TITLE: SMART FACEMASK DETECTOR SYSTEM

STUDENT NAME: **……**

REGISTRATION NUMBER: **…….**

PROGRAMME: **BSc. Information Technology**

A project proposal submitted in partial fulfilment of the requirement for the degree of Bachelor of Science in Information Technology at the School of Computing and Information Technology, Murang'a University of Technology.

YEAR

DECLARATION

I, the undersigned, hereby declare that this thesis proposal is my original work and that it has not been presented to any other university for the award of any academic qualification.

NAME: ……..

SIGNATURE: _____

DATE: _____

APPROVAL

This project proposal has been submitted to the School of Computing and Information Technology with my approval as proposed supervisor.

NAME: Dr Mariga

SIGNATURE: _____

DATE: _____

DEDICATION

I dedicate this work to my mother who has always wanted nothing but the best for me. When I thought of giving up, she encouraged me to push on. The achievements I have held so far is all because you supported me.

ACKNOWLEDGEMENT

This far has been Gods favour; my profound gratitude to my Supervisor, Dr Mariga for his commitment, advice, thoroughness and detailed assessment of my work during the research period. Not only did he create time to make sure I get the best out of this research but also to the one reading this.

My sincere thanks also go to the teaching staff of Murang'a University of Technology for allowing me to carry out this study in their areas of jurisdiction. He has given me strength, wisdom, motivation, good health and peace of mind, patience throughout my semester without which I could not have completed. But because he is faithful God was able to complete it.

He sacrificed his precious time to guide us where possible other times requesting make-up classes so that we can cover more within the limited time.

I am grateful to my fully supportive parents and for their sincere wish to see me succeed.

I would like to express my gratitude to my friends who morally and physically gave me strength throughout the semester.

God bless you all.

ABSTRACT

After the breakout of the overall pandemic COVID-19, there emerges a serious need for security systems, the face mask being the essential one. The COVID-19 pandemic constrained governments across the world to force lockdowns to forestall infection transmissions. Reports show that wearing facemasks while at work diminishes the danger of transmission. We will utilize the dataset to fabricate a COVID-19 face veil identifier with PC vision utilizing Python, OpenCV, and Tensor Flow and Keras. In our proposed framework we will utilize live video transfer lastly in yield it gives an alert sound(buzzer) when somebody not wearing a cover. We will likely recognize whether the individual on picture/video transfer is wearing a face cover or not with the assistance of PC vision and profound learning. The essential point of the task is to recognize the presence of a face cover on human appearances on live real-time video just as on pictures. We have utilized profound figuring out how to foster our face indicator model. The design utilized for the item location intention is Single Shot Detector (SSD) on account of its great presentation precision and high velocity. Close by this, we have utilized fundamental ideas of move learning in neural organizations to finally yield the presence or nonappearance of a face veil in a picture or a video transfer. Trial results show that our model performs well on the test information with 100% and 99% exactness and review, separately To moderate the spread of the COVID-19 pandemic, measures should be taken. We have demonstrated a face veil indicator utilizing SSD engineering and move learning techniques in neural organizations. To prepare, approve and test the model, we utilized the dataset that comprised of 1916 covered faces pictures and 1919 exposed faces pictures. These pictures were taken from different assets like Kaggle and

RMFD datasets. The model was deduced from pictures and live video transfers. To choose a base model, we assessed the measurements like exactness, accuracy and review and chose MobileNetV2 engineering with the best exhibition having 100% exactness and 99% review. It is additionally computationally efficient usingMobileNetV2which make the site simpler to introduce the model to inserted frameworks. This facemask indicator can be conveyed in numerous spaces like shopping centres, air terminals and other substantial traffic spots to screen general society and to stay away from the spread of the infection by checking who is keeping essential guidelines and who isn't...

Table of Contents

CHAPTER ONE: INTRODUCTION

1.0 INTRODUCTION

The year 2020 has shown humankind some stunning series of occasions among which the COVID19 pandemic is the most extraordinary occasion which has alarmed the world since the year started. Influencing the wellbeing and lives of masses, COVID-19 has called for exacting measures to be continued to forestall the spread of illness. From the extremely essential cleanliness guidelines to the medicines in the emergency clinics, individuals are doing everything they can for their own and the general public's wellbeing; facemasks are one of the individual defensive gear. Individuals wear face masks once they get out of their homes and specialists rigorously guarantee that individuals are wearing face veils while they are in gatherings and public spots. To screen that individuals are following this essential security standard, a system ought to be created. A facemask finder framework can be carried out to check this. Facemask discovery intends to recognize if an individual is wearing a veil. The first step to perceive the presence of a veil on the face is to identify the face, which makes the methodology isolated into two sections: to recognize faces and to distinguish covers on those countenances. Face location is one of the utilization of item discovery and can be utilized in numerous spaces like security, biometrics, law implementation and that's just the beginning. There are numerous identifier frameworks created throughout the planet and being executed. In any case, this science needs improvement; a superior, more exact identifier, because the world can't bear the cost of any more expansion in crown cases. In this task, we will be fostering a face veil locator that can recognize faces with covers and

faces without any covers. The pattern of wearing face veils out in the open is ascending because of the COVID-19 Covid pestilence everywhere in the world. Before Covid-19, People used to wear veils to shield their wellbeing from air contamination. While others are reluctant about their looks, they conceal their feelings in general society to shroud their appearances. More than 5,000,000 cases were tainted by COVID-19 in under a half year across 188 nations. The infection spreads through close contact and in swarmed and packed regions. We can handle and foresee new sicknesses with the assistance of new Technologies like man-made consciousness, IoT, Big information, and Machine learning. To all the more likely comprehend disease rates may be decline through our method. Individuals are constrained by laws to wear face covers openly in numerous nations. These standards and laws were created as an activity to the outstanding development in cases and passings in numerous spaces. In any case, the way toward checking huge gatherings of individuals is getting more troublesome in open regions. So we will make a robotization interaction for distinguishing the countenances. Here we present a facemask location model that depends on PC vision and profound learning. The proposed model can be incorporated with Surveillance Cameras to block the COVID-19 transmission by permitting the recognition of individuals who are wearing veils not wearing face covers. The model is reconciliation between profound learning and traditional AI procedures with Open cv, Tensor stream and Keras. We will accomplish the most elevated exactness and burn-through minimal time during the time spent preparing and discovering. In this report, we have proposed a locator which utilizes SSD for face recognition and a neural

organization to distinguish the presence of a face cover. The execution of the calculation is on pictures, recordings and live video streaming.

