UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	MUHAMMAD OTHMAN BIN MALIKI
Date of Birth	:	
Title	:	SYSTEM SETUP ON JETSON NANO FOR SMART CROWD COVID MONITORING SYSTEM
Academic Session	:	2021/2022

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_______________________ (Student's Signature)

_______________________ (Supervisor's Signature)

____
AMRAN BIN ABDUL HADI

Date: 5 MARCH 2022

Date: 5 MARCH 2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date: 5 MARCH 2022

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

Universiti Malaysia PAHANG

Engineering • Technology • Creativity

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Electronic Engineering Technology in (Computer System) with Hons.

____

(Supervisor's Signature)

Full Name : AMRAN BIN ABDUL HADI

Position : LECTURER

Date : 5 MARCH 2022

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : MUHAMMAD OTHMAN BIN MALIKI

ID Number : TG18015

Date : 5 MARCH 2022

SYSTEM SETUP ON
JETSON NANO FOR SMART CROWD COVID
MONITORING SYSTEM

MUHAMMAD OTHMAN BIN MALIKI

Thesis submitted in fulfillment of the requirements
for the award of the Bachelor of
Electronics Engineering Technology (Computer System) with Hons.

Faculty of Electrical & Electronics Engineering

UNIVERSITI MALAYSIA PAHANG

MARCH 2022

ACKNOWLEDGEMENTS

Alhamdulillah, I am sincerely grateful to ALLAH "S.W.T" for giving me wisdom, strength, patience and assistance to complete my project work. Had it not been due to His will and favor, the completion of this study would not have been achievable.

This thesis would not have been possible without the guidance and the help of several individuals who contributed and extended their valuable assistance in the preparation and the completion of this project. I am deeply thanks to my supervisor of this project, Dr. Amran Bin Abdul Hadi as a Senior Lecturer of Faculty Engineering Technology for her patient, guidance, comment, suggestions, and encouragement which helped me in all the time of this project. Not to be forgotten, all the instructors of Faculty Technology Electrical and Electronic (FTKEE) of University Malaysia Pahang, for giving guidance in making this project success especially in fabrication part.

I am also deeply indebted to my parents Maliki Bin Abdullah and Noor Fadzillah Binti Ibrahim for supporting me continuously and keep giving me advice to be strong handling this project till the end. Lastly, a special thanks to my partner, Muhamad Aiman Bin Suhaizi and Amirul Irfan bin Nazaruddin and all my friends for the help and positives word of successes of this project from the beginning until the last. Thank you very much.

ABSTRAK

The purpose of this project is to develop an affordable IoT-based solution that will improve the safety of the COVID-19 Standard Operating System. COVID-19 poses a significant risk to mankind. The COVID-19 virus is now being fought all over the world. Wearing masks is a sensible strategy for successfully managing COVID-19. This project's goal is to create a simple and efficient real-time monitoring paradigm. The proposed model successfully recognizes if an individual is wearing a face mask or not in a crowded place. The NVIDIA Jetson Nano was used to prototype a real-world scenario using the Object Detection Algorithm (YOLOv5) and an edge AI application: a smart camera capable of estimating the percentage of people wearing face masks in its field of view. The object detection algorithm YOLO, which stands for "You Only Look Once," divides images into a grid system. Each grid cell is in charge of detecting objects within itself. Because of its speed and accuracy, YOLO is one of the most well-known object detection algorithms. YOLOv5 need to be install of all the requirements into jetson nano itself. While this algorithm was accurate, it required too much processing power to run on the Jetson Nano in its current form. The algorithm have to simplified to use a single-shot object detection model as part of the Nano transition, recognizing the following object classes: face with mask, face without mask, and face not visible. The training process by using makesense.ai is a free-to-use online tool for labeling photos. It's ideal for small computer vision deep learning projects, as it simplifies and speeds up the dataset preparation process. Labels can be downloaded in one of the many formats that are supported.

ABSTRACT

Tujuan projek ini adalah untuk membangunkan penyelesaian berasaskan IoT mampu milik yang akan meningkatkan keselamatan Sistem Operasi Standard COVID-19. COVID-19 menimbulkan risiko besar kepada manusia. Virus COVID-19 kini sedang diperangi di seluruh dunia. Memakai topeng ialah strategi yang wajar untuk menguruskan COVID-19 dengan jayanya. Matlamat projek ini adalah untuk mencipta paradigma pemantauan masa nyata yang mudah dan cekap. Model yang dicadangkan itu berjaya mengenali jika seseorang individu itu memakai topeng muka atau tidak di tempat yang sesak. NVIDIA Jetson Nano digunakan untuk membuat prototaip senario dunia sebenar menggunakan Algoritma Pengesanan Objek (YOLOv5) dan aplikasi AI tepi: kamera pintar yang mampu menganggar peratusan orang yang memakai topeng muka dalam bidang pandangannya. Algoritma pengesanan objek YOLO, yang bermaksud "Anda Hanya Lihat Sekali," membahagikan imej kepada sistem grid. Setiap sel grid bertanggungjawab untuk mengesan objek dalam dirinya sendiri. Kerana kelajuan dan ketepatannya, YOLO ialah salah satu algoritma pengesanan objek yang paling terkenal. YOLOv5 perlu memasang semua keperluan ke dalam jetson nano itu sendiri. Walaupun algoritma ini tepat, ia memerlukan terlalu banyak kuasa pemprosesan untuk dijalankan pada Jetson Nano dalam bentuk semasanya. Algoritma perlu dipermudahkan untuk menggunakan model pengesanan objek satu tangkapan sebagai sebahagian daripada peralihan Nano, mengiktiraf kelas objek berikut: muka dengan topeng, muka tanpa topeng dan muka tidak kelihatan. Proses latihan dengan menggunakan makesense.ai ialah alat dalam talian percuma untuk digunakan untuk melabelkan foto. Ia sesuai untuk projek pembelajaran mendalam penglihatan komputer kecil, kerana ia memudahkan dan mempercepatkan proses penyediaan set data. Label boleh dimuat turun dalam salah satu daripada banyak format yang disokong.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

LIST OF ABBREVIATIONS

YOLOv5	You Only Look Once
Wi-Fi	Wireless technology
HDMI	High-Definition Multimedia Interface
SSD	Single Shot Detector
FRCNN	Faster Region CNN
LCD	Liquid crystal display
USB	Universal serial bus
IP ADDRESS	Internet protocol address
-it	Means run in interactive mode
--rm	Will delete the container when finished
device	Allow access to video device

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Project Background

On December 2019, the world was shock with a deadly disease that is very contagious and have taken many lives globally called the COVID-19 The pandemic started in Wuhan, China and rapidly spread throughout the world. COVID-19 is an infectious disease that attacks the lungs causes those infected to feel respiratory illness. Critical patients are put on ventilators that are machines with a tube directed into the patient's mouth to help them to breath normally.

