REAL-TIME FACE MASK DETECTION USING CASCADED BI-LEVEL FEATURE EXTRACTION TECHNIQUE FOR MANAGING THE SPREAD OF COVID-19

BY

ADAHADA, Enobong Thomas
M.Tech/SICT/2019/9857

DEPARTMENT OF COMPUTER SCIENCE
FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA.

JULY, 2023

REAL-TIME FACE MASK DETECTION USING CASCADED BI-LEVEL FEATURE EXTRACTION TECHNIQUE FOR MANAGING THE SPREAD OF COVID-19

BY

ADAHADA, Enobong Thomas
M.TECH/SICT/2019/9857

A THESIS SUBMITTED TO THE POSTGRADUATE SCHOOL, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA, NIGERIA, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF MASTER OF TECHNOLOGY IN COMPUTER SCIENCE.

JULY, 2023

ABSTRACT

COVID-19 is a respiratory sickness that dealt the human world with one of the deadliest blow in 2020, it is one of the pandemics that has threaten the very existence of humanity in recent times. Its fast spread has caused widespread devastation and infected tens of millions of people around the world. Due to the lack of specific cure for COVID-19, wearing a face mask has proven to be an effective method of reducing its transmission. This is now required in most public venues, resulting in an increase in the demand for automatic real-time mask detection devices to replace manual reminders. This is because people are not willing to wear the face mask and those who do, are not likely to do so the recommended way. This lead scientists and researchers to integrate surveillance technology with Artificial Intelligence (AI) to create a system that identifies people wearing face mask in public areas. Face mask detection necessitates a large amount of data to be processed in real-time or on devices with limited processing resources, therefore local descriptors that are fast to calculate, fast to match, and memory efficient are in high demand. The goal of this research work is to reduce the continuous spread of the deadly pandemic by creating a face mask identification model to classify face images into face mask present and face mask absent. Therefore, this study offers a cascade of Features from Accelerated Segment Test (FAST) corner detector and Histogram of Oriented Gradient (HOG) feature descriptor to allow faster matching and minimize memory usage and computation cost. To achieve this, the images were preprocessed by performing facial landmark identification using viola-Jones algorithm. The resultant images were converted to grey scale images and finally, the images were smoothed by the application of median filter also known as median blur. This filter removes noise from the images. After all the preprocessing steps were carefully carried out, the images were passed to the FAST corner detector to detect the point of interests. These interest points were then passed as input to HOG for feature description. The features were then classified into face mask present and face mask absent using Support Vector Machine (SVM), Naïve Bayes (NB) and Convolutional Neural Network (CNN). The results obtained had a 99.41% accuracy, which was higher than the prior work's 99.27% and 95% accuracy. In addition, the suggested method extracted face features in 48 seconds for training and testing. This study demonstrated that the technique is capable of detecting face masks in real-time.

TABLE OF CONTENTS

Content **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

GLOSSARY OF ABBREVIATIONS

Abbreviation	Meaning
BRIEF	Binary Robust Independent features
BRISK	Binary Robust Invariant Scalable Keypoints
CNN	Convolutional Neural Network
COVID-19	Corona Virus Desease
CT	Computer Tomography Scans
FAST	Feature from Accelerated Segment Test
FMD	Facemask Datase
FRS	Face recognition systems
HOG	Histogram of Oriented Gradient
KNN	K-Nearest Neighbour
LBP	Local Binary Pattern
MFR	Masked Face Recognition
MMD	Medical Mask Dataset
NAAT	Nucleic Acid Amplification Testing
NB	Naïve Bayes
PCR	Polymerase chain reaction
PRNU	Photo Response Non-uniformity
RMFD	Real-World Mask Face Dataset
SIFT	Scale Invariant Feature Transform
SMFD	Simulated Masked Face Dataset
SURF	Speed Up Robust Feature
SVM	Support Vector Machine
WHO	World health Organization

CHAPTER ONE

1.0 INTRODUCTION

1.1 Background to the Study

COVID-19 (Coronavirus Disease 2019) was discovered in 2019 and has had a worldwide impact. When someone that is virus infected sneezes or coughs, COVID-19 is spread through saliva or nasal secretion droplets. It is very contagious and has the tendency to spread quickly especially in crowded areas. Many countries have made and enforced such laws that mandate persons to use safety masks as a result of the outbreak of coronavirus disease (Blundell *et al*., 2020). Governments around the world have taken it upon themselves as a matter of importance to adopt new strategies to effectively manage space, social distance, and supplies for medical personnel and ordinary citizens. Consequently, health facilities and non-health facilities have had to use new infection-prevention mechanisms to reduce the further spread of the deadly COVID-19 disease (Garg *et al*., 2021).

Coronavirus transmission has been demonstrated to be reduced when Face masks are worn (Shahid *et al*., 2020), making it one of the most effective prophylactic methods known (Jiang *et al*., 2021). The World Health Organization (WHO) advises that the face mask should be adjusted such that it covers the nose, mouth, and chin (Fan & Jiang, 2021). The protection provided by masks is considerably decreased or does not exist at all when they are not worn the recommended way. Security personnel are currently deployed in public spaces, advising individuals to wear masks. Nevertheless, because of its inefficiency, this technique exposes the guards to virus-infected air and generates congestion at the doors. As a result, prompt action is essential (Loey *et al*., 2021a). To contain and halt the COVID-19 outbreak, governments will need to provide instructions and watch over people in crowded public areas to ensure that rules that enforces the compulsory wearing of facial safety masks are strictly followed. This might be accomplished by fusing surveillance technologies with artificial intelligence systems. Computer vision is one of the multidisciplinary fields of study that examines how computers learn from digital images (Tripathi & Maktedar, 2019). Picture processing, image categorization, object identification, and image recognition are examples of traditional computer vision tasks. One of the ideal solutions for the identification of face masks is the use of object detection algorithms, which have the capability to locate examples of visual objects of a particular entity in images (Boulos, 2021). In order to help the global society, facial mask detection has become one of the crucial computer vision problems.

Biometric features such as the face, eye, speech, and fingerprint are natural tools for completing identifying tasks such as border controls, e-Government operations, law enforcement agencies, surveillance systems, e-commerce applications, and user authentication and authorization on mobile devices and handheld, among others (Jassim & Asaad, 2018; Wandzik *et al*., 2018). Face biometric traits are commonly used as a form of identification because the face capture procedure is non-intrusive and easy for consumers to use (Raghavendra *et al*., 2020) Many types of documents, in the likes of voters cards, national identity cards, international passports documents, and drivers licences, use the face as a form of identification. Face recognition technologies often automatically identify people based on their biometric facial features (Kenneth *et al*., 2021). One way computer vision can be used to identify masks on people's faces is through face recognition and verification. The main objective of face detection is to pinpoint the specific region in a photo or video that displays a clear image of a face (Ejaz & Islam, 2019). In this study, face detection algorithm is used to ascertain which regions of an image or video should be targeted by a face mask identification system that says that there is the image has the presence of a mask in it.