1.1 BACKGROUND OF STUDY

Before Covid-19, masks were worn by individuals to shield their wellbeing from air contamination. Researchers have presumed that wearing face veils deals with diminishing COVID-19 transmission. In 2020, the quick spread of COVID-19 drove the World Health Organization to announce COVID-19 as a worldwide pandemic. Another strain of infection was distinguished in people, known as novel Covid, which was never recently been recognized in people. Covers are a wide gathering of infections that cause ailments that reach from fundamental colds to contaminations like Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS). The previously contaminated patient of Covid was found in December 2019. The propensity for wearing face covers while venturing out is ascending because of the COVID-19 Covid scourge. The infection spreads through close contact with people and in swarmed/packed spots. Among them cleaning hands, keeping a protected separation, wearing a cover, abstaining from contacting eyes, nose, and mouth are fundamental, where wearing a veil is the most straightforward one. Shockingly, individuals are not keeping these standards appropriately which is bringing about speeding the spread of this infection. The arrangement can be to distinguish individuals not wearing the veil and educating their specialists. the face veil identification is a procedure to see if the individual is wearing a cover or not. In clinical applications, Deep learning procedures are profoundly utilized as it permits analysts to examine and assess enormous amounts of information. Profound

learning models have shown an incredible part in object identification. These models and structures can be utilized in identifying the veil on a face. Here we present a face cover recognition model which depends on PC vision and profound learning. The proposed model can be incorporated with PC or PC cameras permitting it to distinguish individuals who are wearing veils and not wearing covers. The model has been assembled utilizing profound learning and traditional AI strategies with OpenCV, tensor stream and Keras. We have presented a correlation between three AI calculations to track down the most appropriate calculation that yields the most noteworthy exactness. The spread of COVID-19 infection has decreased however it is as yet not finished. If everybody follows all the security measures, it can conclude. This will help in bringing the cases down to such a level that COVID19 infection can disappear from all over.

1.2 PROBLEM STATEMENT

Over the last 15 months, the need for a face mask detector has risen rapidly. This means more programmers are spending more of their time trying to come up with innovative means of curbing the spread of the deadly virus Covid-19

Implementing the system has become difficult because of issues which include:

1. Lack of enough datasets – are the fuel to python detection systems. Storing and maintaining all datasets is very costly and collecting the datasets is also hard to do.
2. The field of facemask detectors is a new area brought about by the coronavirus pandemic.

3. The database of the current system is not sufficient enough to store all the data and information of the dataset. This leads to the system becoming slow if too many datasets are used.

1.3 OBJECTIVES OF THE STUDY

1.3.1GENERAL OBJECTIVE

To develop a Smart face detector System that will remain its users to wear facemasks before accessing the system.

1.3.2 SPECIFIC OBJECTIVES

i. To analyse the existing face mask detector System to establish system requirements.
ii. To Design a face mask detector System with a simple user-friendly interface.
iii. To develop and implement the designed system
iv. To validate the facemask detecting system.

1.4 RESEARCH QUESTIONS

i. Which features of the current system pose a challenge in the design of the user interface and its implementation?
ii. Which feature of the current system poses a challenge in detecting faces and whether there are masks or no masks?
iii. Which feature poses a security threat in the current system?

iv. What are some of the methods which can be used to validate the newly developed system?

1.5 HYPOTHESES

i. The identification of the features of face mask detector systems that ensure a good design interface and user-friendly interface and their implementation will curb the issue of the poor user interface.

ii. The recognition of the capabilities of python coded system that ensure fast detecting abilities using OpenCV libraries

iii. The identification of the feature of the face mask detector system and its implementation that help mitigate the spread of the deadly virus.

iv. Effective validation of the system before adoption will provide an efficient system to curtail the spread of the virus and aim at the protection of its users

1.6 JUSTIFICATION

The system will bring about the best and the most beautiful user-friendly interface an end-user may need and wish to have. The system will make sure communication is achieved in the system where the face mask detector can detect a face without a mask and alert the person to wear one with immediate effect. It will also mitigate the spread of the Covid-19 virus and thus provides good health to its users.

CHAPTER TWO: LITERATURE REVIEW

2.1 INTRODUCTION

object detention is one of the moving trends in the field of image preparation and PC vision. Rangingfromsmallscalepersonalapplicationstolargescaleindustrialapplications, object detection and acknowledgement is utilized in a wide scope of businesses. A few models incorporate picture recovery, security and knowledge, OCR, clinical imaging and horticultural observing. In object discovery, a picture is perused and at least one article in that picture are sorted. The area of those items is likewise specified by a limit called the jumping box.

Customarily, scientists utilized example acknowledgement to foresee faces dependent on earlier face models. An advancement face discovery innovation then, at that point was created named as Viola-Jones finder that was an improved strategy of utilizing Haar

[1], the computerized image highlights utilized in object recognition. Be that as it may, it fizzled because it didn't perform well on faces in dim regions and non-front facing faces. From that point forward, scientists are anxious to foster new calculations dependent on profound figuring out how to work on the models. Profound learning permits us to learn highlights from start to finish way and eliminating the need to use earlier information for shaping element extractors. There are different techniques for object location dependent on profound realizing which are isolated into two classes: one phase and two-phase object indicators.

In two stages, detectors use two neural networks to detect objects, for instance, region-based convolutional neural networks (R-CNN) and Faster R-CNN. The first neural network is used to generate region proposals and the second one refines these region proposals; performing a

coarse-to-fine detection. This strategy results in high detection performance compromising on speed. The seminal work R-CNN is proposed by R. Girshick et al.

[2]. R-CNN uses selective search to propose some candidate regions which may contain objects. After that, the proposals are fed into a CNN model to extract features, and a support vector machine (SVM) is used to recognize classes of objects. However, the second stage of R-CNN is computationally expensive since the network has to detect proposals in a one-by-one manner and uses a separate SVM for final classification. Fast R-CNN

[3] solves this problem by introducing a region of interest (ROI) pooling layer to input all proposal regions at once. Faster RCNN

[4] is the evolution of R-CNN and Fast R-CNN, and as the name implies its training and testing speed is greater than those of its predecessors.