The virus spreads mainly from physical contact and may spread to people that are physically close to one another (1 meter). The wearing of a 3-ply face mask is very important during the pandemic as another way of the virus to spread is through respiratory droplets that are produce when people cough, sneeze, talk and even breathe. Until March 2021, the virus has infected several 130,633,316 people around the globe and 2,845,709 were recorded dead from it. The first case detected in Malaysia was on 25th January 2020 where the patients were a grandmother, 66 years-old, with two of her grandchildren 11 and 2 years-old. All three of them were from China and are families with the first case patient that was also detected positive of COVID-19 in Singapore. This proves two points that are the virus do easily spread among close contacts and the virus can infect people at any age.

At first, the pandemic was taken lightly and unfortunately it spreads so fast until the Malaysia Government announce the first Movement Control Order (MCO) on 18th March 2020. It was indeed a good call but although actions were taken, the numbers of those infected from it is very worrying. Until March 2021, 347,972 people were tested positive, and a total of 1,283 people died from the virus. The following table shows the number of cases monthly in Malaysia from March 2020.

Table 1.1 Total number of new positive COVID-19 cases monthly

Month, year	**Total number of positive cases (people)**
March, 2020	2,816
April, 2020	3,188
May, 2020	1,827
June, 2020	808
August 2020	366
September 2020	1,966
October,2020	20,237
January 2021	103,230
February 2021	84,867
March, 2021	46,672

Numbers do not lie and with the stated statistics, Malaysia is the third country with the most cases among ASEAN country. If we take a closer look to the Table 1.1, we can see that the numbers decrease slowly during the early time when MCO was enforced but on the other hand the MCO itself had impact to the countries financial. The citizens were also affected especially to those who runs their own business. So, the government decided to loosen the MCO gradually from Conditional Movement Control Order (CMCO), Recovery Movement Control Order (RMCO) and areas with serious cases or called as Red Zone are enforced with Enhanced Movement Control Order (EMCO). Although tight SOPs has been imposed by the government, there are still individuals with low self- awareness and those who purposely disobey the laws.

As we support the government's effort and believes that prevention is better than cure, we plan to design a crowd monitoring system that can be installed at various places. It will also detect objects and as for this system it will detect people in a specific place. Crowded places or places with high numbers of people will have a higher risk of spreading of the virus that we do not want to happen and avoid the most.

1.2 Problem Statement

There are already some initiative systems that have been made but there are always room from improvement. We have done observations and studies on previous or current systems, and we have noticed some flaws that can be upgraded. Most places do not have any automated system where people who like to enter a premise must scan their own temperature and write them down on a book provided. Although they are provided with a hand sanitizer, there a still contact where each person who enters the premises needs to use the same pen and writes on the same book. As mentioned earlier, the virus spreads mainly by physical contact. When a premise does have their own automated system, they will depend on a staff to watch over the entrance. This is because people selfish or people with low self-awareness tend to just enter without even bothering to scan their temperature and writing down their name. So, an extra staff is needed just to guard the entrance of a certain premise. Moreover, most places do not even care how many people is in the premise. If the person writes down their name, the person is good to go. This is not a good thing to do as when many people enter a certain premise at a same time, the physical distancing between one another decreases. So, it is very important to have a head count on people who enters any premises.

1.3 Objective

This proposed project is to achieve the following objectives:

1. To design and develop a crowd monitoring system using deep learning method.
2. To detect the usage of facemask of a person in a crowd by using Object Detection Algorithm (YOLOv5).
3. To display the real-time output of the system using wireless method for monitoring purpose.

1.4 Project Scope

In this project, we are designing a system that have the ability to monitor and detect either the person is wearing a face mask or not. The project is only limited to detect three elements or parameters as mentioned earlier. The main aim of this project is to design and fabricate the smart crowd monitoring system using two main equipment or parts that are software and hardware. Both parts must be design properly to ensure the outcome of the system is as preferred.

1.4.1 Software Design

Several software is used mainly to code the hardware for purposed. The software used are as follow:

1) YOLOv5 – To run object detection in the system.
2) Python – To provide coding for object detection.
3) 3Makesence.ai – To provide labels or annotations for the YOLOv5 custom dataset

1.4.2 Hardware Design

Several hardware is used for different purposes in this system. The hardware involved are as follow:

1) NVIDIA Jetson Nano – A computer that acts as the brain of the system.
2) AC8265 Wi-Fi Bluetooth NIC Module – to enable the Jetson Nano to connect to the Internet
3) Webcam Camera – To detect and identify people for the system

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter we will be explaining about a literature review that we have done to choose the best hardware and software components. The most important part is the object detection model. We also investigated several related works and did a study on how to give the best from our system. So as mentioned, this chapter will be divided into three parts: the object detection model, and related works. By the one of this chapter, we will conclude on what is best for our Crowd Monitoring System.

2.2 The object detection model

Real-time object recognition and detection find wide applications in fields as diverse as medical applications, security surveillance and autonomous vehicles. It is one of the difficult tasks because it requires faster computing power to identify the object at that time. However, the data generated by any real-time system that is unlabeled data often requires large amounts of labeled data for efficient training purposes. There are many machines and deep learning techniques used for object detection and recognition. The emergence of complex neural networks (CNNs) has brought great advancement in the detection and recognition of objects [9, 10]. For our literature review on the object detection model, we will be comparing three models that are the You Only Look Once version 5 (YOLOv5), Single Shot Detector (SSD) and Faster Region CNN (FRCNN). After doing depth study on each model, we will come out with one object detection model for the application of our system.

2.2.1 You Only Look Once version 5 (YOLOv5)

YOLO has a unique neural network that predicts bounding boxes and class probabilities directly from the entire image in a single assessment. The R-CNN family has more complex pipelines and is slower than YOLO. In this model, `` You only look once " the image to predict object classes and their positions. YOLO divides the image into grid cells, each grid cell predicts the bounding cells and the confidence scores for those cells. Figure below shows how YOLO works by dividing an image into grid cell for detecting an object based on its trained dataset.

