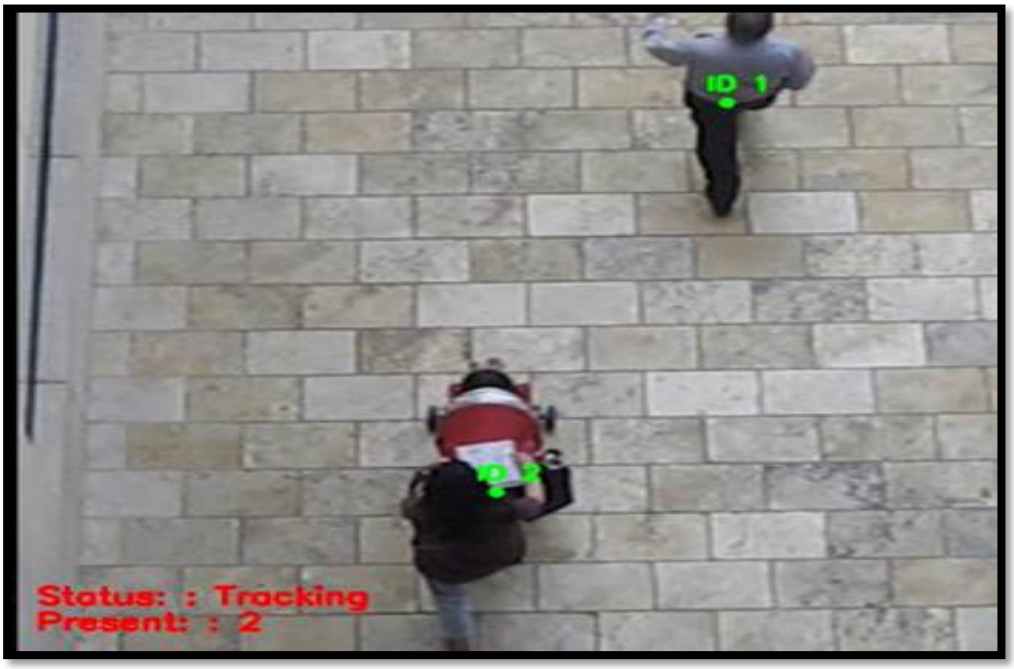
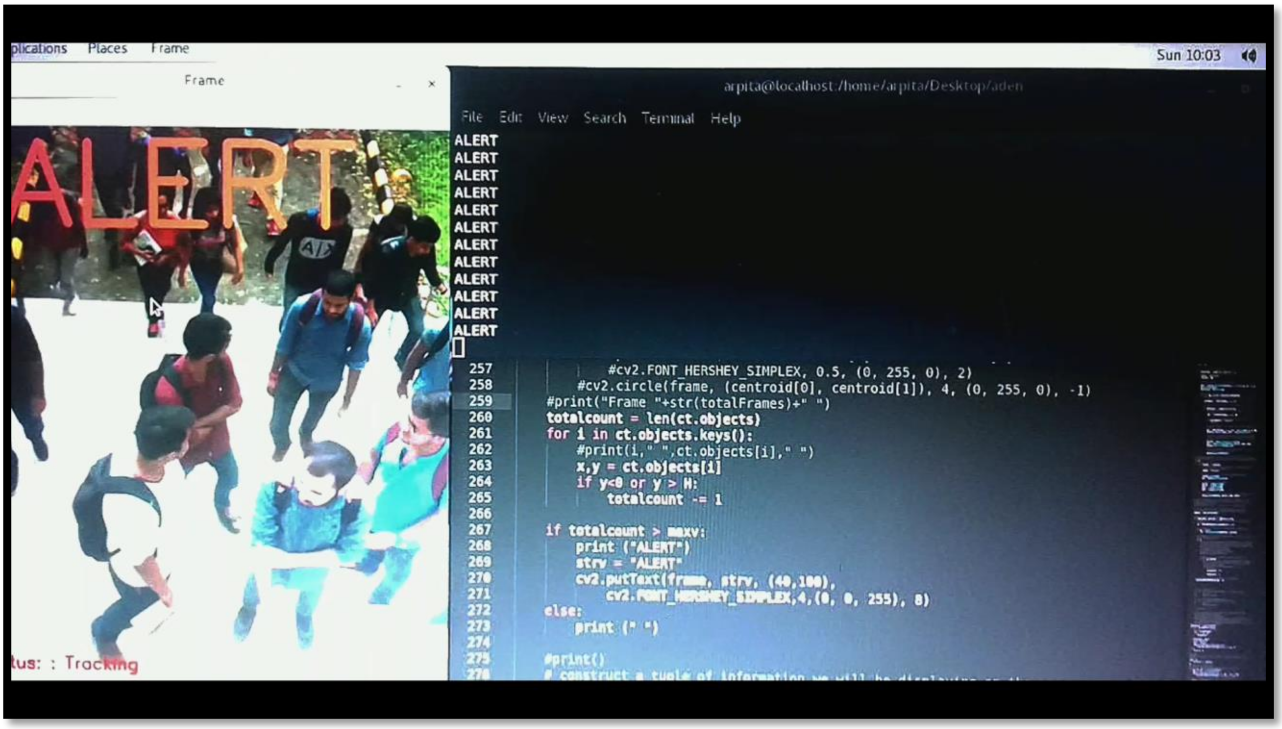


