Face masked and unmasked humans detection and tracking in video surveillance

Yomna Alayary (*Media Engineering Technology*) and **Prof. Mohammed Salem** (*Digital Media Engineering*)

The German university in Cairo

yomna.alayaryr@student.guc.edu.eg

Motivation

After the spread of COVID-19 virus, masks worn for protection became common practice. However, this made current face detection and tracking technology futile, as masks cover the lower half of the face. This hinders identification and tracking applications. Additionally, governments need to ensure that people are following mask-wearing regulations correctly to prevent further spread of COVID-19 virus.

Figure 1: Example of output of face mask detection model

Problem Statement

I propose improving existing face detection and tracking techniques to take into account the limited visible facial features and instead extracting local binary pattern features from the face's eye, forehead and eyebrow regions.
To do so, a deep CNN model will be created for masked face detection. Moreover, the long term multi-face tracking model will follow a tracking-by-detection approach.

Dataset

For training the face detection model, ISL-UFMD dataset is used, which is a collection of images of people in the 3 categories: masks worn properly, masks worn improperly, and not wearing masks. The images were collected from the internet and other known datasets, amounting to overall 9 main different sources. To improve model's accuracy, data was cleaned by removing any images with unclear ROI caused by blurring or sharp angles.
MaskedFaceNet was also added to increase number of improper mask images.

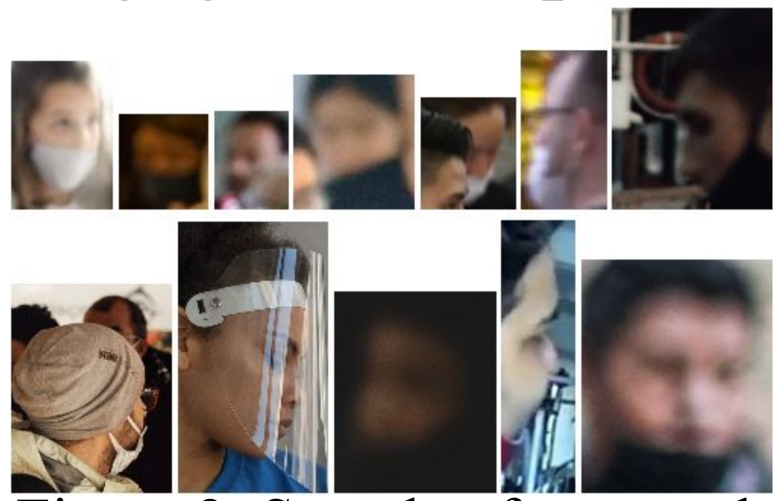


Figure 2: Sample of removed images from ISL-UFMD

Figure 3: Sample of added images from MaskedFaceNet

Methodology

In the methodology, we will create a deep CNN model by combining a masked face detection model and a long-term tracking model.