Capturing frames from a video feed and identifying a human face within it is easily achieved through real-time face detection. The video stream is categorized and divided into key elements that suggest to the model that there are people in the frame. In order to ensure the best outcome, it is imperative to employ an algorithm that can rapidly identify faces, determine if they are masked, and provide entry to communal spaces.

1.2 Statement of the Research Problem

Face masks are required for public use, especially in public areas and large gatherings, in order to contain the spread of the fatal pandemic. This, too, has serious difficulties, since few individuals are willing to wear the face mask, and those who do are likely not to do so in the recommended way, making efforts to stop the fatal pandemic from spreading nearly impossible. Using computer vision, various studies have been undertaken to detect people entering public locations without wearing a facemask. Computer vision technology, such as feature point descriptors, is critical. Because face mask detection requires a considerable quantity of data to be processed in real-time or on devices with limited processing resources, there is an increasing demand for local descriptors that are fast to compute, fast to match, and memory efficient. A useful strategy for accelerating matching and conserving memory is to use short descriptors. Applying dimensionality reduction techniques, such as principal component analysis, on an initial descriptor, in the likes of the Scale Invariant Feature Transform, can provide short descriptors. Scale Invariant Feature Transform (SIFT) (Karami *et al.*, 2017; Rublee *et al.*, 2011) or Speed Up Robust Feature (SURF) (Oyallon & Rabin, 2015; Swapnali & Vijay, 2014) are powerful, but these dimensionality reduction methods applied on them necessitate computing the complete descriptor before proceeding with additional processing, which is time-consuming and computationally complex. Like other techniques, deep neural networks are also computationally complex and require enormous amount of data to function effectively (Zohuri, 2020). This work suggests the use of a cascade of Features from Accelerated Segment Test (FAST) corner detector and Histogram of Oriented Gradient (HOG) descriptor to speed up matching and lessen the need for memory and the complexity of calculation. Combining FAST with HOG will further improve computational speed and accuracy as HOG will only have to describe the points detected by FAST and not the whole points in the image.

1.3 Aim and Objectives

This research aims to create a method of identifying face masks utilizing FAST and HOG. This objective of this study is to guard against COVID-19 continuous transmission.

The objectives of this work are to;

i. Preprocess the data obtained from the online dataset repository.

ii. Develop a cascaded bi-level feature extraction technique.

iii. Classify the features detected from (ii) into Face mask present and no Face mask present.

iv. Assess the method's effectiveness in (ii), we will measure its execution time, precision, recall, accuracy, and f-score.

1.4 Scope of the Study

This research focuses on automatic face mask detection using the FAST, HOG, Binary Robust Invariant Scalable Keypoints (BRISK), Binary robust independent elementary features (BRIEF) and Convolutional Neural Network (CNN). This study did not consider other feature descriptors like the SURF, SIFT, Photo Response Non-uniformity (PRNU) and Local Binary Pattern (LBP). There are several face mask recognition datasets, however, in this study, the model's testing and training was carried out using images from Real-world Masked Face Dataset (RMFD).

1.5 Significance of the Study

There are numerous areas of applications for this subject, its importance cannot be over emphasized. By creating a cascaded bi-level feature extraction method for quick feature extraction and object class detection in digital image and video streams, this research will advance the field of computer vision.

This study will be of benefit to security personals for easy identification of individuals who are not wearing the mandatory face mask when entering into public buildings. Furthermore, researchers will benefit from this study as it gives insight on existing algorithms on face mask detection and classification as well as object detection, thus enhancing the decision-making process towards selecting the appropriate detection and classification technique to implement or to modify towards face mask and object identification.

The study has created a feature extraction method that can be used in existing object detection and recognition algorithms, which can be applied in smart cities and smart homes modules for residents and occupants identification and authorization. It can be integrated with surveillance technology for improved intelligence gathering as well as in immigration and border control to stem the flow of unauthorized border crossing.

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 Overview of COVID-19

Wuhan City, Hubei Province, China experienced an outbreak of pneumonia brought on by the SARS-CoV-2 sickness in December 2019 (Zhu *et al*., 2020a). The illness brought on by SARS-CoV-2 infection was formally dubbed coronavirus disease by the World Health Organization (WHO) on February 11, 2020 (COVID-19). The COVID-19 virus affects the lungs and causes a number of symptoms, such as fever, dry cough, and tiredness. Because SARS-CoV-2 is so contagious, the majority of persons in the general public are at risk of contracting the virus. Currently, the disease is spread by respiratory droplets, close contact with sick individuals, and wild animal hosts (Shi *et al*., 2020).

The SARS-CoV-2 The diameter of a virus ranges from 60 to 140 nanometer, a protein spiked envelope, and genetic code, according to a microscopic picture (Zhu *et al*., 2020b).The general structure resembles that of other *Coronaviridae* viruses. A 30,000-nucleotide long single-stranded positive sense RNA genome is present in SARS-CoV-2 (Wu *et al*., 2020; Zhou *et al*., 2020). RNA-dependent RNA polymerase (RdRP) and four structural proteins are among the 27 proteins encoded by the genome (Sexton *et al*., 2016; Wu *et al*., 2020). To maintain genomic fidelity, RdRP collaborates with non - structural proteins. A section of the RdRP gene in SARS-CoV-2 was determined to be 96% like the RaTG13 overall genome sequence and extremely similar to a part of the RdRP gene discovered in bat coronavirus RaTG13 (Zhou *et al*., 2020). Between December 2019 and mid-February 2020, the genomes of 104 viral strains were sequenced, and it was found that they shared 99.9% sequence homology, although modifications in the viral genome have subsequently been documented, revealing a larger genomic diversity (Tang *et al*., 2020). The SARS-CoV-2 structure is depicted in

Figure 2.1. The image displays SARS-CoV-2 spherical virus particles in a cell as seen through a transmission electron microscope. The virus has a blue tint to it. The structural viral proteins are used to represent the virus's structure (Udugama *et al*., 2020).