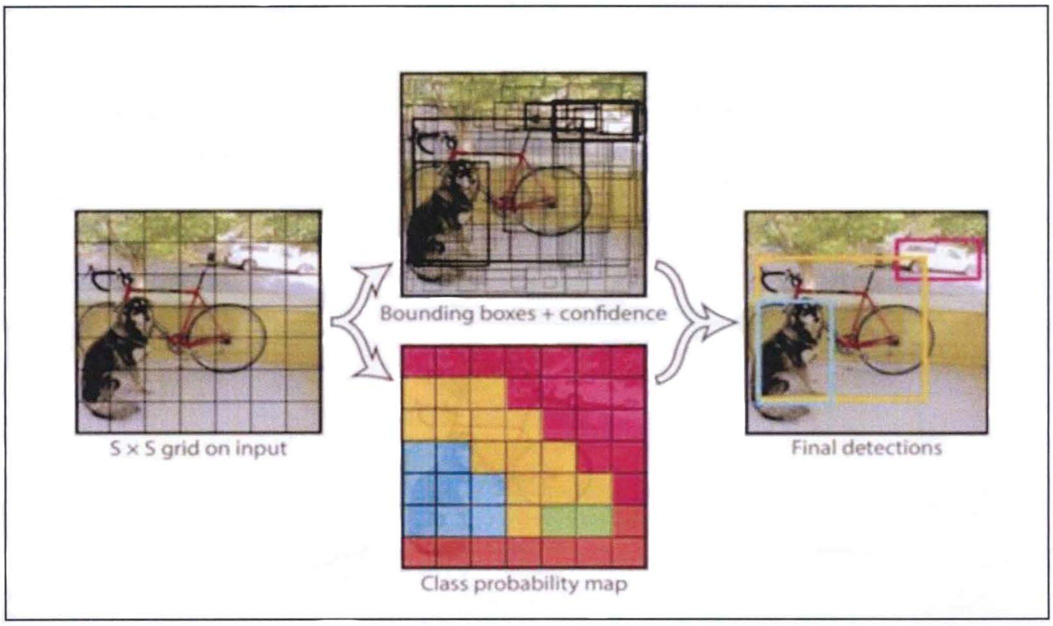


Figure 2.1 YOLO dividing an image into grid cells for object detection

2.2.2 Single Shot Detector (SSD)

SSD uses CNN's pyramid feature hierarchy, which enables efficient object detection and recognition. SSDs offer a good balance between speed and precision. The SSD only uses the CNN in the image once and creates a feature map. Then, convolutional layers are used to predict the bounding box and the detected target object. The SSD uses a VGG-16 model pre-trained on the MobileNet architecture to extract the feature maps. The size of the layers is not fixed, and this helps in detecting objects of different sizes [13]. Figure below shows how a SDD operates. It starts with importing image streams into VGG-16 to extract maps of objects at different scales. Additionally, it predicts object categories and offsets in bounding box locations using built-in filters. The SSD generates a fixed-size collection of bounding boxes and scores for the presence of feature class instances. Then apply the non-maximal removal method and generate the results.

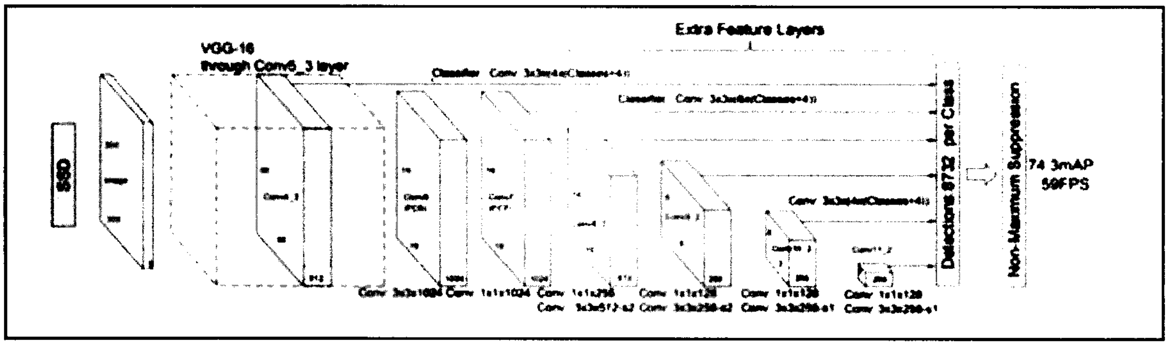


Figure 2.2 The operation of SDD for object detection

A study has also been done in [13] where the results shows that SSD has a better FPS and accuracy than YOLO. This shows that the SSD will have a higher probability of detection. However, the shallow layers of a neural network may not generate enough high-level functionality to make predictions for small objects. As a result, SSDs perform less well for small objects than for larger ones.

2.2.3 Faster Region CNN (FRCNN).

The faster R-CNN object detection network consists of a feature extraction network which is usually a pre-trained CNN, similar to the one we used for its predecessor. Next are two subnets that can be trained. The first is the Area Recommendation Network (RPN), which as the name suggests is used to generate entity recommendations and the second is used to predict the actual class of the entity. So, the main difference for Faster R-CNN is that the RPN is inserted after the last convolutional layer. This is formed to give direct regional recommendations without any external mechanism like selective search. We then use the aggregation of the ROI and the same upstream classifier and the same bounding box regression as Fast R-CNN. Figure below shows a Faster RCNN works by getting the whole image of an object and produces a feature map for object detection.

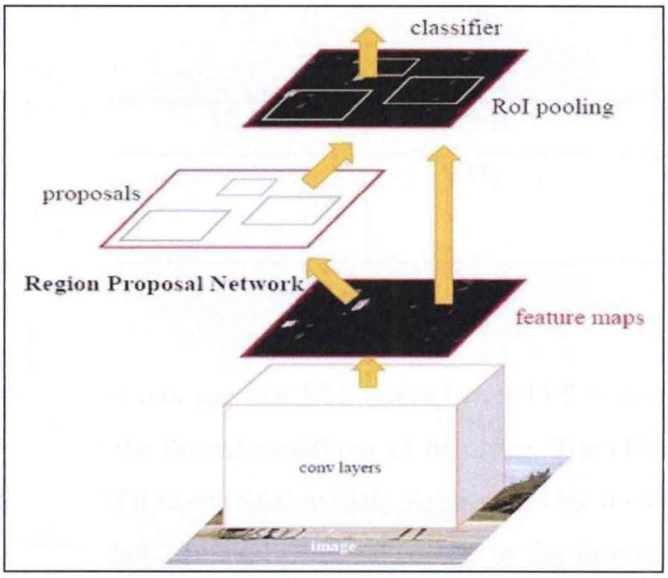


Figure 2.3 The operation of Faster RCNN in image detection

FRCNN uses a region recommendation network named CNN to generate bounding boxes for the target object. This algorithm uses anchor boxes to vary its ability to detect objects with different aspect ratios. They are then passed through several grouping classes to obtain a classifier image. FRCNN is 10 times faster than Fast RCNN with similar accuracy [13]. As for the drawbacks of the Faster RCNN, there might be a small target size that Faster R-CNN cannot detect, which makes the detection rate relatively low [14].

2.2.4 Conclusion from various object detection model

Based on the literature review that has been done, it is quite a difficult to choose which object detection model is the best for our system. Based on [13], The choice of object detection and recognition model depends on the target application. An application that requires more precision but no time, FRCNN will be the best choice. The results from the study that has been made are as below [13]:

Table 2.1 Comparison between YOLO, SSD and FRCNN

	Object detection model		
	YOLO	SSD	FRCNN
Frame per second (FPS)	1.57	3.98	1.39
Runtime (ms)	12.68	16.341	25.943
Probability of detection	95%	97%	100%

From the table we can see that SSD gives better FPS than YOLO and FRCNN. YOLO turned out to be the fastest model out of the three. The FRCNN model provides more accuracy than the YOLO and SSD models. Each model has their very own advantage. As the study has stated that The YOLO model proves to the fastest and easiest of three models [13], we will choose to apply the YOLOv5 in our crowd monitoring system as it has the fastest runtime and easiest configuration among the others. The percentage of probability of detection is still high so we consider it acceptable.

2.3 Related works

The last criteria that had investigated before constructing system is the study on existing related works. But doing this, we will get to know what system that has already been done so we would not build a same system. This part of study will also give us the idea on what to improve from existing systems. There are three systems that we have investigated that are the Wearable Social Distancing Detection System [15], IoT-Based COVID-19 SOP Compliance and Monitoring System for Businesses and Public Offices [16] and lastly Face Mask Recognition System with YOLOV5 Based on Image Recognition [17]. After completing the study on all three existing systems, we will come out with the best way to construct our own system.