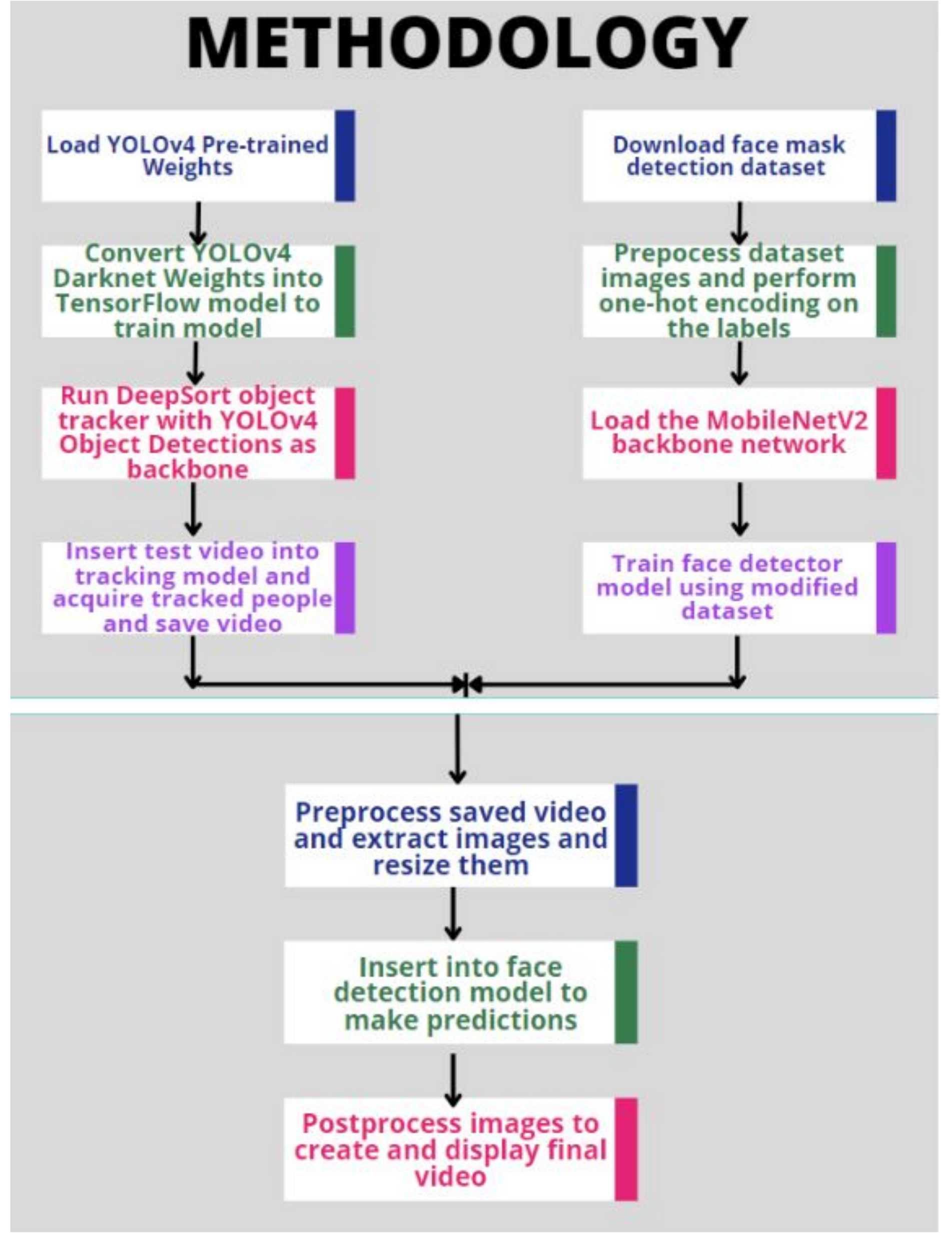


Figure 4: block diagram of methodology

Expected output

When inserting a surveillance video into the completed model, the output video should contain a bounding box tracking each person visible with own identification number and a bounding box around the tracked person's face indicating whether they are wearing a mask correctly, incorrectly or not wearing a mask at all, along with its confidence percentage.