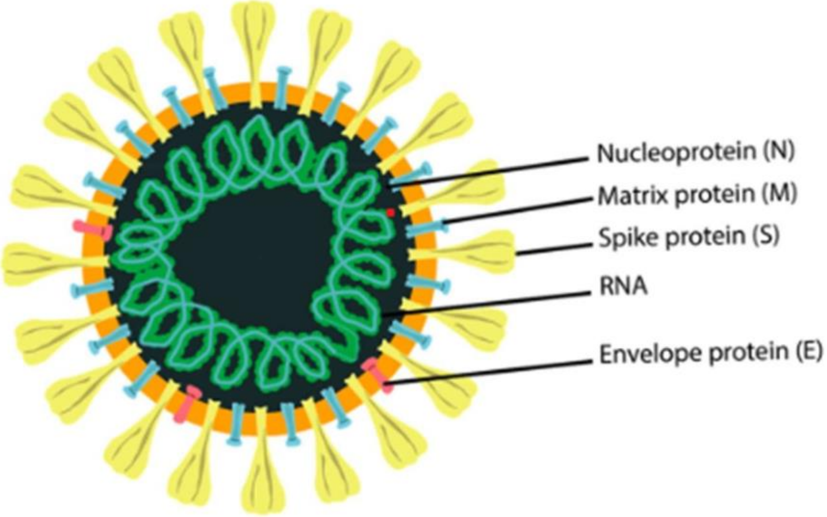

Figure 2. 1 SARS-CoV-2 morphology (Udugama *et al*., 2020)

Small envelope protein (E), spike surface glycoprotein (S), matrix protein (M), and nucleocapsid protein (N) are the four main structural proteins that make up the SARS-CoV-2 virus. Coronaviruses have an S gene that produces the spike protein which helps the virus enter cells by binding to receptors. (Hwang *et al*., 2020). This spike protein is involved in receptor binding and membrane fusion, which affects host tropism and transmission ability (Lu *et al*., 2020). When contrasted to all earlier reported SARS related coronaviruses, the S gene in SARS-CoV-2 is aberrant, with just 75% nucleotide sequence homology. The additional three structural proteins, which are more stable than the spike protein, are required for coronavirus activity in general (Wu *et al*., 2020).These proteins are involved in protein biosynthesis, budding, envelope development, and pathogenicity, as well as enclose the RNA. After learning about the biological characteristics of SARS-CoV-2, researchers were able to develop diagnostics for the identification of Covid-19. (Lim *et al*., 2016; Neuman *et al*., 2011).

2.2 COVID-19 Detection Methods

2.2.1 Molecular diagnosis

Nucleic acid testing is at the heart of molecular diagnosis, which is one of the most significant frontiers in modern medicine. Nucleic acid testing is ideal for reliable coronavirus diagnostic due to its capacity to identify individual viruses. The detection of genetic variations is known as mechanical diagnostics, and it aims to make identification, diagnosis, classification, prognostic, and tracking response to medication easier (McKiernan & Danielson, 2017). The principal infectious agent of the COVID-19 pandemic, the SARS-CoV-2 virus, was identified by the detection of its RNA. Reverse transcription polymerase chain reaction (RT-PCR) encoding the RNA dependent RNA polymerase (RdRP), nucleocapsid (N), envelope (E), and spike (S) proteins of the virus enables sensitive identification of particular genetic sequences (Feng *et al*., 2020). Although RT-PCR tests are widely used and various alternative assays have been developed, current testing capacity and availability are insufficient to meet the unprecedented global demand for quick, accurate, and widely accessible molecular diagnosis (Feng *et al*., 2020). The most popular nucleic acid detection techniques at the moment are nucleic acid amplification assays, CRISPR, and gene sequencing.

2.2.2 Polymerase Chain Reaction (PCR)-based testing methods

PCR is a genetic material detection test for a particular organism, such as a virus (Yang & Rothman, 2004). When a patient is tested, the test will yield a result that indicates the existence of the virus if it is determined that they have been exposed to it. It's important to note that the virus may still be detectable through the test even if the patient has already recovered and is no longer experiencing any symptoms (Azamgarhi *et al*., 2022).

The SARS-CoV-2 virus, which is the source of COVID-19, leaves genetic material (ribonucleic acid or RNA) in your upper respiratory tissue. This genetic material is used in the PCR COVID-19 test (Floriano *et al*., 2021). Small quantities of RNA from specimens are amplified into deoxyribonucleic acid (DNA), which is duplicated until SARS-CoV-2 is identifiable if present, using PCR technique (National Human Genome Research Institute, 2022). Since its approval in February 2020, the PCR COVID-19 test has been the most effective standard for identifying COVID-19. It's precise and dependable (Cleveland Clinic, 2021).

2.2.3 Nucleic Acid Amplification Testing (NAAT)

A NAAT is a form of viral analytic test for SARS-CoV-2 (Deidre *et al*., 2020). The nucleic acids are the genetic code that NAATs detect. SARS-CoV-2 NAATs selectively detect the RNA (ribonucleic acid) sequences that make up the virus's genetic traits (Centers for Disease Control and Prevention, 2021).

NAATs used for SARS-CoV-2 diagnostic specimens from the upper or lower respiratory system. When screening for SARS-CoV-2, the kind of specimen taken is determined by the test being conducted and the maker's recommendations. The NAAT method starts by amplifying, or producing numerous copies of, the virus's genetic material if it is found in the subject's specimen. Due to the amplified nature of these nucleic acids, NAATs are particularly sensitive for the diagnosis of COVID-19 and may detect extremely low quantities of SARS-CoV-2 RNA in a sample. Or to put it another way, NAATs are unlikely to produce a false-negative result and can reliably detect very low amounts of SARS-CoV-2 (Centers for Disease Control and Prevention, 2021a). Reverse transcription polymerase chain reaction (RT-PCR) and isothermal amplification are two of the many techniques that NAATs can employ to amplify nucleic acids and find the virus (Wang *et al*., 2021).

2.2.4 Computed Tomography (CT) Scans

Non-invasive chest CT scans entail capturing many X-ray readings at various angles over a patient's chest to generate cross-sectional pictures (Lee *et al*., 2020; Whiting *et al*., 2015). Radiologists examine the images for aberrant abnormalities that could lead to a diagnosis (Whiting *et al*., 2015). COVID-19 imaging characteristics vary depending on the stage of disease after the start of symptoms. Bernheim *et al*. (2020) found that in the early stages of infection, normal CT findings were more common (56%), with maximal lung involvement reaching about 10 days from the begining of symptoms (Pan *et al*., 2020). Bilateral and peripheral ground-glass opacities (Kobayashi & Mitsudomi, 2013) and pulmonary consolidations are the most prevalent hallmark signs of COVID-19. Ground-glass opacities are most noticeable 0-4 days following symptom start, according to Pan *et al*. (2020). Crazy-paving signs appear as a COVID-19 infection proceeds, in combination to ground-glass opacities, accompanied by increased lung consolidation. Several retrospective investigations have found that CT scans have a better sensitivity (86.98%) and lower false negative rates than RT-PCR based on these imaging parameters (Fang *et al*., 2020; Xie *et al*., 2020).