2.3.1 Wearable Social Distancing Detection System

In this system, a portable social distance detector that uses a microcontroller with an ultrasonic sensor is used to detect the distance between two people and issue an alert if the person breaks the rules. The system can perform social distance detection accurately and can help the movement of people in an area. Figure below shows the hardware of the system. It is worn by strapping the system to a user's body with the ultrasonic sensor facing backwards.

Figure 2.4 Prototype of the ultrasonic sensor system

The system works by sensing a person at the back of the user. It means that the user will have to be responsible for the person in front and beside of the user by minding his or her gap as the system only detects if a person comes too close to the user from the back. Once the system is turn on, the ultrasonic sensor will continuously transmit an ultrasonic to detect either there is a person or not. The output of the sensor will be sent to a STM32F446 microcontroller. The microcontroller will do the distance calculation between the user and anyone from behind the user. If the sensor does no detect anyone, the microcontroller will send a signal to the LCD to give an output of "SAFE DISTANCE". If a person is present from the user from behind in less than 1 meter, the system will trigger a buzzer as a waning sound and the LCD will show "WARNING, STEP BACK" so that the person that is too close to the user moves away for a safer physical distancing. The figure below shows the flowchart of the system explaining in a simple way on how the system works.

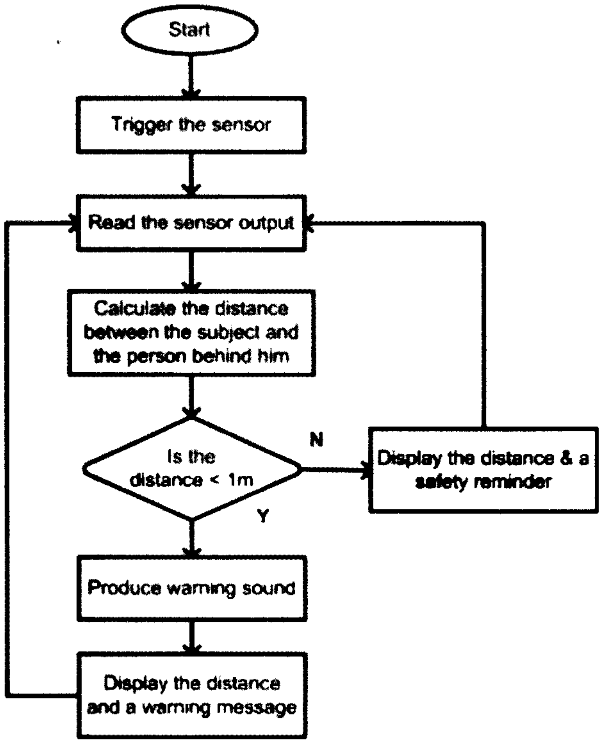


Figure 2.5 The flowchart of the ultrasonic sensor system

2.3.2 IoT-Based COVID-19 SOP Compliance and Monitoring System for Businesses and Public Offices

The next system is applied in the whole area of a certain location. The system starts with a questionnaire that is needed to be answered before entering the premise. the questionnaire is in a digital form so it can be done via phone. Upon entering the premise, an Infrared obstacle detector will detect the person. There will be two of these sensors installed that are at the entrance of the premise and at the exit of the premise. By doing this, the number of people in a particular premise can be monitored. A non-contact IR temperature sensor MLX90614 sensor is installed just after the entrance of the premise. People will not need to go and scan their temperature manually as it is automatically done by the system. At the queue area of the premise, there is an addressable LEDs WS2812B installed to visually display the permissible queueing location. A time of flight (ToF) sensor VL531x is installed at the counter to measure the distance between a person and the counter. The whole premise will be monitored by a Raspberry PI camera located near the queueing area so anyone who violates the SOP will be acted.

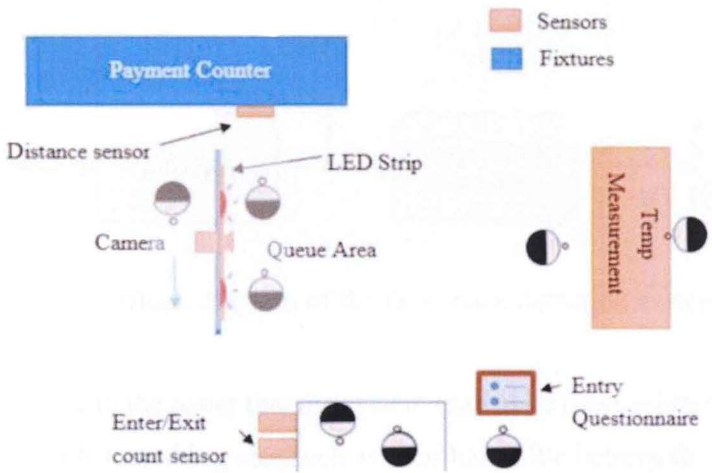


Figure 2.6 Layout of the system

Figure above shows the layout of the premise including all the sensors as explained earlier. In this system all sensors are connected to different ESP8266. Then All ESP8266 transmits the data to the Raspberry PI server using WebSocket communication protocol. By doing this, the system can be set link to Blynk app for IoT application.

2.3.3 Face Mask Detection using YOLOv5 for COVID-19

The last and final existing related work that we have study is a very straight forward system yet very efficient. The system starts with a camera that is installed at an entrance of a mall. The camera will capture the image and send the image to the interface. Then a face mask image recognition is run. In other words, the system will identify either the person is wearing a facemask or not. The system is then connected to the door or the gate of the mall. If the person is wearing a face mask, the system will trigger the door to open and if the person does not wear a mask the door or gate will not open. By implementing this system everywhere, we can achieve a fully face mask environment as everyone knows that without wearing a face mask, there is no use going anywhere. The system uses a YOLOv5 which is an excellent object detection model so it can be said that the system will recognize a 95% or mask wearer. Figure below shows the block diagram of the system as explained earlier.

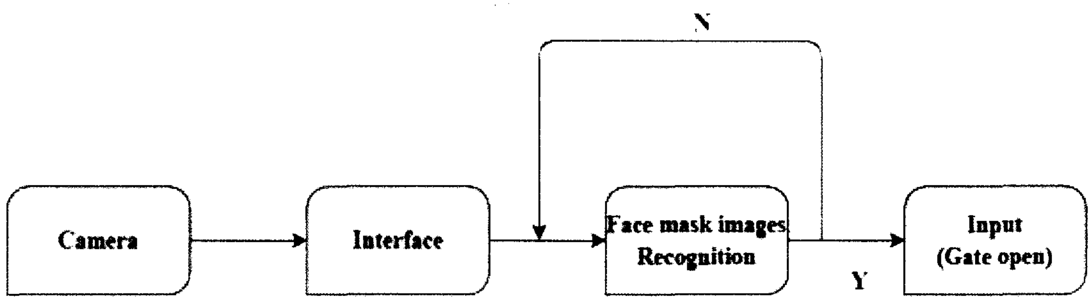


Figure 2.7 Block diagram of the face mask detection system

However, it stated in the paper that the system may have flaws when people tend to cover their facemask with something else such as their hand. We believe that this problem can be overcome by doing a custom dataset. We would only need to train a new dataset for the YOLOv5 with a label of mask covered. Note that the larger the dataset, the greater the YOLOv5 can do object detection.