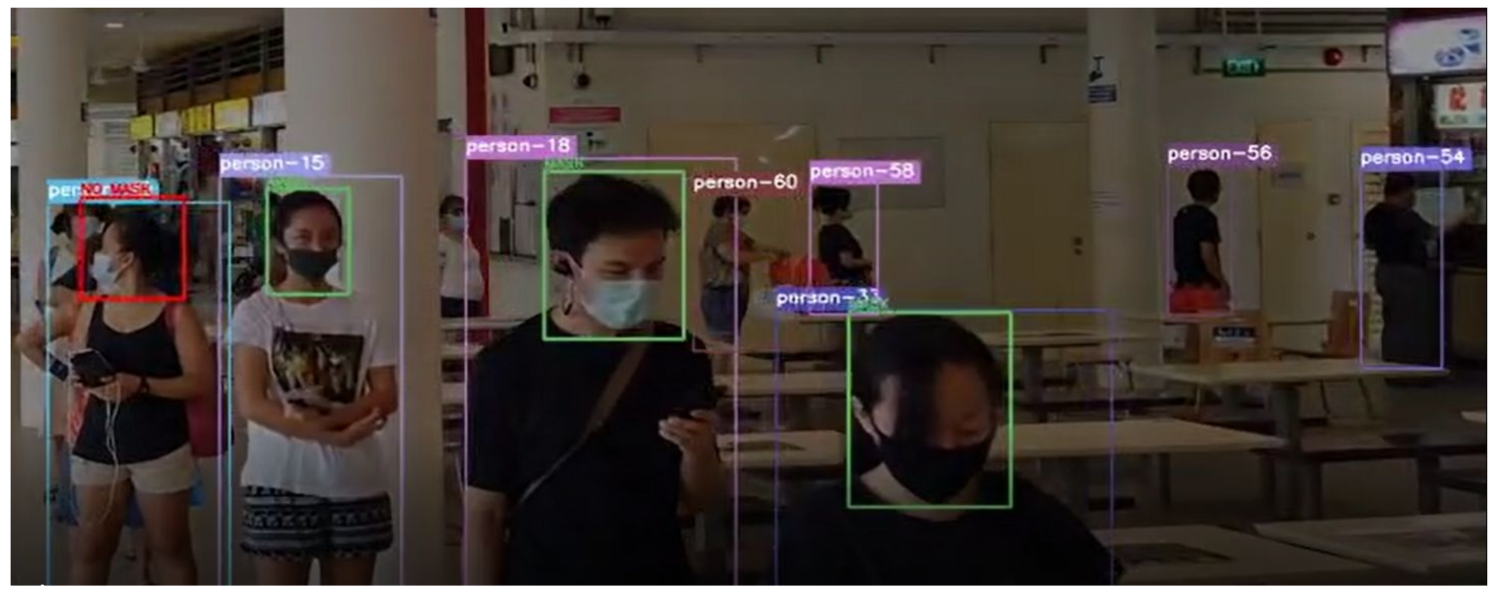


Figure 5: Example of the final output

Results

For the face detection model, when using 2880 images from datasets ISL-UFMD and MaskedFaceNet in training, and utilizing MobileNetV2 backbone, and running on 30 epochs,
the accuracy is 43.06% and loss is 17.7%.

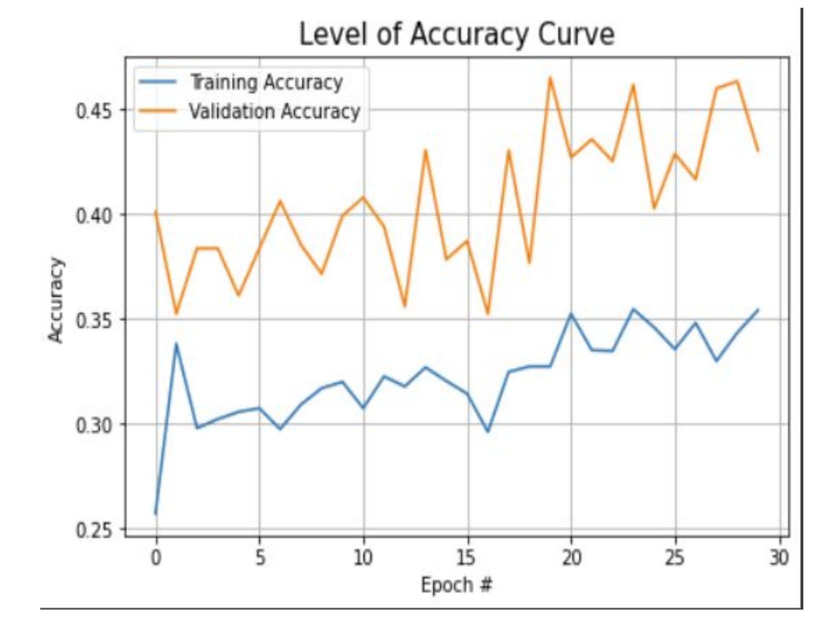
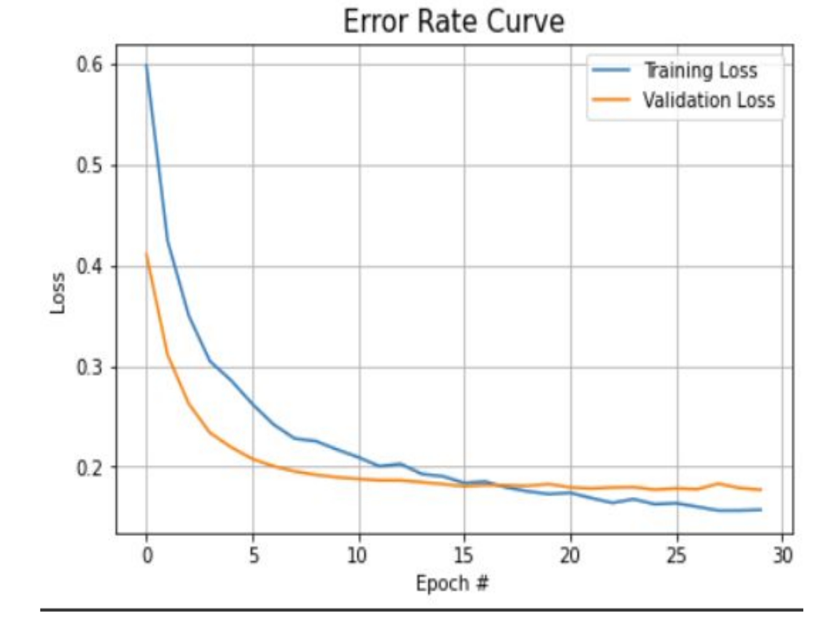