While R-CNN and Fast R-CNN use selective search algorithms limiting the detection speed, Faster R-CNN learns the proposed object regions itself using a region proposal network (RPN). On the other hand, a one-stage detector utilizes only a single neural network for region proposals and detection; some primary ones being SSD (Single Shot Detection) [5] and YOLO (You Only Look Once)[6]. To achieve this, the bounding boxes should be predefined. YOLO separates the picture into a few cells and afterwards coordinates with the bounding boxes to objects for every cell. This, in any case, isn't useful for little estimated objects. Hence, multi-scale recognition is presented in SSD which can identify objects of fluctuating sizes in a picture. Afterwards, to further develop identification precision, Lin et. al [7] proposes Retina Network (RetinaNet) by joining an SSD and FPN (include pyramid organization) to expand location exactness and decrease class lopsidedness. One-stage indicators have higher speed however compromises the identification execution yet then, at

that point just are liked more than two-stage finders. Like item location, face discovery embraces similar structures as one-stage and two-stage indicators, however, to further develop face recognition precision, more face-like highlights are being added. Notwithstanding, there is incidental examination zeroing in on face veil location. Some all-around existing face cover locators have been demonstrated utilizing OpenCV, Pytorch Lightning, MobileNet, RetinaNet and Support Vector Machines. Here, we will examine two tasks. One undertaking utilized Real World

Masked Face Dataset (RMFD)whichcontains5,000maskedfacesof525peopleand90,000normal faces [8]. These pictures are 250 x 250 in measurements and cover all races and identities and are unequal. This task accepting 100 x 100 pictures as info, and along these lines, changed each example picture while questioning it, by resizing it to 100x100. In addition, this task utilizes PyTorch then they convert pictures to Tensors, which is the base information type that PyTorch can work with. RMFD is imbalanced (5,000 covered faces versus 90,000 non-concealed appearances). In this way, the proportion of the examples in train/approval while parting the dataset was kept equivalent utilizing the train test split capacity of SK-learn. Additionally, to manage uneven information, they passed this data to the misfortune capacity to keep away from proportioned step sizes of the enhancer. They did this by doling out a load to each class, as indicated by its address capacity in the dataset. They allowed more weight to classes with few examples so the organization will be punished more if it commits errors foreseeing the mark of the set classes. While classes with huge quantities of tests, they doled out to the smaller weight. This makes their organization preparing freethinker to the extent of classes.

To stack the information efficiently this venture utilized the information loader. For example, in this venture, they utilized the PyTorch lighting, and to stack them for preparing and approval they separated information into 32 bunches and appointed crafted by stacking to the

4 number of labourers, and this methodology permitted them to perform multi-measure information stacking. Like the greater part of the tasks, this undertaking additionally utilized an Adam analyzer. On the off chance that any Model has a high pace of learning, it learns quicker, however, it skips a great deal to come to the worldwide minima and may wander from the worldwide minima. In any case, a little learning rate may set aside considerably lower effort to prepare, however it compasses to the worldwide minima. Assuming the deficiency of the model decays rapidly for any learning rate, that learning rate would be the best learning rate. Nonetheless, it appears to be that this undertaking considered the 0.00001 learning rate would be the awesome their model so it could work efficiently. To prepare the model they defined a model checkpointing get back to where they needed to save the best exactness and the least misfortune. They attempted to prepare the model for 10 ages and in the wake of finding ideal age, they saved the model for 8 ages to test on the genuine information. To dispose of the issue of impediments of the face which raises a ruckus face identifiers to distinguish veils in the pictures, they utilized an underlying OpenCV profound learning face recognition model. For example, the Haar-Cascade model could be utilized yet the issue of the Haar-Cascade model is that the location outline is a square shape, not a square. That is the reason, without catching the part of the foundation, the face outline jars fit the sum of the face, which can meddle with the face cover model forecasts. In the second undertaking [9], a dataset was made by Prajna Bhandary utilizing a PyImage Search peruser. This dataset comprises 1,376 pictures having a place with all races and is adjusted. There are 690 pictures with veils and 686 without covers. Initially, it took typical pictures of countenances and afterwards made a modified PC vision Python content to add face covers to them. In this manner, it made a genuine world appropriate artificial dataset. This technique utilized the facial tourist spots which permit them to recognize the various pieces of the countenances like eyes, eyebrows, nose, mouth, facial structure and so forth To utilize the

facial tourist spots, it snaps a photo of a not wearing an individual cover, and, then, at that point, it recognizes the bit of that individual's face. In the wake of knowing the area of the face in the picture, it separated the face Region of Interest (ROI). After limiting facial tourist spots, an image of a veil is put into the face. In this task, implanted gadgets are utilized for an organization that could diminish the expense of assembling...

MobileNetV2 design is utilized as it is an exceptionally efficient engineering to apply on implanted gadgets with a restricted computational limit like Google Coral, NVIDIA Jetson Nano. This undertaking performed well, be that as it may if a huge bit of the face is blocked by the veil, this model couldn't identify if an individual is wearing a cover. The dataset used to prepare the face locator didn't have pictures of individuals wearing face veils accordingly, if the enormous segment of countenances is blocked, the face finder would likely neglect to recognize appropriately. To dispose of this issue, they should assemble genuine pictures of individuals wearing veils instead of artificially created pictures.

2.2 EXISTING SYSTEMS

Sujatha and Chatterjee [1] proposed a model that could be useful to foresee the spread of COVID-19 by using linear regression, Multilayer perceptron and Vector autoregression model on the COVID-19 Kaggle data to envision the epidemiological example of the malady and pace of COVID-2019 cases in India. Navares et al. [2] introduced an answer for the issue of anticipating everyday medical clinic confirmations in Madrid because circulatory and respiratory cases dependent on biometeorological markers. Cui and Singh created and applied the MRE hypothesis for the month to month streamflow prediction with spectral power as a random variable. A system [4] that restrict the growth of COVID-19 by finding out people who are not wearing any facial mask in a smart city network where all the public places are monitored with ClosedCircuit Television (CCTV) cameras. Firstly, CCTV cameras are used

to capture real-time video footage of different public places in the city. From that video footage, facial images are extracted and these images are used to identify the mask on the face. Another model [5] for face detection using semantic segmentation in an image by classifying each pixel as face and non-face i.e. effectively creating a binary classifier and then detecting that segmented area. It works very well not only for images having frontal faces but also for non-frontal faces. The most helpful project for us, proposed [6] a method for automatic door access system using face recognition technique by using python programming and from OpenCV library Haar cascade method. Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones [7]. Another research [8] in which a hybrid model using deep and classical machine learning for face mask detection is presented. A face mask detection dataset consists of with mask and without mask images, then using OpenCV to do real-time face detection from a live stream via webcam. Another tutorial [9], had a two-phase COVID-19 face mask detector, detailing how a computer vision/deep learning pipeline will be implemented. The trained COVID-19 face mask detector will implement two more additional Python scripts used to detect COVID-19 face masks in images and detect face masks in real-time video streams.