2.2.5 Point-Of-Care tests

Point-of-care diagnostics are used to diagnose patients without sending samples to processing facilities, enabling regions without adequate laboratory capacity to identify infected individuals (Song *et al*., 2021; World Health Organization, 2020b). The point-of-care detection of COVID-19 can be achieved through rapid test antigen detection (Xiang *et al*., 2020). Two lines are drawn on a tissue ribbon that resembles paper for rapid test experiments. Existing in one line and catching antibodies in the other are gold nanoparticle-antibody conjugates. (Cai *et al*., 2020). Capillary action causes the proteins to travel across the strips once the patient's sample is placed on the membrane. The antigens bond to the gold nanoparticle-antibody conjugates as it passes through the first line, and the pair travels through the membranes together. After finishing the second line, the capture antigens trap the chemical, resulting in a red or blue line. Because of plasmon band coupling, a diluted solution with clustered gold nanoparticles looks blue, whereas the individual particles are red (Centers for Disease Control and Prevention, 2020; Foudeh *et al*., 2012).

Another alternative for use at the point-of-care is a microfluidic system. The gadgets consist of a hand chip with micron-sized reaction chambers and pathways carved into it. To mix and separate liquid samples, the chip makes advantage of electrokinetic, capillary, suction, and other forces. Materials such poly dimethyl sulfoxide, acrylic, and paper can be used to make these chips (Spengler *et al*., 2015). Small size, low sample size, rapid diagnostic times, and mobility are all benefits of adopting microfluidics (Foudeh *et al*., 2012).

2.2.6 Protein testing

Detecting COVID-19 can be accomplished by identifying virus protein antigens and antibodies generated from a SARS-CoV-2 infection (Liu *et al*., 2020). The identification of viral proteins can pose a challenge as the viral load tends to fluctuate during the course of the illness. To-Kelvin *et al*. (2020) has discovered that the peak amount of the virus found in saliva is typically observed during the initial week of symptoms and then experiences a gradual decrease over time. Antibodies produced in reaction to viral proteins, on the other hand, may give a longer allowable time period for diagnosing SARS-CoV-2 inadvertently. Antibody testing are especially helpful for COVID-19 monitoring. Responsiveness of SARS-CoV-2 antigens with immunoglobulin developed vs more coronaviruses is a possible difficulty in generating appropriate serological testing. Cross-reactivity was found in 15 plasma samples obtained from individuals who have tested positive for COVID-19 when Lv *et al*. (2020) compared them to S proteins from SARS-CoV-2 and SARS-CoV.

2.3 Structure of Facial Recognition

The COVID-19 outbreak poses a significant risk to global public health. As a result, national approaches have centered on effective spread reduction. To prevent the virus's import and export, numerous countries have established border controls and travel bans (Udugama *et al*., 2020). The usage of a face mask is a key measure for minimizing virus spread. Face recognition algorithms can quickly identify a mask on a person's face.

Face recognition systems (FRS) were created to detect whether a person is present in a photograph (Kenneth *et al*., 2021). Identifying a person can be done with confidence through face recognition. The process includes four essential steps: feature extraction, face alignment, face detection, and face recognition.

2.3.1 Face detection and localization

Determining the area of a digital photo that denotes a face and detect the location of these faces inside the image are the primary objectives of this initial phase. After completing this phase, the given data can be transformed into patches using each face as the input image. The Viola Jones technique is a prominent face detection and localization technique.

The Viola Jones approach has a higher chance of detection and a reduced chances of false positives. In the Viola-Jones algorithm, the Haar-basis filters, which are a scalar object among the photo and certain Haar-like designs, are utilized (Wang, 2014).

The four elements of the Viola-Jones face identification system are cascade classifier, integral picture selection, AdaBoost training, and Haar feature selection (Viola & Jones, 2004). Prior to using the Viola-Jones face detection algorithm, the input image is converted into an integral image. Calculating the number of pixel distributions in a rectangle in a photo using the integral image is a useful technique.

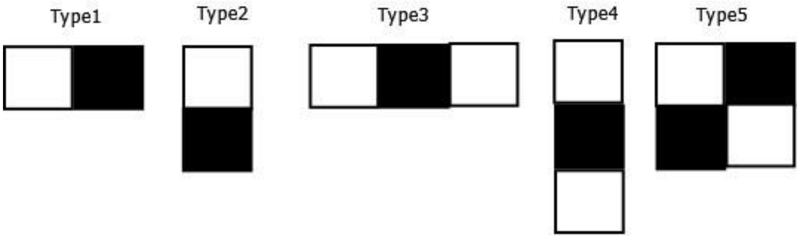

Figure 2. 2 Haar features *(Viola & Jones, 2004)*

The height and width of the Haar characteristics in Figure 2.2 vary. Figure 2.2 illustrates how the image is represented using black and white pixels. Calculating the output is as simple as adding up the black rectangles and subtracting the total of the white rectangles. This straightforward process will yield a single, accurate result. If the estimated value is higher there, it denotes a face feature like the mouth, nose, eyes, or jawline (Deshpande & Ravishankar, 2017). AdaBoost is a machine learning enhancing strategy that can generate a strong classifier from a weighted combination of poor classifiers. Ada boost lowers the number of duplicate features by detecting the crucial and inconsequential features. After that, the AdaBoost assigns weight to all of them after determining which aspects are relevant and which are not. A robust classifier is formed because of linear clustering of weak classifiers (Šochman & Matas, 2004). Equation 2.1 describes a weak classifier statistically.

$$h(\square, \square, \square, \emptyset) = {}_{0}\begin{cases} 1 & \text{if cf(y)}\Sigma c\emptyset \\ \text{otherwic} & \end{cases} \tag{2.1}$$

Where □ is a 24-by-24-pixel sub-window, □ is the implemented feature, □ is the polarity, and ∅ is the threshold for categorizing y as sure (a face) and negative (no face) (Deshpande & Ravishankar, 2017). It is estimated that there are roughly 2500 characteristics in all. As a result, cascading is used to reduce the amount of computations required. The cascaded classifier has a strong classifier at each stage. The purpose of each level is to establish if a certain sub-window is unquestionably not a face or possibly be one. A sub-window is automatically removed when a certain point identifies it as a non-face. On the other hand, a sub-window that may be facing the wrong way is advanced to the next step of the cascade. As a result, the more stages a sub-window udergoes, the more chances it is to have a face in it. The principle is illustrated in three phases in Figure 2.3.