Figure 6 and 7: Accuracy and Error rate curves



When testing on custom images, we get the following results

Figure 8, 9, 10: Results for face mask detector

The object tracking model uses pre-trained YOLOv4 weights. YOLOv4 achieves state-of-the-art results (43.5\% AP) for real-time object detection and is able to run at a speed of 65 FPS on a V100 GPU.

Prepared for Thesis Poster Display Conference

11th -12th June 2022

Faculty of Management Technology
THESIS Poster Display Conference 2022

Face masked and unmasked humans detection and tracking in video surveillance

Bachelor Project 2022
Yomna Islam Youssef Alayary
Faculty of Media Engineering and Technology
German University in Cairo
Supervised by Prof. Mohammed Salem

Contact: yomna.alayary@gmail.com

Motivation

After the spread of COVID-19 virus, masks worn for protection became common practice. However, this made current face detection and tracking technology futile, as masks cover the lower half of the face. This hinders identification and tracking applications. Additionally, governments need to ensure that people are following mask-wearing regulations correctly to prevent further spread of COVID-19 virus.

Figure 1: Example of output of face mask detection model

Problem Statement

I propose improving existing face detection and tracking techniques to take into account the limited visible facial features and instead extracting local binary pattern features from the face's eye, forehead and eyebrow regions.
To do so, a deep CNN model will be created for masked face detection. Moreover, the long term multi-face tracking model will follow a tracking-by-detection approach.

Dataset

For training the face detection model, ISL-UFMD dataset is used, which is a collection of images of people in the 3 categories: masks worn properly, masks worn improperly, and not wearing masks. The images were collected from the internet and other known datasets, amounting to overall 9 main different sources. To improve model's accuracy, data was cleaned by removing any images with unclear ROI caused by blurring or sharp angles. MaskedFaceNet was also added to increase number of improper mask images.

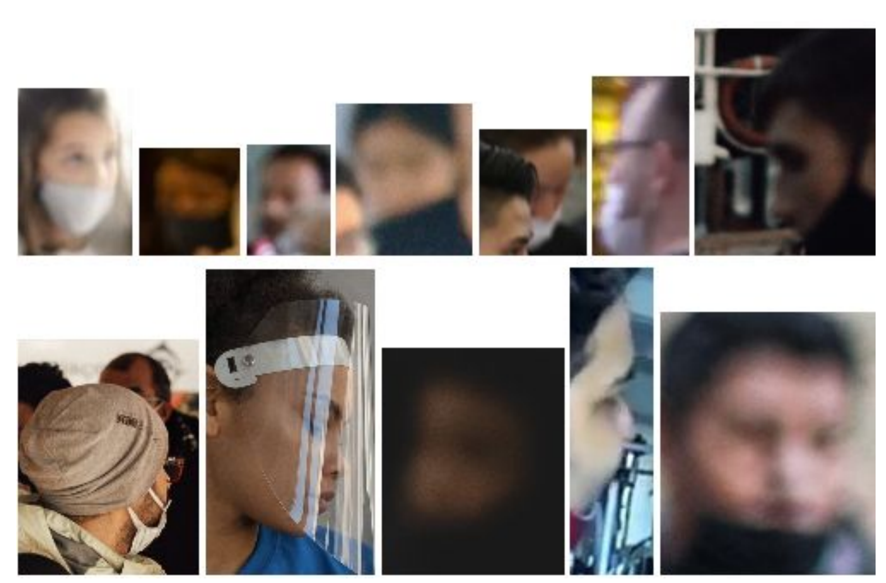


Figure 2: Sample of removed images from ISL-UFMD

Figure 3: Sample of added images from MaskedFaceNet

Methodology

In the methodology, we will create a deep CNN model by combining a masked face detection model and a long-term tracking model.