Figure 18: Implementation of Person Counter


plications Places Frame
Sun 10:03
Frame
ALERT
tus: : Tracking
arpita@localhost:/home/arpita/Desktop/aden
File Edit View Search Terminal Help

```
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
ALERT
```

```
                #cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2)
            #cv2.circle(frame, (centroid[0], centroid[1]), 4, (0, 255, 0), -1)
        #print("Frame "+str(totalFrames)+" ")
        totalcount = len(ct.objects)
        for i in ct.objects.keys():
            #print(i," ",ct.objects[i]," ")
            x,y = ct.objects[i]
            if y<0 or y > H:
                totalcount -= 1

        if totalcount > maxv:
            print ("ALERT")
            strv = "ALERT"
            cv2.putText(frame, strv, (40,100),
                cv2.FONT_HERSHEY_SIMPLEX,4,(0, 0, 255), 8)
        else:
            print (" ")

        #print()
        # construct a tuple of information we will be displaying
```

Figure 19: Small Test Run

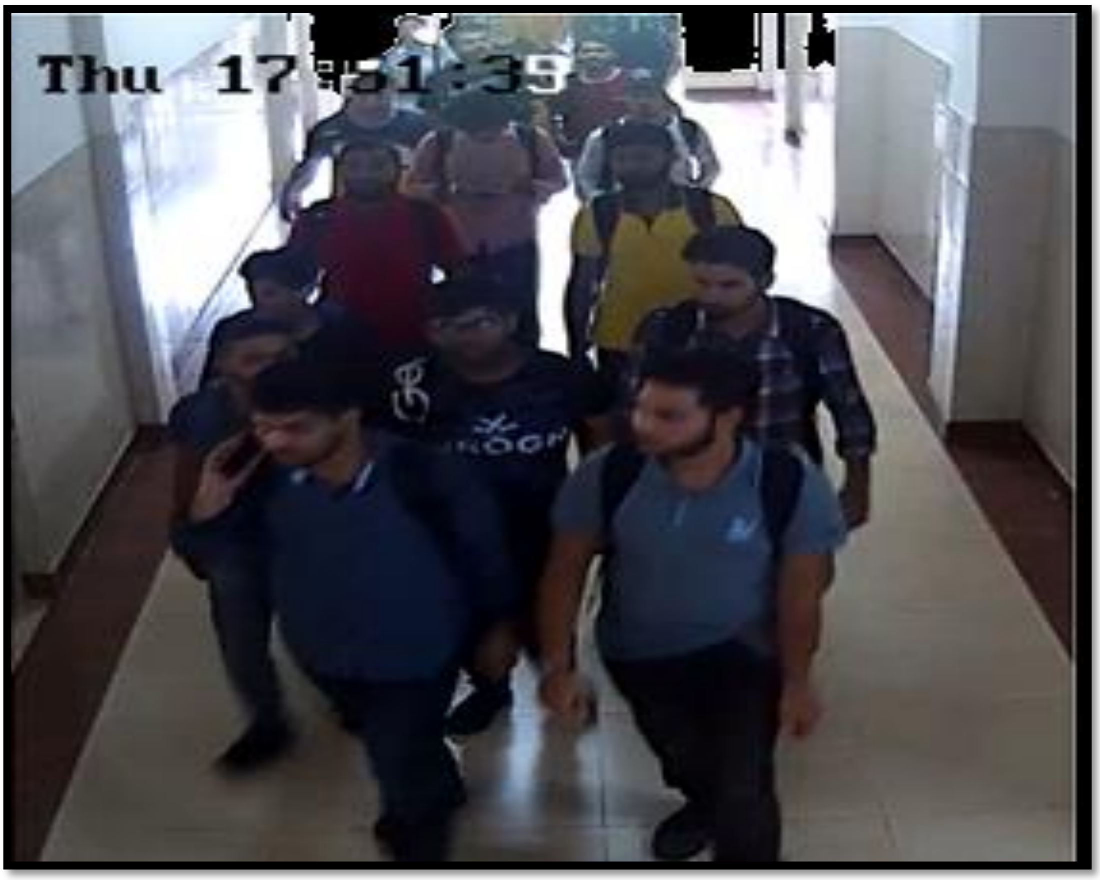
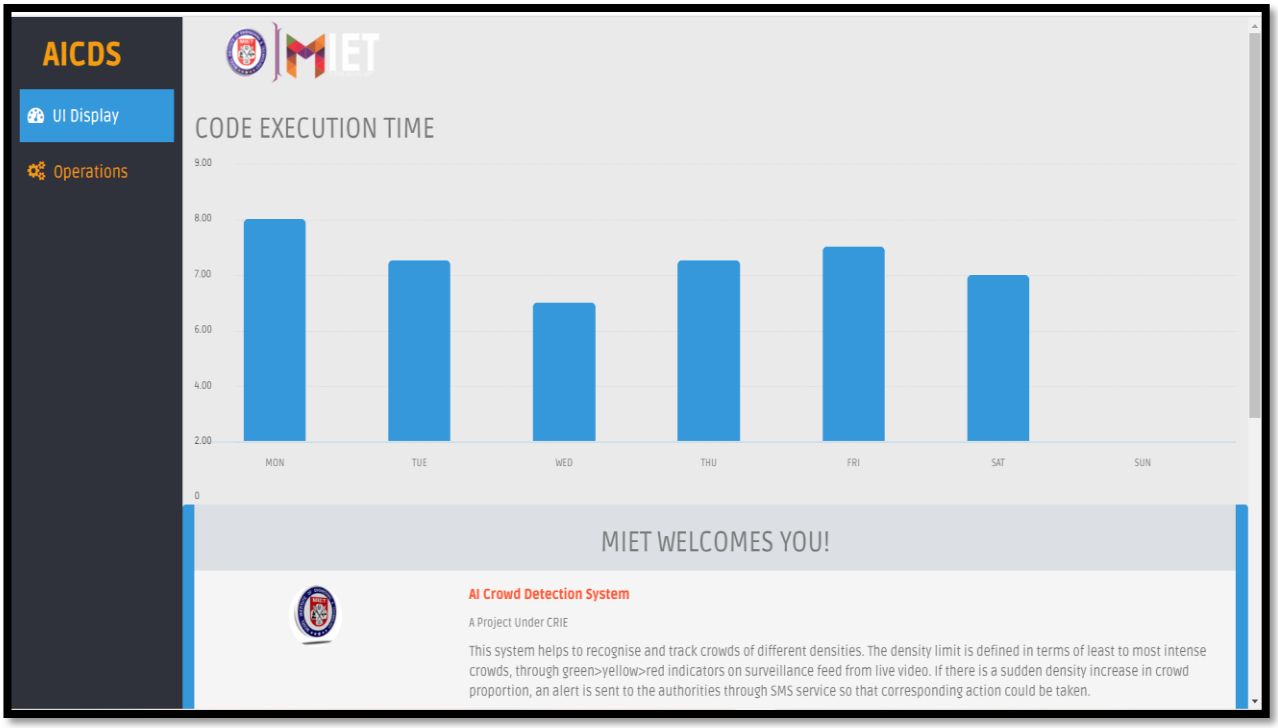


Figure 20: Implementation Large Scale



MIET WELCOMES YOU!