2.3 PROS AND CONS OF EXISTING SYSTEMS

PROS

- Manual Monitoring is extremely hard for officials to check if the people groups are wearing a mask. So in our method, We are utilizing a webcam to recognize individuals' countenances and to keep from infection transmission.

- It has fast and high accuracy

- This system can be implemented in ATMs, Banks etc

- We can keep people safe from our technique.

- It provides a buzzer sound to wear a mask.

CONS

There were not many challenges faced but the two problems that were time-consuming and made the tasks tedious are discussed as follows.

- One was the unnecessary information stacking time in Google Colab Notebook while stacking the dataset into it. Since the runtime restarting invigorates every one of the cells, the cell for dataset stacking took more often than not while running.
- Secondly, the entrance issue in Google Colab Notebook: it didn't permit the entrance of webcam which represented an obstacle in testing pictures and live video transfer through Google Colab Notebook. In this way, we needed to run the code locally on the PC through which we tried the code on the live video transfer.

2.4 GAPS OF THE EXISTING SYSTEM

This area tries to show the evolvement of the facemask detecting systems since the first system was created. From one transition to another, there is a clear gap that is implemented in the proceeding step.

- Xun et al. (2020) designed a comprehensive UML model of the facemask detecting system. The system was designed for a local data analytics company wishing to experiment with and data analytic company. The reporting capabilities modules built into the system were poor and too complex for the user to understand.

2.4.1 User Interface and End-User

The end-user of an item connects with, or controls, an item, programming or equipment gadget. UI is intended to permit people to control machines adequately and proficiently, and great UI ought to According to Fred Churchville (2019), User interface (UI) is any technique or means by which make the communication between the client and the item as smooth, easy and charming as could be expected. The current framework has represented a hole in this space whereby the UI isn't cordial and along these lines, the end client thinks that it's hard to utilize and consequently tedious interaction before one will be recognized by the framework. The proposed framework will execute this where the cycle of location will be basic and direct. For the instance of an interface, it will be intuitive with the most intriguing and natural images and tones, consequently making the undertaking smooth, easy and as agreeable as could be expected. This will empower the UI to perform proficiently and as needed by the client subsequently fulfilling end clients' necessities and assumptions adequately. In the proposed framework the accompanying interface components will be considered to assist with accomplishing the objective of a delightful UI.

2.4.2 Authentication and Fraud

Fraud is an intentionally deceptive action designed to provide the perpetrator with an unlawful gain or to deny a right to a victim. Types of fraud include tax fraud, credit card fraud, wire fraud, securities fraud, and bankruptcy fraud. Fraudulent activity can be carried out by one individual, multiple individuals or a business firm as a whole. (James, March 2021)

According to James (2021), in an article reviewed in March; there are key takeaways to this. They include:

- Fraud involves deceit to illegally or unethically gain at the expense of another.
- In finance, fraud can take on many forms including making false insurance claims, cooking the books, pump & dump schemes, and identity theft leading to unauthorized purchases.
- Fraud costs the economy billions of dollars every year, and those who are caught are subject to fines and jail time.

In the proposed facemask detector system will authenticate the user before login into the system with a valid user email address and a one-time password which will be stored in the system database.

2.4.3 System Validation Techniques

According to Full Software Validation and verification – Definition and Methods. (September 13, 2017).

Software validation is the cycle that watches that the product item fulfils or fits the planned use. There are two different ways to perform programming approval: inner and outside. For the proposed framework, the two strategies would be applied to altogether test it. Inner approval would be led to guarantee that the prerequisite particulars are met and outside approval to ensure it addresses the issues of every user.

2.4.4 AUTHENTICATION SERVER

Authentication Server: is the piece of the framework that handles authentication and authorization. It's anything but a substance utilizing the situation that has the privilege to play out the expected activity like refreshing, recovering, moving, and so forth, on the house data gave. The verification worker awards admittance to approved clients and denies unapproved clients' admittance to records or assets on the framework. This is accomplished through the age of usernames and passwords for inhabitants, specialists, landowners and managers of the application. Any individual from the administrator is required to sign in to the framework with their username and secret word. The framework contrasts the username and secret word and those in the neighbourhood information base and awards admittance to the client if they match, in any case, the client is denied access.

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

This chapter examines the System improvement philosophy for the proposed system. There are various techniques to foster a framework and some incorporate; Waterfall Model/Spiral Model/Scrum Model/Rapid Application Development Model, Spiral Model. Have dissected the various models and the Waterfall model will be utilized in framework improvement. This is a direct result of its numerous benefits over different models and generally given its effortlessness and simplicity to utilize and comprehend. In this part, there is two significant sections and keys to any framework improvement and that is System Analysis and System Design. Each is talked about with its captions and how they will be accomplished in the proposed framework.

3.2 SYSTEM ANALYSIS

System Analysis is an interaction of gathering and deciphering realities, distinguishing the issues, and deterioration of a framework into its segments. Realities were gathered utilizing various methodologies and from various sources and afterwards investigated for use.

3.2.1 Data Collection Methods

Introduction

For this research, first of all, secondary sources of information were used following the inductive methodology. Serious perusing must be directed to discover which of these sources had information applicable to the facemask distinguishing framework. A few magazines, books and websites were utilized to discover data about-face veil recognizing system.

For the secondary sources, the information was gathered through an exhaustive perusing of the material. Notwithstanding, the essential information assortment strategy was through messages, as a large portion of the correspondence which occurred through polls was done that way. The meetings were directed through gatherings with the individuals who were accessible which kept going around 20 minutes. Be that as it may, for the inaccessible individuals, they were led through video conferencing which were recorded and later translated. After the secondary sources were exhausted, the deductive approach began where a combination of two classic research tools – questionnaires and interviews were used. These questionnaires and interviews were given to the sample of the population selected.

Primary Data Collection

a) Questionnaires.

These were picked because they are a solid and speedy approach to gather information from various sources in a productive and ideal way. This technique gives more data from different people and offers more prominent adaptability as the chance to rebuild questions. This procedure is favoured because it's anything but a nearer contact between the clients and the designer consequently dispersing the likelihood of the finished framework being dismissed by use.