2.3.4 Conclusion from various existing related works

The first, system is a good application for physical distancing, but it is too simple. Firstly, physical distancing is to be practice form all direction not only from the back. Secondly, a 1-meter gap is not safe if there is a person without a facemask as they may sneeze or cough and the safer distance if that happens is at least 2 meters. The system is also not applicable if the premise has already too many people. It will be very hard to maintain a safe distance between one another. So, for us, it is essential to first limit the number of people inside the premise so that a safe physical distancing can easily be practice. It is also important to filter or keep a person without a mask from entering a premise as it is too risky for other people and a very irresponsible act. This kind of person must not be allowed to enter any premises at all.

The second system is also a very good system wear lots of parameters have been taken into account. There are still room for improvement for the second system where a facemask detector can be installed at the entrance of the premise. this is because a usage facemask is very essential during this pandemic. The LEDs WS2812B that are installed to visually display the permissible queueing location is too much. Note that it cost almost RM65 per item. The price is high as it is not a regular LED but is an LED that can be programmed. Tapping the floor with masking tape can also make people aware of the safe physical distance. However, the camera can still be maintained in the system for monitoring purposes.

Lastly, the third system is almost the same as the first system. It only focuses on one detection and ignoring other important matters. There is no need to eliminate the system, but it is better to improvise the system with other features too such as a temperature detector. This will not only make the environment safe from non-mask people but can also filter out people with a high fever from entering a particular place. Note that having a fever is the most common symptom of COVID-19.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The COVID-19 pandemic has created a need for understanding the number of individuals wearing or not wearing face masks in public areas. The application, Smart Crowd Monitoring System, addresses this need by detecting masked and unmasked individuals in a broad area of coverage.

Smart Crowd Monitoring System is intended to help both public and private organizations understand mask compliance. For example, a local transportation authority might use Smart Crowd Monitoring System to understand the percentage of people wearing masks on a train station platform at rush hour. Similarly, it can help a store manager see the week-over-week increase in mask wearers after they implement a "masks required" policy. Ultimately, Smart Crowd Monitoring System allows business owners, public officials, and other users to be better informed in their efforts to limit the spread of COVID-19.

To address the demands of the application, this application identified both top-level and detailed requirements, and from those, this Smart Crowd Monitoring System had successfully created the product specifications.

3.2 Application Design

Figure shows the block diagram for the Smart Crowd Monitoring System. All hardware is presented in the block diagram where the process starts with a webcam camera collecting the input for the system. The input is images and video or in system, the real-time video. The real-time video is just another term for a live camera. The input will then be sent to an NVIDIA Jetson Nano where a YOLOv5 data frame is run on it.

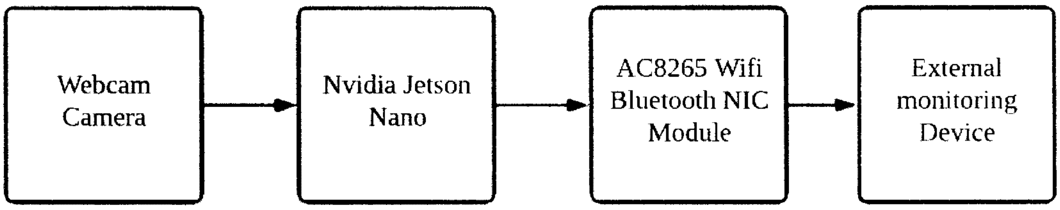


Figure 3.1 Block diagram of the face mask detection system

You Only Look Once (YOLO) is a real-time object detection framework. The ability of the YOLOv5 is used to detect a face mask on a person and will also detect the numbers of person present at a particular time.

Initially, the Jetson Nano does not come with an internet receiver. There are two ways to connect the Jetson Nano to the internet or network that is by either using a Wi-Fi Network Card Module or Wi- Fi USB Dongle. In the system, will use a Network Card Module that is the AC8265 Wi-Fi Bluetooth NIC Module for Jetson Nano. The reason why the system has to install the module is because its need to designing the system with an ability to monitor a particular place anywhere if the user has internet connection. By only entering the IP address of the Jetson Nano on another computer, the current situation can be monitored wirelessly from around the world

3.3 Application Design

In figure, the main jetson nano where it will give the instruction for webcam to collect the data of image or video in real-time. Next, the data will be sent to Nvidia Jetson Nano to store and saved. The YOLOv5 algorithm will run on Jetson Nano. The system will indicate or monitor either people wear a mask or not. Green box would appear if the person wore mask properly, but red box is against it. Data from Nvidia Jetson Nano would send an output via Wi-Fi. As for results, an output data display can be monitoring by using host's PC or smartphone. System was repeat in many times of frame.

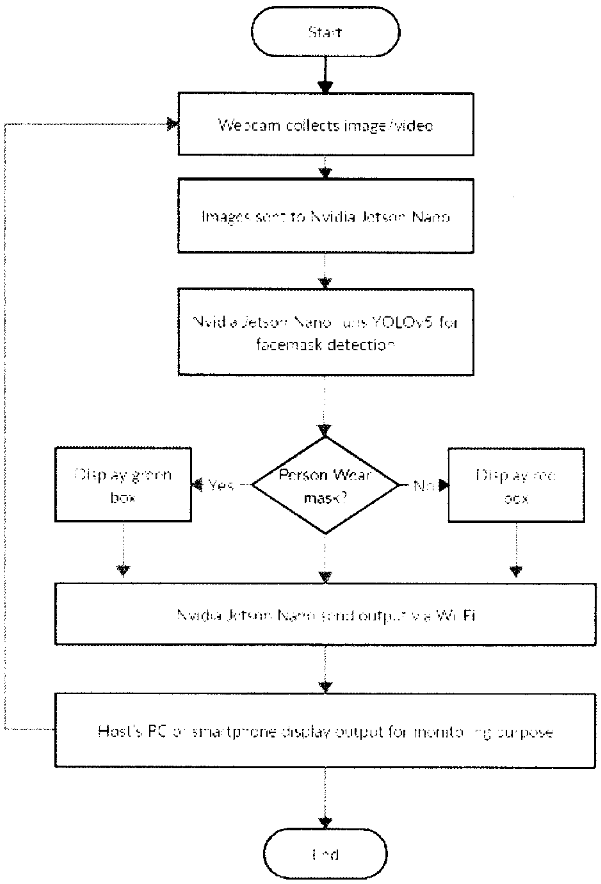


Figure 3.2 Flow chart of Smart Crowd Monitoring System

3.4 System Hardware

The Smart Crowd Monitoring System consists of three main components shown in Fig. 3.4.1. The first component is Nvidia Jetson Nano Developer Kit. Nvidia Jetson Nano were connected to computer display (HDMI or DP). The Jetson Nano Developer Kit uses a microSD card as a boot device and for main storage.