AI Crowd Detection System

A Project Under CRIE

This system helps to recognise and track crowds of different densities. The density limit is defined in terms of least to most intense crowds, through green>yellow>red indicators on surveillance feed from live video. If there is a sudden density increase in crowd proportion, an alert is sent to the authorities through SMS service so that corresponding action could be taken.

Figure 21: CDS Dashboard

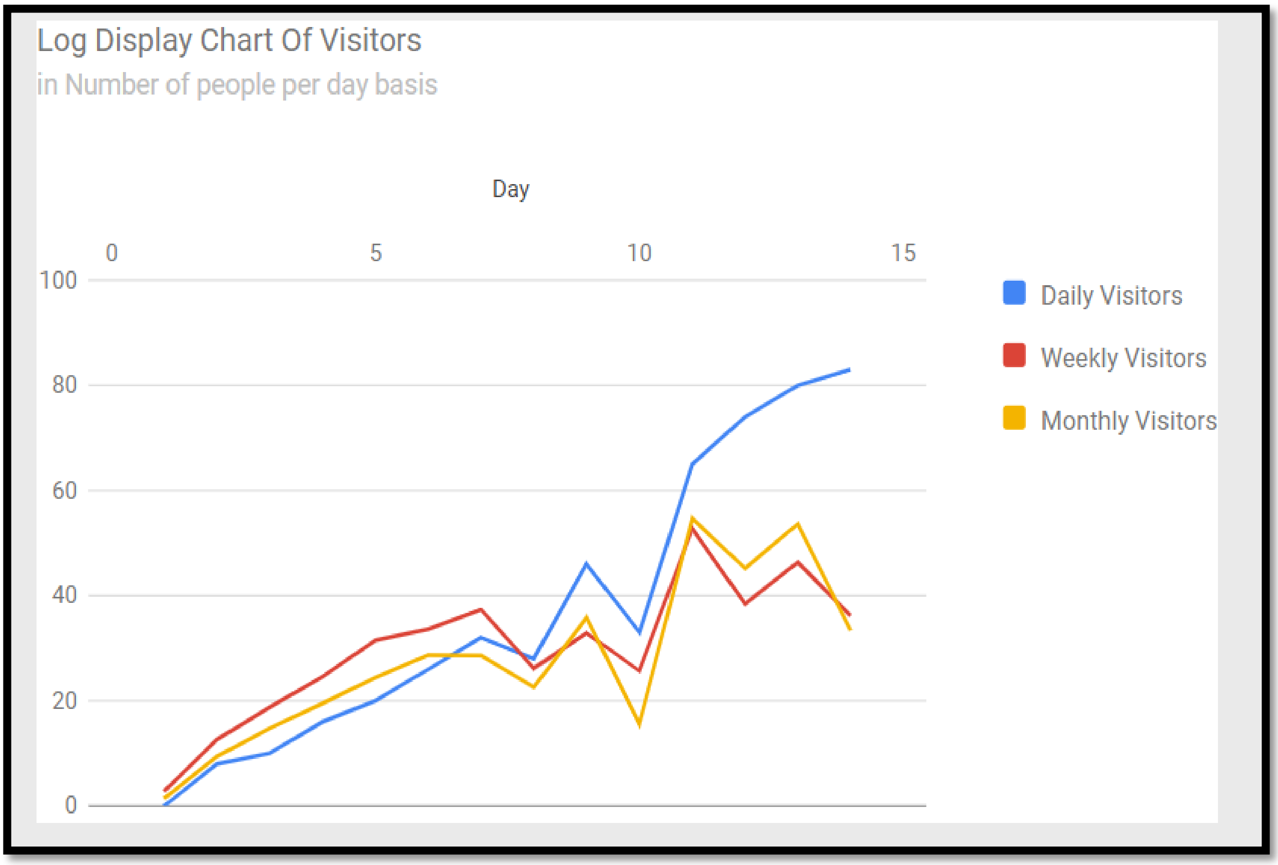