Some questions were prepared to guide the research toward the satisfaction of the research objectives.

These questions include:

Question 1: Do you think the current facemask detector system is efficient enough to mitigate the spread of Covid-19?

Question 2: Is the smart facemask detector system efficient enough to be installed in ATMs, public vehicles and public places?

Question 3: Have you ever had any software difficulties like the face mask detector delaying to identify your face?

Question 4: Do you think it is time the system is replaced with another faster and efficient system?

b) Interviews.

These were utilized as a reciprocal exploration technique, as they take into consideration a more inside and out, open conversation, and are more casual, the free association between the questioner and the interviewee (Potter, 2002; Sarantakos. 2013).

Secondary Data Collection

This information I will gather from existing sources e.g., books, web, diaries and magazines that were gathered by different specialists and examination was finished. It is from that information that I will then, at that point contrast and the essential information and settle on a choice and end.

Data Analysis Methods.

Investigation of the polls was done through topical examination. This was to recognize the normal answers and thoughts that surfaced over and again. For the meetings, the content examination was applied to recognize the examples in the recorded discussions.

3.2.2 System Requirements

3.2.2.1 User Requirements

It entailed user involvement and statements of facts and assumptions that define the expectations of the system in terms of mission objectives, environment, constraints and measures of effectiveness and suitability. The users:

i) A system that improves the efficiency of information storage and retrieval.

ii) A system that is easy to learn and use

iii) A system that is fast in processing transactions

iv) A system that is flexible, safe and convenient

3.2.2.2 Functional Requirements

This is a necessary task, action or activity that was accomplished. The proposed system will be able to:

i) Allow users, to practice basic actions like wearing a face mask.

ii) Remains the user to stay safe in the public area

iii) Allow the user to keep the mask on at all times

iv) Allow the user to add more pictures in the database to act as a dataset

3.2.2.3 Hardware Requirements

Operating System: 64-Bit operating system, x86/x64-based processor

Processor: Intel Core i5 or higher

Installed Memory: 8.00GB or higher

Speed: 2.80GHz or faster

3.2.2.4 Software Requirements

- Operating System – Windows 10 or higher
- Datasets: have two folders are used(with facemask and without facemask)
- Python language: Opencv, Numpy, Matplotlib, Tensorflow
- IDE & Tools: Anaconda IDE

3.3 SYSTEM DESIGN

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. (VanGigch, J.P.2013).

In the wake of amassing every one of the prerequisites for the framework, the genuine framework will be created utilizing the accessible instruments and systems recognized in the past section.

Design considerations

In this design, coming up next are consolidated into the facemask location framework. The significant prerequisite for executing this venture utilizing python programming language alongside Deep learning. AI, Computer vision and python libraries. The engineering comprises of Mobile Net as the spine, it very well may be utilized for high and low calculation situations. We are utilizing CNN Algorithm in our proposed framework.

Implementation:

We have four modules

1. Datasets Collecting: We collect no data sets with face masks and without masks. we can get high accuracy depends on collecting the number of images.
2. Datasets Extracting: We can extract the features using mobile net v2 of mask and no mask sets
3. Models Training: We will train the model using open cv, Keras (python library).

4. Facemask Detection: We can detect Preprocessing images and also detect via live video. If people wear a mask, it will permit them, if not then it will give the buzzer to wear a mask to prevent them from virus transmission.

3.3.1 DESIGN PHASES

3.3.1.1 Conceptual Design

The conceptual design was the absolute first period of plan wherein drawings or strong models were the predominant apparatuses and items. The applied plan stage gave a portrayal of the proposed framework as far as a bunch of incorporated thoughts and ideas about what it was to do, act and resemble, that was justifiable by the clients in the way planned.

3.3.1.2 DATABASE DESIGN

The proposed system will utilize Datasets as a data set and the data set holds its significant information with their details. Data set will be planned by the clients of the framework where distinctive datasets (with facemask) and (without facemask) with information appropriately will be made. The data set effectively stores information utilized in the framework. Information can be recovered from the data set when required. The dataset will contain two arrangements of information, one organizer with a facemask and the other envelope marked without a facemask. When your register into the framework, your information will be caught in the data set which will be utilized in future. Then, at that point, the framework will permit the client to utilize the facemask locator. This will empower the verification of the worker basics.*3.3.1.3 PHYSICAL DESIGN*

This step will involve the system development methodology, the steps in the method and the description of each step. In the proposed system the Waterfall model of system development will be used.