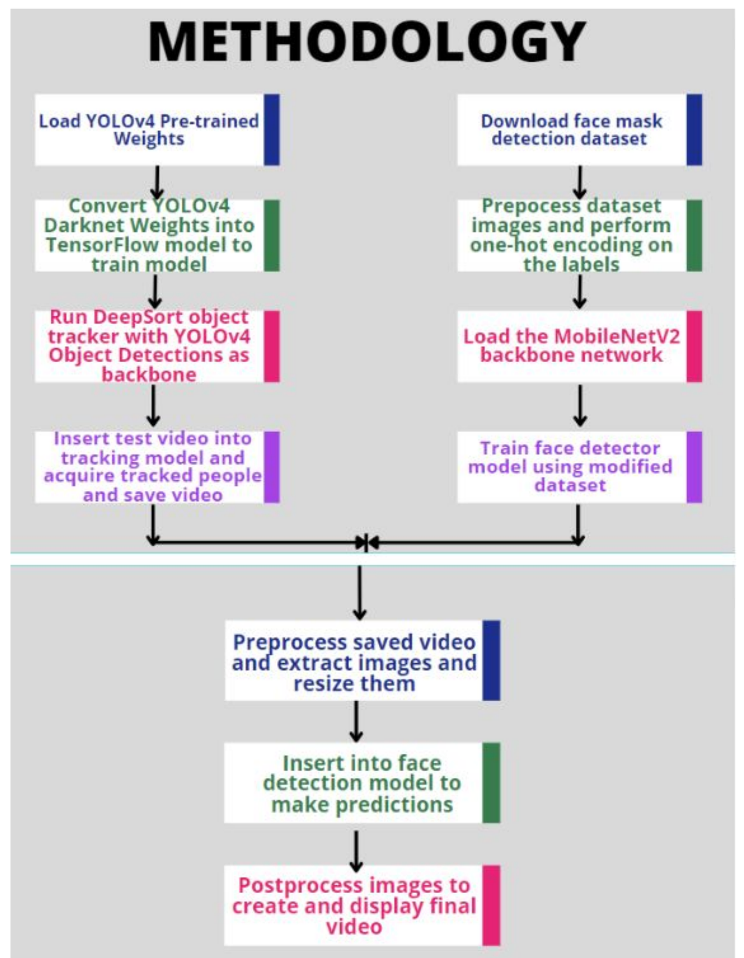


Figure 4: block diagram of methodology

Expected output

When inserting a surveillance video into the completed model, the output video should contain a bounding box tracking each person visible with own identification number and a bounding box around the tracked person's face indicating whether they are wearing a mask correctly, incorrectly or not wearing a mask at all, along with its confidence percentage.

