PROJECT TITLE - Sentinel: Intelligent Multi Camera Face Detection, Recognition And Tracking System For Advanced Video Surveillance

- Batch Number: G17

Roll Number	Student Name
20201CAI0128	RISHI RAGAV V
20201CAI0107	ISRAR AHMED
20201CAI0117	RAKSHITH MB
20201CAI0090	MD FAIZAN USMAN SAIT

Under the Supervision of,

Mr. SK JAMIL AHMED
Assistant Professor
School of Computer Science & Engineering
Presidency University

Abstract

Video and Security surveillance assistant using the current technological advancements like Artificial Intelligence, Machine Learning, Computer Vision and Face Tracking.

Using these technology to create an end to end solution for monitoring and video surveillance.

Literature Survey

- It was found that computer vision is a data heavy and computationally expensive process that required powerful system and high quality as well extensive dataset. Hence models were trained in the server side so that the user can focus on other parts of their project. The application of edge computing where the use of training data locally on the device itself has become popular. To facilitate this process, there has been improvements made in the field of computer vision and machine learning algorithms. One such example is YOLO V8 whose highlighting features are transfer learning, improved loss function and improved speed for data crunching of large datasets. The architecture is given by 8 convolution layers and 4 maxpool layers.

Object Detection Performance Comparison (YOLOv8 vs YOLOv5)

Model Size	YOLOv5	YOLOv8	Difference
Nano	28	37.3	+33.21%
Small	37.4	44.9	+20.05%
Medium	45.4	50.2	+10.57%
Large	49	52.9	+7.96%
Xtra Large	50.7	53.9	+6.31%

*Image Size = 640

Objectives

Primary Objectives

1. Enhanced Face Detection: Develop a highly accurate face detection algorithm that can efficiently identify and locate faces in real-time, regardless of variations in lighting conditions, angles, and facial expressions.

2. Robust Face Recognition: Create a robust face recognition system that can match detected faces to a pre-defined database of individuals, allowing for the positive identification of persons of interest and generating alerts as necessary.

3. Multi-Camera Integration: Implement the capability to seamlessly integrate with multiple cameras across a surveillance network, enabling simultaneous monitoring and tracking of individuals across different camera views.

4. Real-Time Tracking: Develop an intelligent tracking system that can monitor and track recognized faces as they move within the camera network, providing real-time information on their whereabouts and activities.

5. Alert and Reporting System: Design a comprehensive alert and reporting system that can automatically generate alerts for suspicious activities, unauthorized access, or individuals on watchlists, and provide detailed reports for post-incident analysis and evidence gathering.

Optional Objectives

1. Behavioral Analysis: Incorporate behavioral analysis algorithms to detect and alert on unusual or suspicious activities, such as loitering, sudden crowd dispersal, or unusual gestures, enhancing the system's overall security capabilities.

2. User-Friendly Interface: Develop an intuitive and user-friendly interface for security personnel to interact with the system, allowing for easy configuration, monitoring, and response to incidents, and potentially enabling remote access and control for authorized personnel.