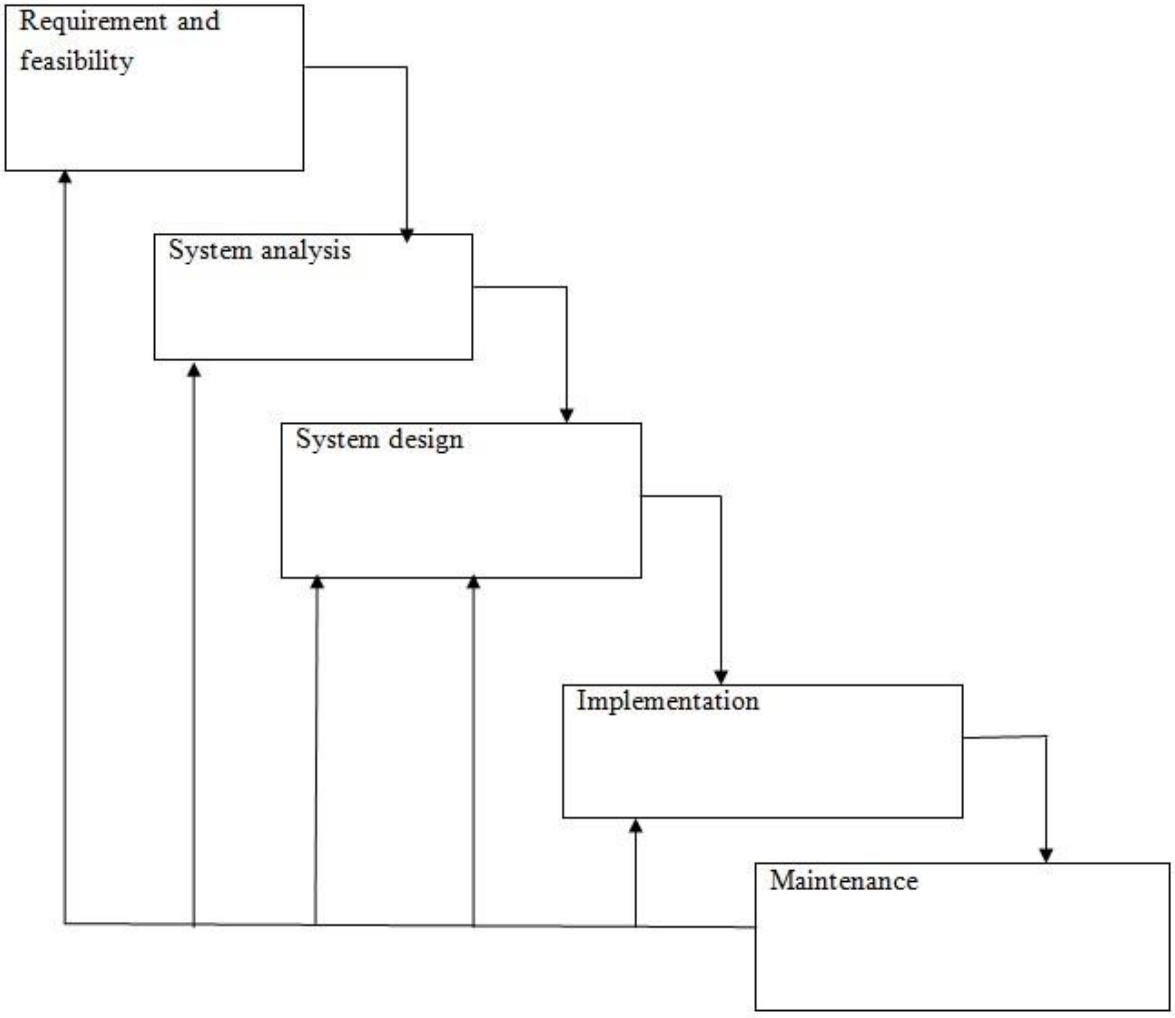
3.3.2 System Development Methodology

Waterfall Model

It is involved the stages that the designer will utilize when fostering the framework. It is a consecutive model consequently, the name cascade. The designer needs to get done with one phase before going to the following one. It contains the plausibility study, investigation stage, plan stage, coding stage, testing stage, execution stage lastly the support stage. It is a straightforward model and simple to utilize and comprehend. With cascade advancement

based approaches, the examiners and clients continue successively starting with one stage then onto the next. The expectations from each stage are voluminous and are introduced to the undertaking backer for endorsement as the task moves from one stage to another. When the stage is supported by the support it closes and the following stage starts.

The following is a waterfall diagram showing the different activities in each stage.



The following are the explained stages of the waterfall model and how they will be achieved in the proposed system.

3.3.2.1 Feasibility Study

Here, I will complete an investigation to acquire a comprehension of the face cover indicator current framework and issues experienced in this framework through meetings, perceptions, and interests. I will utilize the acquired information to decide the practicality of the framework being proposed as far as specialized, monetary and social attainability.

3.3.2.2 Requirements and Analysis

At this stage, I will accumulate data about what the client needs and characterize the issues the framework is relied upon to settle. I will likewise incorporate clients recognition settings, dataset and python (code) capacities and their similarity. I will assemble necessity, for example, programming like the programming language to utilize, information base model and equipment required like PC.

3.3.2.3 Design

At this stage, I will make a general plan of the framework engineering and actual plan which incorporates User Interface and Database plan. It is at this stage that I will recognize any issues before moving onto the following stage. The yield of this stage is the plan determination which is utilized in the following phase of execution.

3.3.2.4 Coding/Implementation

At this stage, I will start coding according to the plan specification(s). The yield of this progression is at least one item parts worked by a pre-characterized coding standard and repaired, tried and incorporated to fulfil the framework engineering necessity.

3.3.2.5 Testing

At this stage, I will guarantee both individual and coordinated entire are systematically checked to guarantee they are sans mistake and fulfil buyer prerequisite. I will include both unit testing of the individual code module, framework testing of the incorporated item and acknowledgement testing directed by or for a customer. I will guarantee bugs found are amended before moving to the following stage. I will likewise get ready, survey and distribute item documentation at this stage.

3.3.2.6 Installation

It is done once the product has been tested and certified as fit for use. The system is prepared for use at the customer site.

3.3.2.7 Maintenance

This stage occurs after installation. It includes changes on the framework to further develop execution. Such changes are client started or because of bug being found which were at first not know. These alterations are recorded for documentation and framework updates.

APPENDICES

Appendix 1: Project Schedule/workplan

This is the planned schedule for the development of the system.

Activities	June 2021	July 2021	August 2021	September 2021
Req. Analysis				
Design				
Coding				
Testing /Implementation				

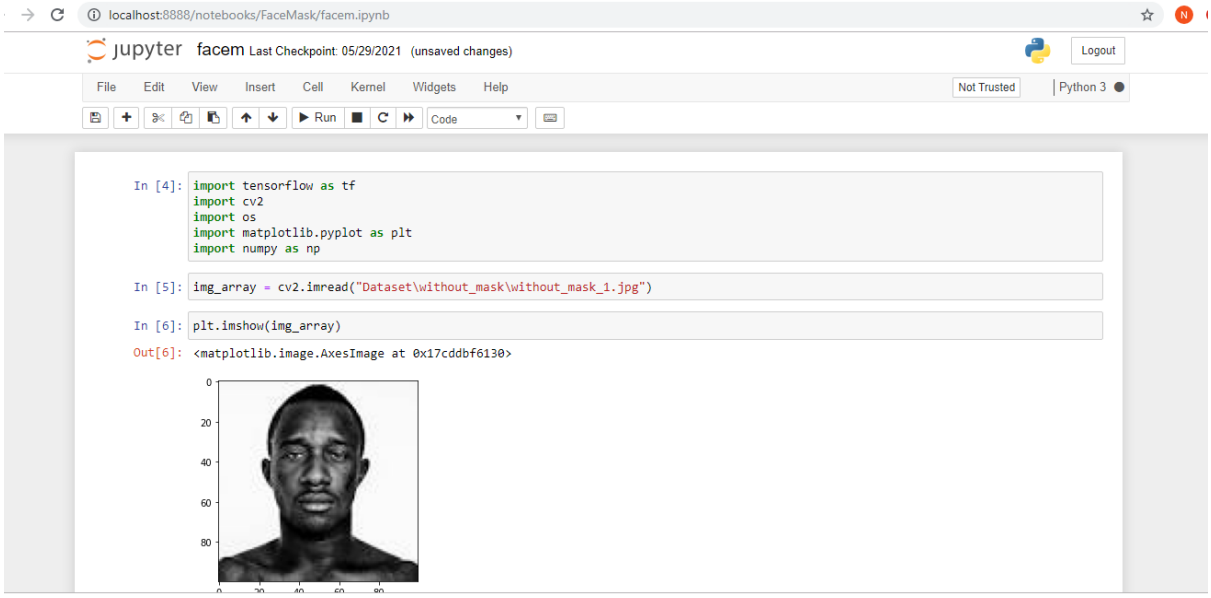
Appendix 2: Budget

This is the estimated amount of money that will be used throughout the development process of the proposed system.