Figure 22: Peak Hour Analytics

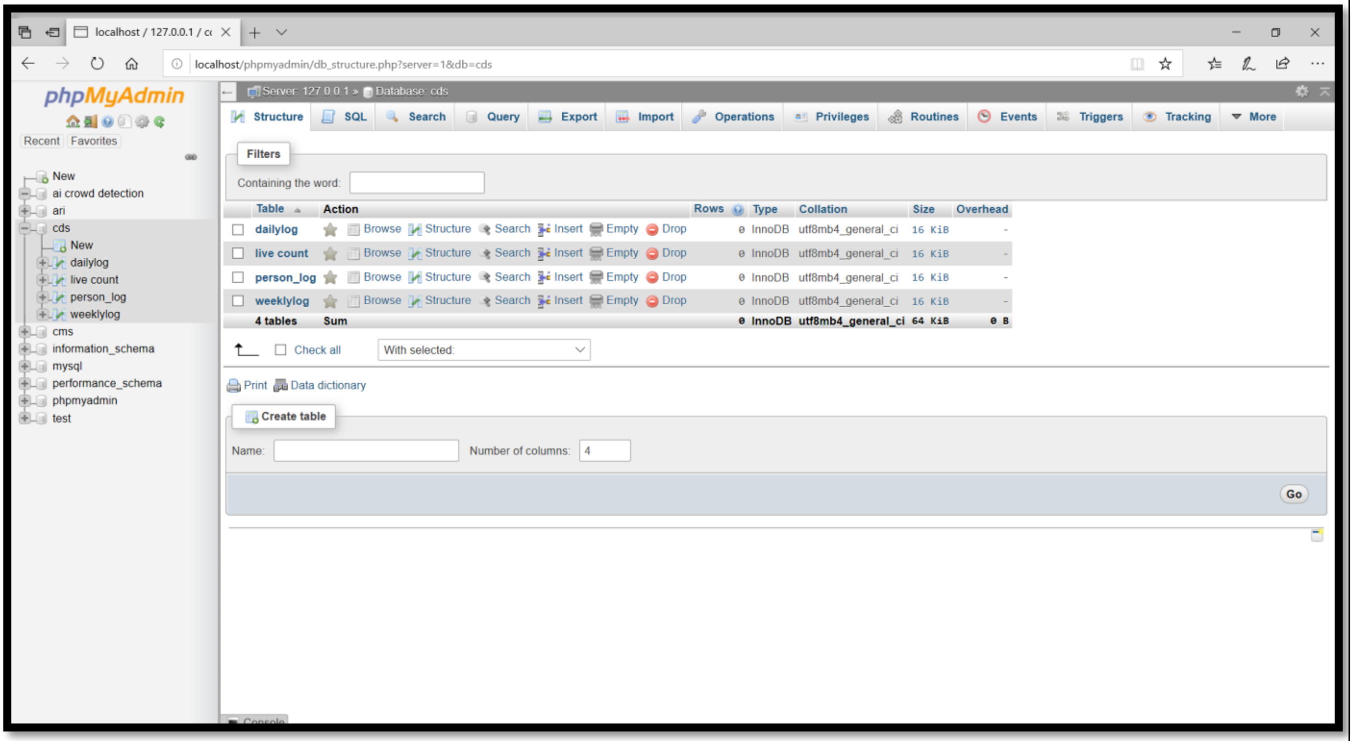


	Table	Action	Rows	Type	Collation	Size	Overhead
□	dailylog	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16 KiB	-
□	live count	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16 KiB	-
□	person_log	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16 KiB	-
□	weeklylog	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16 KiB	-
	4 tables	Sum	0	InnoDB	utf8mb4_general_ci	64 KiB	0 B

Figure 23: Crowd Detection System Database