The second component is AC8265 Wi-Fi Bluetooth NIC Module with M.2 interface. It was connected to the Jetson Nano via Wi-Fi and Bluetooth for wireless communication. The antenna can be attached to acrylic case, and the wireless card is screwed to the Jetson Nano board. It has 5G/2.4G dual band Wi-Fi, wide network coverage, and a 3db large gain dual antenna, so it can have good wireless penetration even in a complex wireless environment with multi-interference, making data reception smoother. This wireless network card does not require the installation of any drivers before it can be used, and Bluetooth and Wi-Fi will be automatically detected. The two antennas can also rotate within 90 degrees.

The third component is webcam. USB were connected the to the monitor. It will detect facemask or non-facemask image in real time and able to transmit the result to monitoring devices such as host's PC or smartphone.

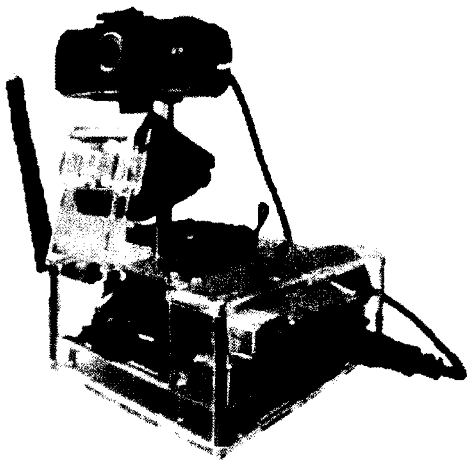


Figure 3.3 System Hardware

3.5 Detection Algorithm

While this algorithm was accurate, it required too much processing power to run on the Jetson Nano in its current form. Simplified the algorithm to use a single-shot object detection model as part of the Nano transition, recognizing the following object classes:

- Face with mask
- Face without mask
- Face not visible

The face not visible class is required for tracking, as the system need to track heads even when they are not looking at the camera.

Next, manually created a dataset of mask training images by downloading images from the Internet and extracting frames from downloaded videos because there was no previous object detection dataset with these categories. MakeSense.AI, an open-source image labelling tool, was used to label the images. This process used dataset for the majority of the development process because it contains JPEimages and Anotations with 1,706 across all classes. The result is shown in Figure. This dataset were expanded after the development process was largely completed by combining it with other publicly available sources.

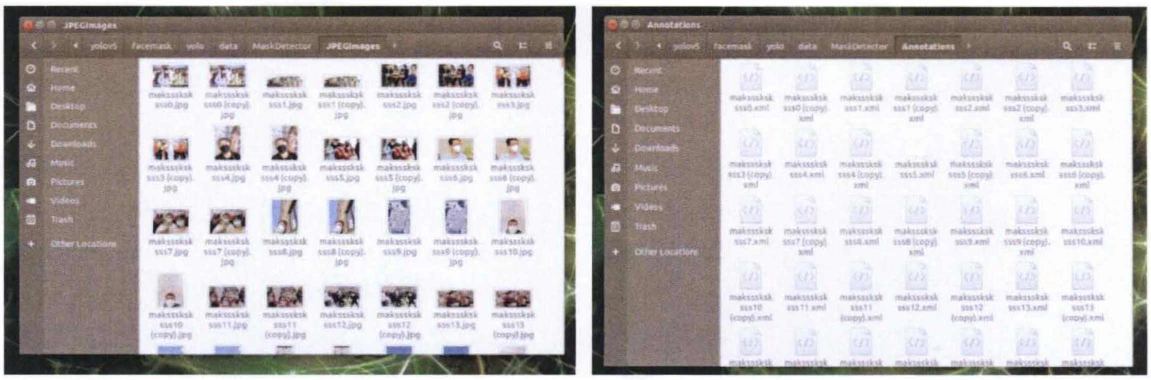


Figure 3.4 Annotation & JPEG Images of dataset

3.6 Embedded Program

Based on Figure 3.4, shows the code for Jetson Nano streaming in real-time and have connection between the input which are the webcam by using USBCAM protocol. The Visual Studio Code software were used to implement for running coding on Jetson Nano. Microsoft's Visual Studio Code is a source-code editor for Windows, Linux, and macOS. Debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and integrated Git are among the features. Users may customize the theme, keyboard shortcuts, and preferences, as well as install extensions that offer new features.

```
from maskcam.prints import print_run as print
from maskcam.config import config, print_config_overrides
from maskcam.common import USBCAM_PROTOCOL, RASPICAM_PROTOCOL
from maskcam.common import (
    CMD_FILE_SAVE,
    CMD_STREAMING_START,
    CMD_STREAMING_STOP,
    CMD_INFERENCE_RESTART,
    CMD_FILESERVER_RESTART,
    CMD_STATUS_REQUEST,
)
```

Figure 3.5 Webcam program code

For the mask detection monitoring process, Nvidia Jetson Nano will determine percentage of mask wearers in an area from a surveillance camera perspective. The system will track the number of people in a large viewing area (90°+ horizontal field of view, at up to 35 feet from the camera). Also, the system will determine whether individuals are masked or unmasked. The streaming process will display on stream camera view with detection results and live statistics over IP. Figure 3.6 shows the code for the statistic to detect the face mask detection.

```python
def is_alert_condition(statistics, config):
    # Thresholds config
    max_total_people = int(config["maskcam"]["alert-max-total-people"])
    min_visible_people = int(config["maskcam"]["alert-min-visible-people"])
    max_no_mask = float(config["maskcam"]["alert-no-mask-fraction"])


    without_mask = int(statistics["people_without_mask"])
    with_mask = int(statistics["people_with_mask"])
    visible_people = with_mask + without_mask
    is_alert = False
    if statistics["people_total"] > max_total_people:
        is_alert = True
    elif visible_people >= min_visible_people:
        no_mask_fraction = float(statistics["people_without_mask"]) / visible_people
        is_alert = no_mask_fraction > max_no_mask

    print(f"[yellow]ALERT condition: {is_alert}[/yellow]")
    return is_alert
```

Figure 3.6 Alert condition program code

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter will be focusing the real-time result obtained during the project period. The project of implementing real-time for Jetson Nano for facemask detection in a crowd place. This project is to verify and analyses the effectiveness of implementing monitoring system. The result will behighlighted in this chapter.

4.2 Running System on a Jetson Nano

Firstly, based on Figure 4.1; we use software Putty on the laptop to run the system by via-Wi-Fi or using IP address. The reason we use Putty because we want eliminates the usage of cables that are not necessary to be use. So, figure 4.2.1 below show the interface of the Jetson Nano by using Putty for monitoring system.