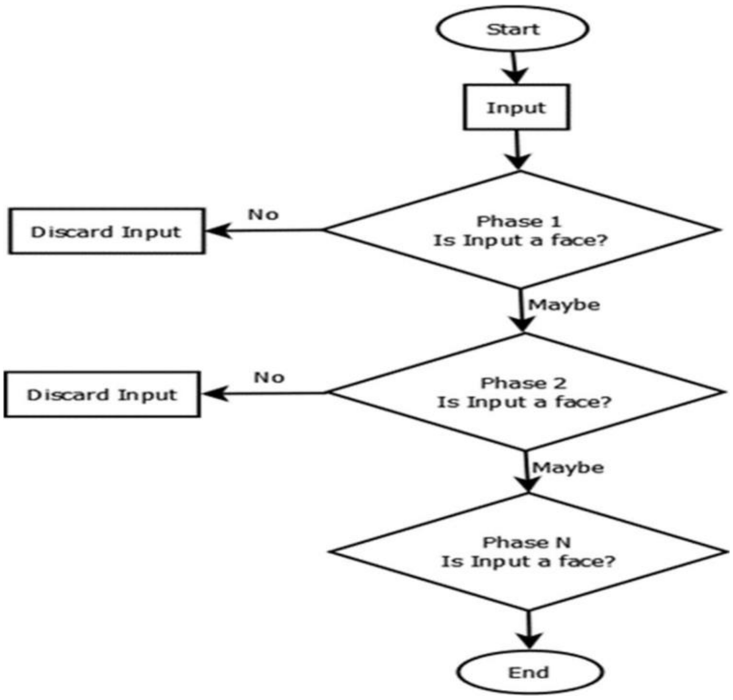

Figure 2. 3 The Cascaded Classifier (Deshpande & Ravishankar, 2017)

To reduce the chances that a prediction is positive when it is actually negative in a single stage classifier, false negatives must usually be considered. False positives are not really considered a problem early in the staggered classifier's development, but they must be handled later. As a result, Viola-Jones advises factoring in a large proportion of false positives at initially. As a result, the final phased classifier's number of false negatives is expected to be quite low. Viola-Jones refers to the cascading classifier as a perceptual cascade (Deshpande & Ravishankar, 2016). This name relates to the fact that parts of the image that are thought to include faces receive increased focus.

2.3.2 Face alignment

Face alignment is a type of computer vision task that identifies the geometrical structure of human faces in digital photographs. The shape of the eyes and nose is precisely calculated by the system, which takes into account the position and size of the individual's face (Xiong & de La Torre, 2013).

During the second stage, the algorithm adjusts the face image to ensure it is consistent with the database. This is achieved by identifying the geometry and photometrics of the face. Essentially, the system scans the face image after it has been captured by the camera (Guo *et al.*, 2020). Facial recognition systems commonly utilize two-dimensional images as they are more compatible with public images and databases (Bulat & Tzimiropoulos, 2017). As the system analyzes an image, it carefully observes and focuses on crucial facial characteristics like the depth of the eye sockets, the contour of the cheekbones, the proximity between the eyes, the distance from the forehead to the chin, and the shape of the lips, chin, and ears. Face alignment is utilized to evaluate the scales, resolution, brightness, zoom levels, and orientations of the patches that were done in the preceding stage, which acts as face recognition preprocessing (Cao *et al*., 2013).

2.3.3 Feature extraction

The system confidently begins its photo processing by normalizing the face and expertly extracting face patches. Its ultimate aim is to effectively translate any image into data that accurately reflects the unique facial features of the individual depicted. The system is prone to extracting the most important data from these photos, where the system should find the most important bits of data while disregarding any noises (Benedict & Kumar, 2017). Facial recognition systems expertly extracts distinctive features and identifies relevance, while efficiently minimizing data size, noise, and dimensions. Every person's faceprint is calculated during face characteristics extraction process. The face characteristics extraction strategies in the setting of masked face recognition may be split into two effective approaches of representing information, that is the shallow and deep approaches. Shallow feature extraction may be said to be a time-honored way of directly formulating a set of custom-made features with minimal adaptation or improvement processes. A number of solutions rely on handmade low-level features to locate and dismiss obscured local parts from identification. LBPs (Nanni *et al*., 2010), SIFT (Satya & Saravanan, 2016), HOG (Attallah *et al*., 2017; Shu *et al*., 2011), and codebooks (Yuan & Park, 2019) is one of the most prominent descriptors for comprehensive learning, local characteristics, and shallow learning methodologies. They already have a reliable accuracy and resilience against several face changes that includes lighting, affine, rotation, scale, and translation in non-occluded face recognition challenges. The performance of shallow features has declined while dealing with occluded faces, such as face masks, and deep representations created by deep learning models have largely overtaken shallow representations. In order to extract features from faces, a number of methods have been developed and tested using deep learning.

Li *et al*. (2020) reasoned that the characteristics of masked faces frequently include mask region-related data that should be represented separately, thus he trained two centers for each category, one for full-facial photos and one for masked facial image, instead of just one. Song *et al*. (2019) Max proposed a multi-stage mask learning technique based mostly on CNN, with the goal of detecting and discarding damaged features from identification. Several additional attention-aware and context-aware algorithms have acquired the critical facial regions by using an extra subnet to retrieve the image attributes (Li *et al*., 2017; Li *et al*., 2018; Wang *et al*., 2018). In the area of masked face identification, restoration, and recognition, graph image reconstructions utilizing deep graph convolutional networks (GCN) is being employed as well. Using spatial or spectral filters created for a shared or fixed graph representation, GCNs have demonstrated an impressive level of proficiency in learning and organizing face photographs. Due to the high computational complexity and the enormous number of GCN layers, learning graph representations is often limited. Researchers have also studied occluded 3D face recognition using 3D spatial features (Alguzo *et al*., 2021; Dagnes *et al*., 2018; Tran *et al*., 2018). The 3D face recognition algorithms are made to mimic actual human vision and understanding of facial traits, which can improve the efficiency of the 2D recognition systems that are now in use. The 3D facial attributes are not significantly affected by a number of face adjustments, including changes in illumination, facial motions, and face directions.

2.3.4 Face recognition

When it comes to the last stage, the system is in charge of differentiating between the faces of different persons. By taking many photos of everyone's face and extracting and saving the details in the system database, a face database may be created so that the system can do automated recognition (Tolba *et al.*, 2014). When necessary, the system then performs face detection and feature extraction from the original photos. After that, the algorithm compares the picture attributes to The 3D face recognition algorithms are designed to closely resemble how humans actually see and comprehend facial characteristics, which can increase the effectiveness of the 2D recognition systems currently in use. Numerous alterations to the face, such as altered lighting, altered facial motions, and altered face directions, barely affect the 3D facial traits. Each recorded faceprint in the database to determine whether access is granted or denied. Face recognition systems can be combined with face mask detection (FMD) systems to identify people wearing or not putting on a mask, reducing the spread of Covid-19. The following section discusses the FMD system, its application, and the FMD techniques.