ITEM	QUANTITY	UNIT PRICE(Ksh)	TOTALS (Ksh)
Laptop	1	60,000	60,000
Software	1	4,000	4,000
Internet and Airtime	-	3,000	3,000
Printing and Binding	-	3,000	3,000
Local transport	5	3,000	15,000
subsistence	6	1,000	6,000

Appendix 3: The Output

The following are some of the sample's screenshots of the proposed system.



Out[35]: <matplotlib.image.AxesImage at 0x17c81f574c0>

In [36]:
```
final_image =cv2.resize(frame, (224,224))
final_image = np.expand_dims(final_image, axis = 0)
final_image = final_image/255.0
```

In [37]:
```
predictions = new_model.predict(final_image)
```

In [38]:
```
predictions
```

Out[38]: array([[2.4603166e-14]], dtype=float32)

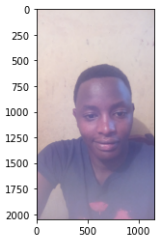
In [39]:
```
frame = cv2.imread('IMG_20210504_174107.jpg')
```

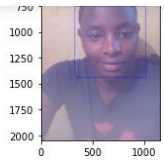
In [40]:
```
frame.shape
```

Out[40]: (2048, 1152, 3)

In [41]:
```
plt.imshow(cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))
```

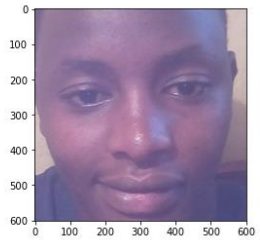
Out[41]: <matplotlib.image.AxesImage at 0x17c829097c0>





In [47]:
```
plt.imshow(cv2.cvtColor(face_roi, cv2.COLOR_BGR2RGB))
```

Out[47]: <matplotlib.image.AxesImage at 0x17c829fe9a0>



In [48]:
```
final_image = cv2.resize(face_roi, (224,224))
```

APPENDIX 4: Code Output

The following are samples of the code which have already coded using python code with anaconda.

```
In [13]: import random

         random.shuffle(training_Data)

In [14]: x = []
         y = []

         for features,label in training_Data:
             x.append(features)
             y.append(label)


         x = np.array(x).reshape(-1, img_size, img_size, 3)

In [15]: x.shape
Out[15]: (144, 224, 224, 3)

In [16]:

         x= x/255.0;

In [17]: y[60]
Out[17]: 1
```

```
         #reading pics and turning the into arrays

In [10]: training_Data = []

         def create_training_Data():
             for category in Classes:
                 path = os.path.join(Datadirectory, category)
                 class_num = Classes.index(category)
                 for img in os.listdir(path):
                     try:
                         img_array = cv2.imread(os.path.join(path, img))
                         new_array = cv2 .resize(img_array, (img_size, img_size))
                         training_Data.append([new_array,class_num])
                     except Exception as e:
                         pass

In [11]: create_training_Data()

In [12]: print(len(training_Data))
         144

In [13]: import random

         random.shuffle(training_Data)
```

```
In [18]: Y= np.array(y)

In [19]: import pickle

         pickle_out = open("X.pickle", "wb")
         pickle.dump(x, pickle_out)
         pickle_out.close()

         pickle_out = open("Y.pickle", "wb")
         pickle.dump(y, pickle_out)
         pickle_out.close()

In [20]: pickle_in = open("X.pickle","rb")
         X = pickle.load(pickle_in)

         pickle_in= open("Y.pickle", "rb")
         y = pickle.load(pickle_in)

In [21]: #Deep learning model for training transfer learning

In [22]: import tensorflow as tf
         from tensorflow import keras
         from tensorflow.keras import layers

In [23]: model = tf.keras.applications.mobilenet.MobileNet()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/mobilenet/mobilenet_1_0_224_tf.h5
17227776/17225924 [==============================] - 5s 0us/step
```

In [24]:
```
model.summary()
```

```
conv_dw_6_relu (ReLU)        (None, 14, 14, 256)       0
_________________________________________________________________
conv_pw_6 (Conv2D)           (None, 14, 14, 512)       131072
_________________________________________________________________
conv_pw_6_bn (BatchNormaliza (None, 14, 14, 512)       2048
_________________________________________________________________
conv_pw_6_relu (ReLU)        (None, 14, 14, 512)       0
_________________________________________________________________
conv_dw_7 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
_________________________________________________________________
conv_dw_7_bn (BatchNormaliza (None, 14, 14, 512)       2048
_________________________________________________________________
conv_dw_7_relu (ReLU)        (None, 14, 14, 512)       0
_________________________________________________________________
conv_pw_7 (Conv2D)           (None, 14, 14, 512)       262144
_________________________________________________________________
conv_pw_7_bn (BatchNormaliza (None, 14, 14, 512)       2048
_________________________________________________________________
conv_pw_7_relu (ReLU)        (None, 14, 14, 512)       0
```

#transfer learning

In [25]:
```
base_input = model.layers[0].input
```

In [28]:
```
new_model = keras.Model(inputs = base_input, outputs = final_output)
```

In [29]:
```
new_model.summary()
```

```
conv_pw_2_relu (ReLU)        (None, 56, 56, 128)       0
_________________________________________________________________
conv_dw_3 (DepthwiseConv2D)  (None, 56, 56, 128)       1152
_________________________________________________________________
conv_dw_3_bn (BatchNormaliza (None, 56, 56, 128)       512
_________________________________________________________________
conv_dw_3_relu (ReLU)        (None, 56, 56, 128)       0
_________________________________________________________________
conv_pw_3 (Conv2D)           (None, 56, 56, 128)       16384
_________________________________________________________________
conv_pw_3_bn (BatchNormaliza (None, 56, 56, 128)       512
_________________________________________________________________
conv_pw_3_relu (ReLU)        (None, 56, 56, 128)       0
_________________________________________________________________
conv_pad_4 (ZeroPadding2D)   (None, 57, 57, 128)       0
_________________________________________________________________
conv_dw_4 (DepthwiseConv2D)  (None, 28, 28, 128)       1152
_________________________________________________________________
conv_dw_4_bn (BatchNormaliza (None, 28, 28, 128)       512
```