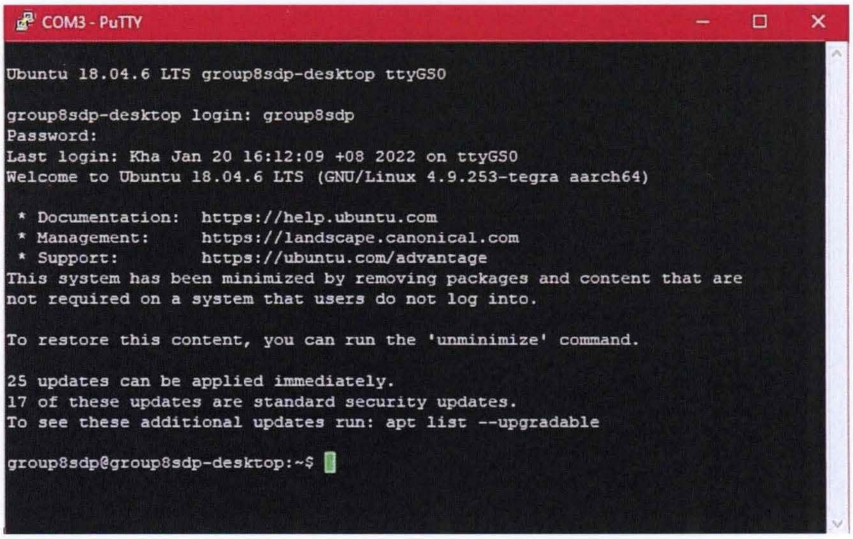


Figure 4.1 Software Putty Setup

Figure 4.2 IP address command

Based on Figure 4.2, the first step to run the system are we need find local Jetson Nano IP address using ifconfig. This address will be used later to view a live video stream from the camera and to interact with the Nano from a web server. Next, make sure a USB camera is connected to the Nano, and then start the system by running the following command:

sudo docker run --runtime nvidia --privileged --rm -it –en MASKCAM_DEVICE_ADDRESS=<your-jetson-ip> -p 1883:1883 -p 8080:8080 -p 8554:8554 maskcam/maskcam-beta

The system container should start running the maskcam_run.py script, using the USB camera as the default input device (/dev/video0). It will produce various status output messages (and error messages if it encounters problems). If there are errors, the process will automatically end after several seconds.

Otherwise, after 30 seconds or so, it should continually generate status messages (such as Processed 100 frames...)

4.3 System Output

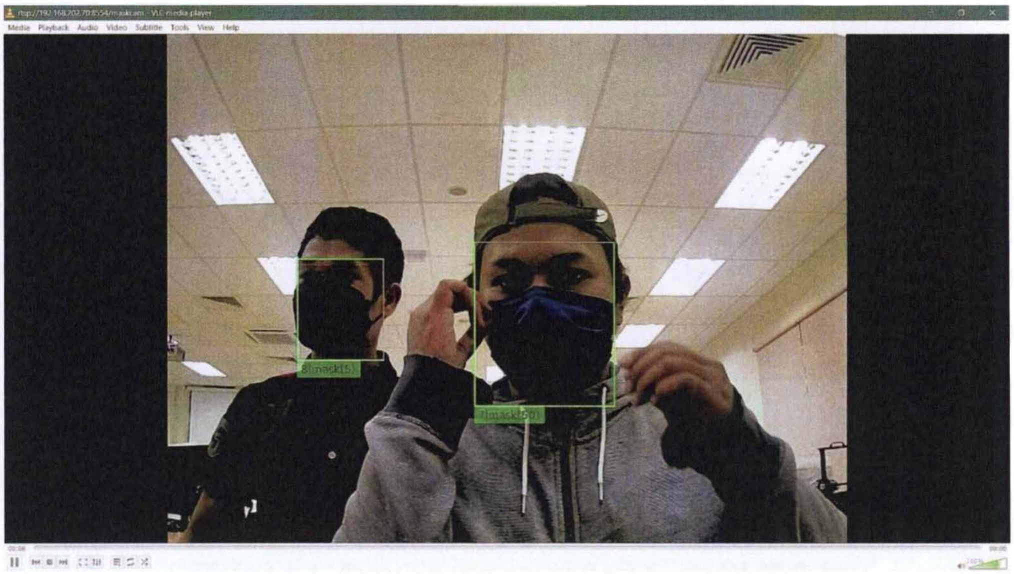


Figure 4.3 Output display on VLC

Figure 4.3 shows the output result from the whole system. The output display can be monitoring by using VLC media player software. VLC is a free and open-source cross-platform multimedia player and framework that plays most multimedia files as well as DVDs, Audio CDs, VCDs, and various streaming protocols. The system had been testing at PAP Café, UMP Pekan as shown in Figure 4.4 and had successfully got the good result based on the face mask detection. The system will indicate either the person wears a mask or not. If the person wears a mask, green box will appear on the face of person and otherwise if the person does not wear a mask the red box will indicate during the streaming process.

Figure 4.4 Running test run at PAP Café, UMP Pekan

4.4 Discussion

This project covers a fair number of technologies from hardware to software, and then combine them to develop crowd monitoring system. A vast body of technical research and testing were done to find out appropriate technologies for these projects. During development, there are several problems arising which help the developer learns not only technical but also problem-solving skills by finding out how to face those challenges. It is obvious that many functions still can be developed further.

4.5 Obstacles

The crowd monitoring system meets most of requirements, but some problems still exist. Therefore, further developments can be considered. There are two obstacles that we found in doing this project.

First, the monitoring on streaming process need the good or high-speed internet connection. On the testing runs, the error could occur anytime if the internet connection is not in good condition. So, we need the high-speed internet connection to run the system.

Lastly, although this merged dataset was substantially larger than our initial dataset, it had two key flaws: it only had two object classifications (mask and no mask), and it lacked consistent labelling criteria because it was a compilation of several datasets collected for unrelated purposes. Some datasets, for example, only included labels for mask-wearing faces, whereas non-masked faces were ignored. Any additional faces were omitted in one of the datasets because there was only one label in the scene's centre.

4.6 Limitation

The limitation of the system is although the precision of mask and no mask objects has increased, the system has a tougher time with not visible objects due to their limited quantity. As a result, the tracker occasionally loses track of people who are not looking at the camera, although the system as a whole works better for those whose faces are plainly visible. Overall, we believe this model is a quantitative and qualitative improvement over our previous model.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

In conclusion, the Jetson Nano had to be a useful enabling technology for edge applications requiring significant processing power. The NVIDIA hardware and software ecosystem accelerates the timeline for developing intelligent devices.

After completing the efforts to containerize the crowd monitoring system application, we have a fully functional prototype of the mask detection camera. The camera can monitor crowds places and detect on mask detections. A live stream of the camera feed, with detections drawn on each frame, can be viewed over a local network. There are a few drawbacks to utilising the Jetson Nano, particularly when migrating apps to other platforms, however NVIDIA's support network helps to mitigate these. While the Jetson Nano's pricing and power consumption may not suit all applications, for those that do, team members wishing to construct edge AI processing systems can explore the Na

5.2 Recommendation

To summarize, to improve this project in future, several recommendations have been identified there is still work that needs to be done for Nvidia Jetson Nano to make it more powerful for building IoT applications. We can improve on object detection model.

Since the goal of this project was to create a smart camera prototype and evaluate the Jetson Nano rather than create the most accurate face mask detector possible, there's room for improvement in components like the object detection model and tracking parameters, which can vary significantly depending on camera setup (distance, angle, and lighting conditions).