2.4 Face Mask Detection

The unique Covid-19, which has produced a global disaster affecting more than 172 million people and killing about 3.7 million, according to the WHO's global assessment published on June 2, 2021 (Oumina *et al*., 2020). Individuals, contaminated objects, and human contact are all ways that the virus spreads from person to person. In March 2020, the World Health Organization declared Covid-19 to be a pandemic. (Sanjaya & Rakhmawan, 2020). During the Covid-19 epidemic, America had the highest percentage of verified incidence and mortality, followed by Europe. Due to the peculiar nature of Covid-19 as a newly discovered virus, scientists are attempting to develop effective vaccines to eradicate it.. The pathogen has spread faster than any other in a brief span of time. There are a many of reasons for this heinous spreading, including a lack of awareness, failure to maintain social distance, failure to wear facemasks in public places. According to recent studies, wearing a facemask can considerably slow the spread of the coronavirus and other respiratory diseases (Susanto *et al*., 2020). Despite the fact that coronavirus vaccinations have been created and widely dispersed since early December 2020, they do not actually treat the virus; instead, they just lessen its symptoms and fatality. In order to protect oneself from this infection, it is therefore advised to use a protective mask. This strategy has been shown to be both secure and effective (Fan *et al*., 2021). Because it stops the virus from spreading through the nose and oral cavities, the WHO strongly suggests wearing a facemask in public and outside. In the range of 50% to 95%, facemasks confer resistance to the Covid-19 virus. (Loey *et al*., 2021a). It is crucial to wear a facemask all the time in the aforementioned circumstance to prevent contracting Covid-19. Several administrations have made mask wearing obligatory. 'No mask, no service' labels have been developed to raise awareness.

In this scenario, determining if someone wearing a mask in public or among other individuals has been the subject of extensive investigation. Conventional methods, such as physical force or security, don't always work to identify facemask wearers. Consequently, Facemask Detection (FMD) using machine learning or deep learning is needed. Many studies have been conducted recently to assess whether someone is using a facemask in public as a precaution.

An object detection technique called a facemask detection device (FMD) uses bounding boxes to find and identify facemasks in an image or video stream. To find items, images are classified and their locations are pinpointed. Image classification establishes an object's class. For instance, the facemask classification algorithm will classify images as having "masked faces" or "no-masked faces." The facemask's location is determined by the image localization, and bounding boxes are created around it. Presently, some FMD algorithms only focus on image classification, while others only focus on image localization. Additionally, modern object detecting technologies like You Only Look Once (YOLO) (Loey *et al.*, 2021b), The Single-Shot Detector (SSD) and convolutional neural networks (CNNs) have integrated and trained FMD algorithms. The majority of FMD techniques proposed following Covid-19 are DL-based methods, which are a subset of machine learning (ML) techniques (Jiang *et al.*, 2021; Song *et al.*, 2019).

2.4.1 Application areas of Face Mask Detection (FMD) system

The following are the applications for face mask detection systems:

1. **Public Transport:** To find people travelling without masks on public transportation, FMD systems can be used in airports, train stations, and buses. If a passenger without a face mask is discovered, the relevant authorities can receive the information from the face mask detectors. (Centers for Disease Control and Prevention, 2021b; Kumar *et al.*, 2022).
2. **Healthcare Organizations:** The FMD system in hospitals and healthcare facilities allows for easy identification of isolated individuals who are needed to wear face masks (World Health Organization, 2020a).
3. **Workplaces:** FMD systems are helpful in workplaces like banks so that authorities can take the proper actions, warn staff members and clients to follow instructions to wear masks, and maintain safety standards. (Ramesh *et al.*, 2020; WorkSafe, 2020).

4. **Religious Gathering:** A lot of people gather during religious activities which can easily lead to spread of Covid-19. Hence, FMD systems will be useful in enforcing the mandatory use of facemask at this religious gathering (Baker *et al.*, 2020; Mubarak & Zin, 2020).
5. **Surveillance Systems:** To enforce the need for face mask use in public settings, FMD systems can be linked with current surveillance setups. Face mask detection technology is now crucial for shielding our society from new diseases as a result. (Srinivasan *et al.*, 2021).

2.5 Face Mask Detection Techniques

Masks have become a requirement for people's life since the advent of the pandemic and detecting masks on pedestrians has become a key direction.

Wang *et al*. (2017) proposed a new anchor-level attention algorithm for occluded face detection, which could boost the features of face regions and improve the accuracy without comprising the speed by combining the anchor point allocation strategy and data expansion. However, the work by Wang *et al*. (2017) did not address the issue of mask detection.

Cabani *et al*. (2021) presented masked face images based on facial feature landmarks and developed a huge dataset of 137,016 masked face photos, allowing for more training data. Simultaneously, a smartphone application was created that taught individuals how to properly wear masks by detecting if the masks covered both the nose and the mouth. However, the detecting speed of the models was not addressed.

Saponara *et al*., (2021) proposed a Convolutional Neural Network (CNN) system for social distancing classification of persons using thermal images. The suggested system measured and classified the distance between persons and automatically check if the social distancing rules are respected or not. The proposed approach is applied to images acquired through thermal cameras, to establish a complete AI system for people tracking, social distancing classification, and body temperature monitoring. The training phase is done with two datasets captured from different thermal cameras. Ground Truth Labeler app is used for labeling the persons in the images. The drawback of this proposed approach is that it only works on images captured with thermal camera.

Tomás *et al*. (2021) proposed incorrect face mask-wearing detection. The proposed model uses Convolutional Neural Networks (CNN) with transfer learning to detect whether a mask is used and other errors that are usually not taken into account but may contribute to the virus spreading. This study collected data by requesting participants to take different selfies through an app and place the mask in different positions. A limitation of the proposed model is that it is expensive to train as it consumes a large number of computational resources like memory space and time.

(Fan & Jiang, 2021) constructed a significant face detector model called RetinaFacemask, which has the infrastructure of a backbone network, a neck, and head networks. The backbone network refers to the feature extraction segment in deep learning. RetinaFacemask adopts ResNet as a preliminary backbone, and MobileNet as a backbone for comparison. As a biological neck lies between the back and head, the neck of this framework also implies the strategy. The neck comprises a Feature Pyramid Network (FPN) (Lin *et al*., 2017) built inside the CNN for high-level precision. The head refers to the classifier or the detector, where a context attention module has been introduced to increase the detection performance. In this algorithm, transfer learning is applied because of the limited dataset. The RetinaFacemask model is comprised of a very strong network, which sometimes results in a high computation overhead.