Proposed Method

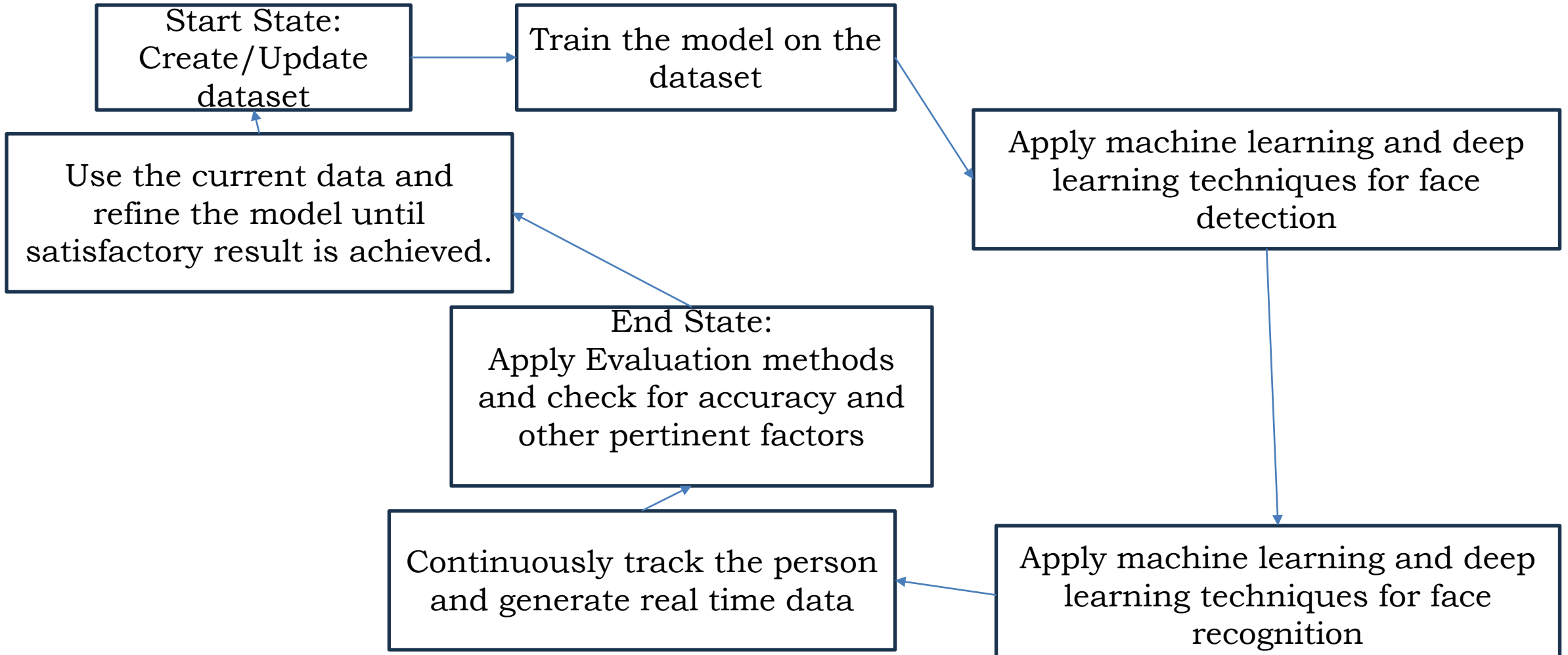


Fig 1. Flowchart of the process

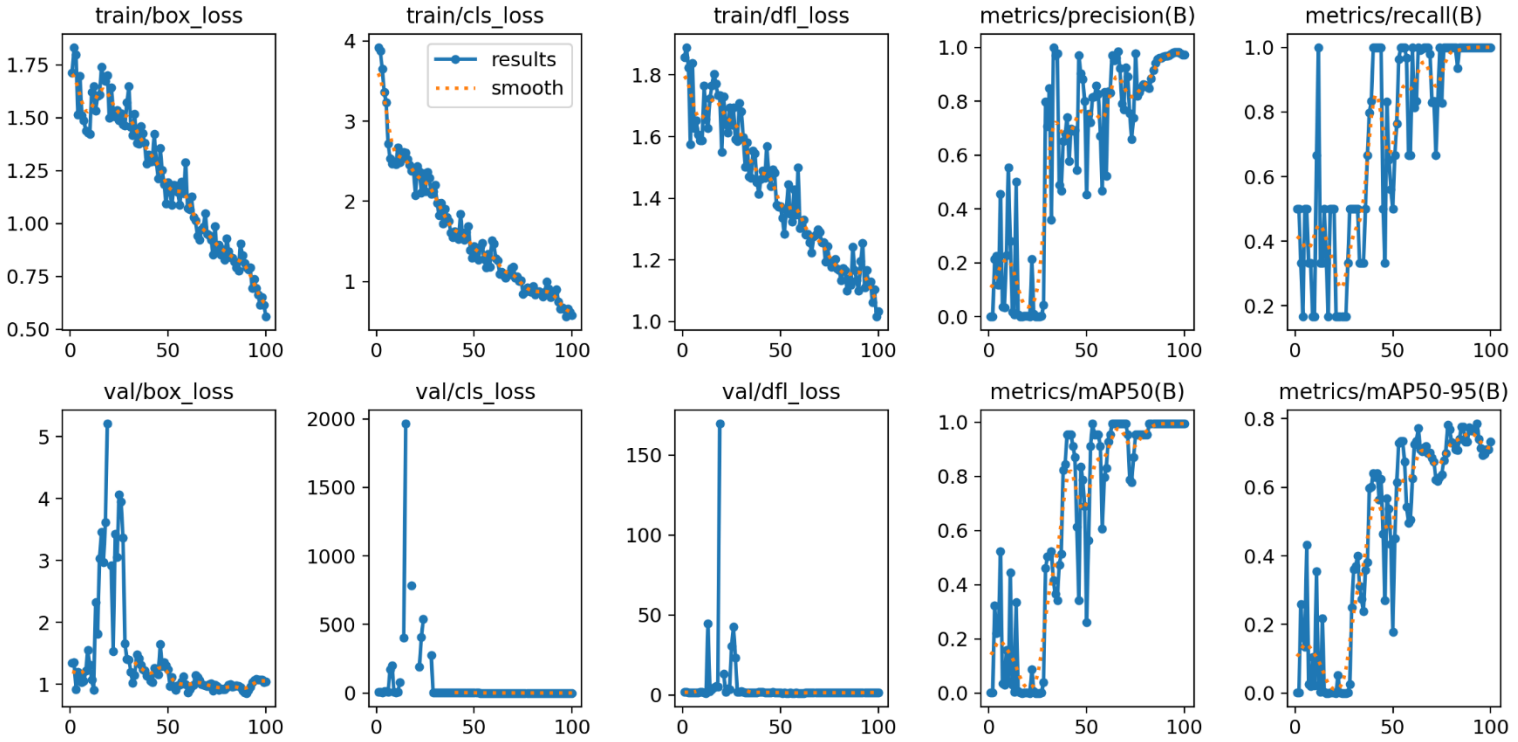
Hardware and Software Details

Hardware Components:
Web camera
Requirement to run the model : 8GB GPU minimum(Tested for yolov8m)
Minimum usage for training: 7.1 GB.

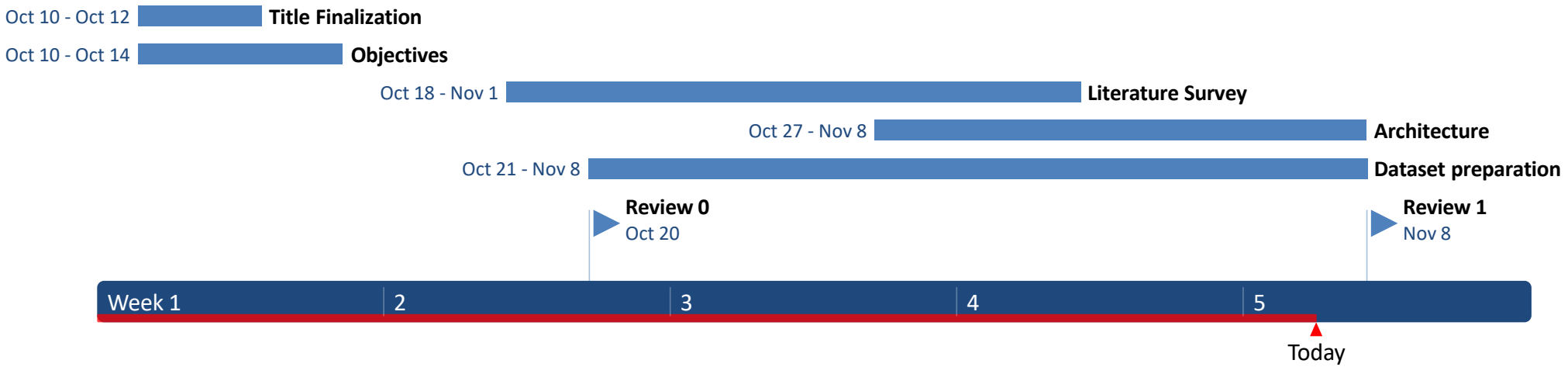
Software Components:
Roboflow – create and instantiating dataset
Visual Studio code – Code Editor
Github – Collaborative coding
Google Colab – Model Training

Minimum images to be trained on 100 unique image per person -> later converted to other types by adding blur, rotate and other types of image manipulation. Final image count: minimum 400 without image manipulation and 1200 after image manipulation
Epochs to be trained on 100-500.
Each epoch duration - 5 to 7 seconds (tested on 28 uploaded image dataset).

Preliminary Result



Gantt Chart



2023

Thank You