For the object detection model, steps that could be taken to improve performance include:

- Provide object visual features to the tracker, to improve matching accuracy.
- Consider detection consistency and the number of frames the tracker needs to interpolate to find a better balance between model accuracy and inference time.
- Crop frame areas to improve object resolution at the object detector input (evaluate doing it automatically using optical flow)
- Curate the dataset more carefully, including all labels for the not visible class at the very least.

REFERENCES

[1] A. Basulto-Lantsova, J. A. Padilla-Medina, F. J. Perez-Pinal and A. I. Barranco-Gutierrez, "Performance comparative of OpenCV Template Matching method on Jetson TX2 and Jetson Nano developer kits," *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, 2020, pp. 0812-0816, doi: 10.1109/CCWC47524.2020.9031166.

[2] A. A. Süzen, B. Duman and B. Şen, "Benchmark Analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN," 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2020, pp. 1-5, doi: 10.1109/HORA49412.2020.9152915.

[3] Raspberry pi, https://static. raspberrypi.org /files/productbriefs/ 200521+Raspberry+Pi+4+Product+Brief.pdf [Online; accessed on 01.05.2021].

[4] "Raspberry Pi 4 Model B Datasheet" Copyright 2019 Raspberry Pi (Trading) Ltd. All rights reserved., 2019 June

[5] W. Vijitkunsawat and P. Chantngarm, "Comparison of Machine Learning Algorithm's on Self-Driving Car Navigation using Nvidia Jetson Nano," 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2020, pp. 201-204, doi: 10.1109/ECTI-CON49241.2020.9158311.

[6] NVIDIA "DATA SHEET NVIDIA Jetson Nano System- On -Module", | Copyright © 2014 – 2020 NVIDIA Corporation. All Rights Reserved

[7] C. Chang and S. Liou, "A Blind Aid System based on Jetson TX2 Embedded System and Deep Learning Technique," 2019 8th International Conference on Innovation, Communication and Engineering (ICICE), 2019, pp. 25-29, doi: 10.1109/ICICE49024.2019.9117526.

[8] NVIDIA. "NVIDIA® JETSON™ TX2 supercomputer on a module for AI at the edge". 2017 NVIDIA Corporation. All rights reserved. NVIDIA, the NVIDIA logo, Jetson, NVIDIA Pascal, CUDA, and Tegra are trademarks and/or registered trademarks of NVIDIA

[9] S. P. Kaarmukilan, S. Poddar and A. Thomas K., "FPGA based Deep Learning Models for Object Detection and Recognition Comparison of Object Detection Comparison of object detection models using FPGA," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 471-474, doi: 10.1109/ICCMC48092.2020.ICCMC-00088.

[10] S. Kanimozhi, G. Gayathri and T. Mala, "Multiple Real-time object identification using Single shot Multi-Box detection," 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), 2019, pp. 1-5, doi: 10.1109/ICCIDS.2019.8862041.

[11] G. Yang et al., "Face Mask Recognition System with YOLOV5 Based on Image Recognition," 2020 IEEE 6th International Conference on Computer and Communications (ICCC), 2020, pp. 1398-1404, doi: 10.1109/ICCC51575.2020.9345042.

[12] F. Zhou, H. Zhao and Z. Nie, "Safety Helmet Detection Based on YOLOv5," 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), 2021, pp. 6-11, doi: 10.1109/ICPECA51329.2021.9362711.

[13] S. P. Kaarmukilan, S. Poddar and A. Thomas K., "FPGA based Deep Learning Models for Object Detection and Recognition Comparison of Object Detection Comparison of object detection models using FPGA," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 471-474, doi: 10.1109/ICCMC48092.2020.ICCMC-00088.

[14] M. Kang, X. Leng, Z. Lin and K. Ji, "A modified faster R-CNN based on CFAR algorithm for SAR ship detection," 2017 International Workshop on Remote Sensing with Intelligent Processing (RSIP), 2017, pp. 1-4, doi: 10.1109/RSIP.2017.7958815.

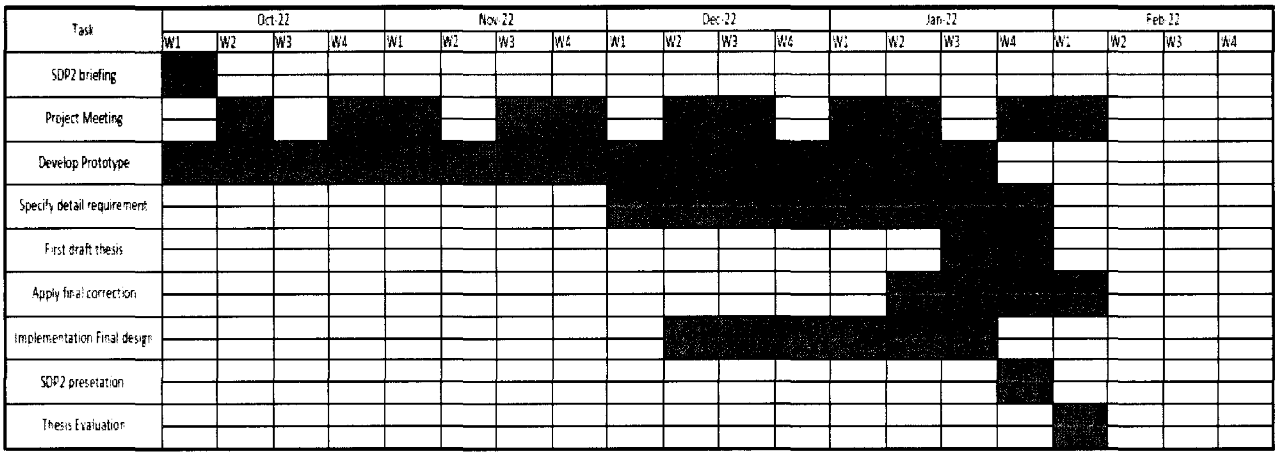
[15] F. A. A. Naqiyuddin, W. Mansor, N. M. Sallehuddin, M. N. S. Mohd Johari, M. A. S. Shazlan and
A. N. Bakar, "Wearable Social Distancing Detection System," 2020 IEEE International RF and Microwave Conference (RFM), 2020, pp. 1-4, doi: 10.1109/RFM50841.2020.9344786.

[16] Bashir, Afnan & Izhar, Umer & Jones, Christian. (2020). IoT based COVID-19 SOP compliance and monitoring system for businesses and public offices. Engineering Proceedings. 2. 10.3390/ecsa-7- 08267.

[17] G. Yang et al., "Face Mask Recognition System with YOLOV5 Based on Image Recognition," 2020 IEEE 6th International Conference on Computer and Communications (ICCC), 2020, pp. 1398-1404, doi: 10.1109/ICCC51575.2020.93450

APPENDICES

Appendix A: Gantt Chart for the Development of the System Setup on Jetson Nano for Smart Crowd Covid Monitoring System.



Appendix B: List Component Price for Project development

No	Component	Quantity	Status	Price (RM)
1.	NVIDIA Jetson Nano Development Kit Rev.B01	1	To buy	550.00
2.	AC8265 Wi-Fi Bluetooth NIC Module for Jetson Nano	1	To buy	96.80
3.	Webcam Fantech Luminous c30	1	To buy	129.00
Total				775.8