Chavda *et al*. (2021) developed a two-stage-based detector using two pre-trained CNN models. The first stage of the detector completes the face detection in an image, and the next stage classifies the detected images into a mask and no-mask class. Various datasets are combined to produce versatile and geologically bias-free data. In the first stage, the input RGB image was passed through a face detector that could detect faces, even though any scenario of two overlapping faces occurs. The ROI was extracted and passed to the next stage detector, which classified the faces retrieved from the first phase as masked face or unmasked face. The development of a face detector requires a considerable amount of completely trained datasets and a lengthy processing time. Therefore, the authors chose the RetinaFacemask (Deng *et al*., 2020) as the first stage detector and NASNetMobile (Zoph *et al*., 2018) as the classifier model for comparison. Between the mentioned phase, there is another intermediary phase that collects the detected faces from stage one and batches them, enlarges the bounding box of faces according to height and width, and resizes them according to the requirement to pass through the next stage for classification. The algorithm would allow a further extension, where live video streams could be used as an input. There were some drawbacks of this model. It used two different detectors that created complexity. In addition, the video frame rate was comparatively low.

Nieto-Rodríguez *et al*. (2015) introduced a real-time face-mask detection system that triggers an alarm when healthcare staff do not wear surgical masks in the medical or operating room. Two detectors and two-color filters for each detector were used. One of them was a face detector, and another was a medical mask detector. The face detection was done using the traditional Viola-Jones face detection algorithm. A variant AdaBoost called LogitBoost for detecting face masks was used. One of the challenges was that any clothing near the mask area could give a false mask detection result because it is based on color filtering.

Vinitha & Velantina, (2020) proposed a model based on computer vision and a deep learning approach. This model was capable of real-time facemask detection from surveillance cameras and images. They used TensorFlow, OpenCV, Keras, and MobileNetV2 architectures in their model, which was trained on a large dataset. In this model, most of the images were added by OpenCV. Pre-processing of the input images was achieved by resizing them, and they were applied to color filters (RGB) over the channels. The images were scaled using the standard mean of Pytorch build in weights 4.5. Finally, it was converted to tensors (Similar to NumPy array). The model was tested with real-time images and real-time video streams. However, the dataset used, and the performance of the proposed system was not reported. The report only mentioned the use of a large dataset.

Msigwa *et al*. (2022) developed a method for facemask detection from speech. This model consisted of two parts, i) training Generative Adversarial Networks (GANs) with cycle -consistency loss to transform the unpaired utterances in between two classes (with mask and without mask), ii) Assigning opposite labels to each transformed pronunciation, producing new training accents using cycle-consistent GANs. The initial and transformed accents were converted to spectra that were used as input to ResNet networks with different depths. The networks were grouped by classifying the SVMs. In this process, augmented spectrograms were used to train the model. Training spectrograms were also converted from one class to another class using G and G'. The ResNet was used; the start layer was 18, and the end layer was 101. All the outputs of ResNet were mixed in concatenated feature vectors and were considered as input for the SVM, which predicts the result. The datasets were provided by the ComParE organizers having 36554 samples. Among them, 10,895 samples were used for training, 14,647 samples for development, and the remaining samples for testing. It was reported that the data augmentation method yielded better results than other baseline methods. The model required high processing time. Therefore, the ratio of the consumption of time and accuracy is the main drawback of this model.

Loey *et al*. (2021b) introduced a novel deep learning-based facemask detection model by implementing YOLOv2 (Redmon & Farhadi, 2017) and ResNet-50 (He *et al*., 2016) together. They paid attention to medical or surgical facemasks. The YOLOv2, an updated version of YOLO, is a feature extracting and classification algorithm, and ResNet-50 is mainly a deep transfer learning-based residual network for feature extraction. Facemask Dataset (FMD) (Larxel, 2021) and Medical Mask Dataset (MMD) (Shretash, 2020) were combined to test and train the model. A data augmentation strategy was used to manipulate the data. An estimation of the anchor boxes was used to improve the model. Two optimizers (SGDM (Sutskever *et al*., 2013) and ADAM (Kingma & Ba, 2014)) were used to compare their performances. The ADAM optimizer reaches greater accuracy than SGDM. The mean Intersection over Union (IoU) (Rezatofighi *et al*., 2019) is used to estimate the anchor boxes. The model could not identify masked faces from videos.

Bu *et al*. (2017) proposed a cascaded CNN architecture for masked face identification, which was comprised of three complete CNN. They also proposed a dataset and called it the ‘‘MASKED FACE Dataset.’’ Their dataset contained only 200 images, which are not sufficient to feed into any deep learning-based framework. Deep learning-based frameworks require a large number of datasets for training to achieve higher performance. To overcome this problem, a pre-trained model with the WiderFace dataset (Yang *et al*., 2016) was used, and then tuned with their dataset. The first, second, and third CNN architectures have the number of five, seven, and seven layers, respectively. The first layer is a very shallow CNN that scales the input image. The classification ability of the CNNs ranges from low to high. Each of the first scales the image and then evaluates the result according to a pre-set threshold. If the evaluated value or probability is less than the given threshold, it is considered a false detection and is discarded. An attribute called the non-maximum suppression is added after the three CNNs to unify the overlapped candidate windows. Here, the use of three cascaded CNNs has an advantage and disadvantage. The three-stage CNN makes the prediction stronger because each false detection is eliminated and is not calculated. The demerit is that it also increases computational complexity.

Khandelwal *et al*. (2020) proposed a model and deployed it in a real-life application that detects whether a mask is used or not in an image. The work was separated into twosteps. One was face detection in the image, and the other was mask classification. They used a pre-trained CNN-based MobileNetV2 model for face detection. A face size less than a certain number of pixels cannot be detected using this model. After detecting faces, the data was prepared to feed into the mask detection stage by cropping and labeling faces using semi-supervised learning (Zhou Z.-H., 2021). MobileNetV2 was used to build the model. Before feeding into the network, the images were resized in accordance with their requirement. They also used an augmentation strategy to bring diversity to the data. They took a validation set of 840 images combined with a mask and no mask among 4,225 annotated images. This work achieved high performance and was already implemented, but the model had two major drawbacks. First, classification or detection of partially overlapped faces cannot be done using this method. Secondly, this model cannot detect faces where the height of the camera exceeds 10 feet.

Militante and Dionisio (2020) proposed a simplistic real-time mask detection framework using VGG-16 CNN. Their dataset was labeled and contained 25,000 images to train the model. Although they did not mention the source of their dataset, the number was quite acceptable. The images were preprocessed to avoid unnecessary information like other models, and the segmentation and extraction of the mask-covered area from the face then take place. The classification was done using the VGG-16 CNN model. In the training phase, they used the ADAM optimizer for optimizing the parameters (Kingma & Ba, 2014). ADAM is derived from the term estimation of adaptive moment. Comparisons with other state-of-the-art models are not described here.

Ejaz and Islam (2019) proposed a model for face mask recognition. The proposed approach consists of first detecting the facial regions. Then facial features extraction is performed using the Google FaceNet embedding model. And finally, the classification task was performed using Support Vector Machine (SVM). The model performance was evaluated using accuracy without considering other performance measures such as time, precision, and f-score.