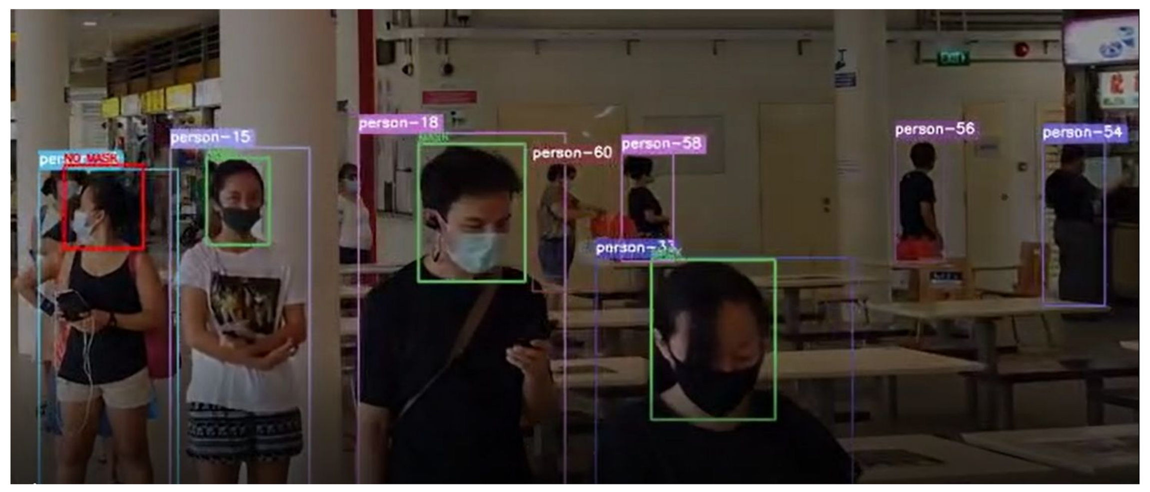


Figure 5: Example of the final output

Results

For the face detection model, when using 2880 images from datasets ISL-UFMD and MaskedFaceNet in training, and utilizing MobileNetV2 backbone, and running on 30 epochs, the accuracy is 43.06% and loss is 17.7%.

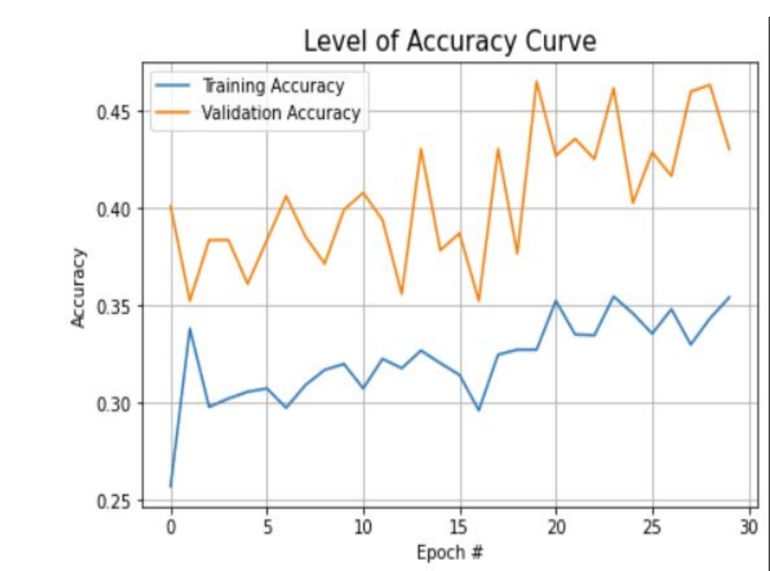
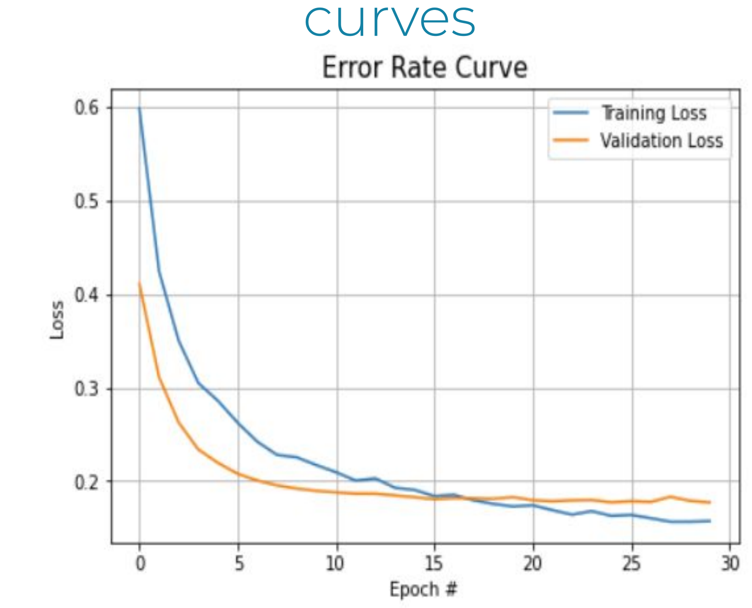


Figure 6 and 7: Accuracy and Error rate curves



When testing on custom images, we get the following results

Figure 8, 9, 10: Results for face mask detector

The object tracking model uses pre-trained YOLOv4 weights. YOLOv4 achieves state-of-the-art results (43.5\% AP) for real-time object detection and is able to run at a speed of 65 FPS on a V100 GPU.