In [30]:
```
new_model.compile(loss="binary_crossentropy", optimizer = "adam", metrics = ["accuracy"] )
```

In [31]:
```
new_model.fit(X,Y, epochs = 1,validation_split = 0.1)
```

In [ ]:
```
```

In [42]:
```
faceCascade = cv2.CascadeClassifier(  'D:/FaceMask/haarcascade_frontalface_default.xml')
```

In [43]:
```
gray =cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

In [44]:
```
gray.shape
```

Out[44]:
```
(2048, 1152)
```

In [45]:
```
faces = faceCascade.detectMultiScale(gray,1.1,4)
for x,y,w,h in faces:
    roi_gray = gray[y:y+h, x:x+w]
    roi_color = frame[y:y+h, x:x+w]
    cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0, 0), 2)
    facess = faceCascade.detectMultiScale(roi_gray)
    if len(facess) == 0:
        print("Face not Detected")
    else:
        for (ex,ey,ew,eh) in facess:
            face_roi = roi_color[ey: ey+eh, ex:ex + ew]
```

In [46]:
```
plt.imshow(cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))
```

Out[46]:
```
<matplotlib.image.AxesImage at 0x17c829a5e50>
```

In [48]:
```
final_image = cv2.resize(face_roi, (224,224))
final_image = np.expand_dims(final_image, axis = 0)
final_image = final_image/255.0
```

In [49]:
```
predictions = new_model.predict(final_image)
```

In [50]:
```
predictions
```

Out[50]:
```
array([[7.219057e-08]], dtype=float32)
```

Real time video

In [ ]:
```
import cv2
path = "D:/FaceMask/haarcascade_frontalface_default.xml"
font_scale = 1.5
font = cv2.FONT_HERSHEY_PLAIN

rectangle_bgr = (255,255,255)

img = np.zeros((500, 500))

text = "some text in a box!"

(text_width, text_height) = cv2.getTextSize(text, font, fontScale=font_scale, thickness= 1)[0]

text_offset_x = 10
```

In [ ]:
```

```

In [42]:
```
faceCascade = cv2.CascadeClassifier(  'D:/FaceMask/haarcascade_frontalface_default.xml')
```

In [43]:
```
gray =cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

In [44]:
```
gray.shape
```

Out[44]:
```
(2048, 1152)
```

In [45]:
```
faces = faceCascade.detectMultiScale(gray,1.1,4)
for x,y,w,h in faces:
    roi_gray = gray[y:y+h, x:x+w]
    roi_color = frame[y:y+h, x:x+w]
    cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0, 0), 2)
    facess = faceCascade.detectMultiScale(roi_gray)
    if len(facess) == 0:
        print("Face not Detected")
    else:
        for (ex,ey,ew,eh) in facess:
            face_roi = roi_color[ey: ey+eh, ex:ex + ew]
```

In [46]:
```
plt.imshow(cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))
```

Out[46]:
```
<matplotlib.image.AxesImage at 0x17c829a5e50>
```

REFERENCES

[1] R. Sujatha, Jyotir Chatterjee and Aboul Ella Hassanien, “A machine learning methodology for forecasting of the COVID-19 cases in India,” Apr 18, 2020. [Online].Available https://www.techrxiv.org/articles/preprint/A_machine_learning_methodology_for_forecasting_of_the_COVID-19_cases_in_India/12143685/1. [Accessed Feb. 10, 2021]

[2] Ricardo Navares, Julio Díaz, Cristina Linares and José L. Aznarte, “Comparing ARIMA and computational intelligence methods to forecast daily hospital admissions due to circulatory and respiratory causes in Madrid,” Mar 3, 2018. [Online]. Available: https://link.springer.com/article/10.1007%2Fs00477- 018-1519-z. [Accessed Jan. 31, 2021]

[3] Huijuan Cui and Vijay P. Singh, “Application of minimum relative entropy theory for streamflow forecasting,” Sept 8, 2016. [Online]. Available: https://link.springer.com/article/10.1007%2Fs00477-016-1306 7. [Accessed Feb 1, 2021]

[4] Mohammad Marufur Rahman, Md. Motaleb Hossen Manik and Md. Milon Islam, 2020 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS), " An Automated System to Limit COVID-19 Using Facial Mask Detection in Smart City Network," Oct. 2020. [Online]. Available: https://www.researchgate.net/publication/344563082_An_Automated_System_to_Limit_COVID19_Using_Facial_Mask_Detection_in_Smart_City_Network. [Accessed Feb 2, 2021]

[5] Toshan Meenpal, Ashutosh Balakrishnan and Amit Verma, 2019 4th International Conference on Computing, Communications and Security (ICCCS), " Facial Mask Detection using Semantic Segmentation," Oct. 2019,[Online]. Available: (PDF) Facial Mask Detection using Semantic Segmentation (researchgate.net). [Accessed Feb 2, 2021]

[6] Tejas Saraf, Ketan Shukla, Harish Balkhande and Ajinkya Deshmukh, International Research Journal of Engineering and Technology (IRJET), "Automated door access control system using face recognition," Apr 4, 2018. [Online]. Available: https://www.irjet.net/archives/V5/i4/IRJET-V5I4671.pdf. [Accessed Feb 3, 2021]

[7] Paul Viola and Michael Jone, Conference On Computer Vision And Pattern Recognition 2001, "Rapid Object Detection using a Boosted Cascade of Simple Features," [Online]. Available: https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/violacvpr-01.pdf. [Accessed Feb 3, 2021]

[8] Vinitha. V and Valentina. V, International Research Journal of Engineering and Technology (IRJET), "Covid-19 facemask detection with deep learning and computer vision," Aug 2020. [Online]. Available: https://www.irjet.net/archives/V7/i8/IRJETV7I8530.pdf. [Accessed Feb 4, 2021]

[9] Adrian Rosebrock, "COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning," May 4, 2020. [Online]. Available: https://www.pyimagesearch.com/2020/05/04/covid-19-facemask-detector-with-opencvkeras-tensorflow-and-deeplearning/. [Accessed Feb 5, 2021