Loey *et al*. (2021a) adopted the hybrid transfer learning model and machine learning methods for better feature extraction and classification. The proposed model consists of two components. The first component is designed for feature extraction using Resnet50. While the second component is designed for the classification process of face masks using decision trees, Support Vector Machine (SVM), and ensemble algorithm. Three face masked datasets were used. The Three datasets are the Real-World Masked Face Dataset (RMFD) (Tomás *et al*., 2021; Wang *et al*., 2020), the Simulated Masked Face Dataset (SMFD), and the Labeled Faces in the Wild (LFW) (Alzu'bi *et al*., 2021). The final accuracy reached 99.64% on the RMFD, 99.49% on the SMFD, and 100% on the

Labeled Faces in the Wild LFW dataset. Nonetheless, the work concentrated on the accuracy of detecting masks, and the speed of detection was not properly addressed.

Jiang *et al*. (2021) proposed Squeeze and Excitation (SE)-YOLOv3 and created a mask detector with relatively balanced effectiveness and efficiency. The attention mechanism was integrated by introducing the SEblock into Darknet53 to obtain the relationships among channels so that the network can focus more on the important features. The research adopted GIoUloss, to better describe the spatial difference between predicted and ground truth boxes to improve the stability
of bounding box regression. Focal loss was utilized for solving the extreme foreground-background class imbalance. To further improve the robustness of the proposed model on the specific task, the research carried a corresponding image augmentation technique. The research came up with a fast and accurate mask detector that include a channel attention mechanism that to enhance feature extraction mechanism. The problem with this research is that the dataset was significantly small, and deployment on portable devices was not taken into consideration, precision, F1-score was not addressed.

Alzu'bi *et al*. (2021) presented a comprehensive survey of the recent Masked Face Recognition (MFR) works based on deep learning techniques. The study identified use of real-world faces with masks in the benchmarking datasets remains avital challenge for the effectiveness of MFR systems. Despite the availability of data augmentation and face masking tools that generate synthetic face masks, there is a demand to evaluate the MFR algorithms under different types of real masks including textured masks. The deep learning-based techniques for most MFR scenarios encounter enormous algorithmic complexities during the training phase and therefore require computational power during testing and operation, which is unfavorable for compact devices and real-time systems. The review only considered deep learning techniques for MFR without considering shallow learning methods for MFR.

Dey *et al*. (2021) proposed a MobileNet Mask model, which is a deep learning-based multi-phase face mask detection model for preventing human transmission of SARS-CoV-2. Two different face mask datasets were utilized to train and test the model for detecting with and without a face mask from images and video stream. The first dataset consists of 3835 images belonging to two different classes including 1916 images of with mask and 1919 images of without mask. The sample images of first dataset were gathered from numerous sources including the Kaggle datasets, Bing search API, and the Real-World Masked Face Dataset (RMFD). The second dataset consists of 1376 images belonging to two different classes including 690 images of with mask and 686 images of without mask. The second dataset is based on simulated or artificially created Masked Face Dataset (SMFD). Experimental results show that with 770 validation samples the proposed method achieved an accuracy of approximately 93% whereas with 276 validation samples it attains an accuracy of nearly approximately 100%. The face mask identification model speed, on the other hand, was not considered given that the proposed model will be used in real time.

Teboulbi *et al*. (2021) developed a real-time implementation of face mask detection and social distancing measuring system using AI. The research paper focuses on implementing a Face Mask and Social Distancing Detection model as an embedded vision system. the pretrained models such as the MobileNet, ResNet Classifier, and VGG are used in this research context. A confidence score of 100% was achieved on deployment of the model. The paper also provides a comparative study of different face detection and face mask classification models. the system performance is evaluated in terms of precision, recall, F1-score, support, sensitivity, specificity, and accuracy that demonstrate the practical applicability. The system achieved a F1-score of 99%, sensitivity of 99%, specificity of 99%, and an accuracy of 100%. The solution tracks the people with or without masks in a real-time scenario and ensures social distancing by generating an alarm if there is a violation in the scene or in public place. The limitation of this work is the execution time that was not accounted for.

Qin and Li (2020) identified that the effectiveness of using facemask to minimize the spread of covid-19 has been diminished, mostly due to improper wearing. Hence, study proposed a new facemask-wearing condition identification method by combining image super-resolution and classification networks (SRCNet), which quantifies a three-category classification problem based on unconstrained 2D facial images. The proposed algorithm contains four main steps: Image pre-processing, facial detection and cropping, image super-resolution, and facemask-wearing condition identification. The proposed method was trained and evaluated on the public dataset Medical Masks Dataset containing 3835 images with 671 images of no facemask-wearing, 134 images of incorrect facemask-wearing, and 3030 images of correct facemask-wearing. The proposed SRCNet achieved 98.70% accuracy. Nevertheless, there were several limitations. The dataset was relatively small and lower in attributes. The detection speed was slow.

Joshi *et al*. (2020) proposed deep learning framework to detect face masks from video footage. This is an approach for detecting facial masks in video steams using deep learning. The proposed framework relies on the MTCNN face detection model to identify the faces and their corresponding facial landmarks present in the video frames. These facial images and cues are then processed by a neoteric classifier that uses the MobileNetV2 architecture as an object detector for identifying masked regions of the facial images. After testing the proposed model, the methodology demonstrated its effectiveness in detecting facial masks by achieving high precision, recall, and accuracy values on the chosen dataset which contained videos with varying occlusions and facial angles. The limitation of this work however includes the paper did not mention the various factors that may affect effective face detection on video streams like motion blur. The paper also did not address the model performance in terms of speed, F1-score since the effectiveness of the facial mask classifier largely confides on the ability of the face detection algorithm to accurately identify faces in the video frames.

In summary most existing works on face mask detection made of the deep learning-based models. However, Deep learning-based techniques for most mask face detection and recognition scenarios encounter enormous algorithmic complexities during the training phase and therefore require computational power during testing and operation, which is unfavorable for compact devices and real-time systems. This research work makes use of HOG short descriptor and FAST corner detector to reduce computational complexities and resource consumption. Table 2.1 gives a summary of the related literature.

Table 2.1 Summary of Related works

S/N	AUTHOR(S) and YEAR	TECHNIQUE	STRENGTH	WEAKNESSES
1	Wang *et al.* (2017)	Attention Network	Proposed method boosts the features of face regions and improve the accuracy without comprising the speed	Did not address the issue of mask detection
2	Cabani *et al.* (2021)	Facial feature landmark	Developed et of 137,016 masked face images	speed of the models was not addressed
3	Saponara *et al*., (2021)	Convolutional Neural Network (CNN)	Performs people tracking and social distancing classification	Approach only works on images captured with thermal camera
4	Tomás *et al.* (2021)	CNN	Detects incorrect face mask wearing	Model is expensive to train as it consumes a large number of computational resources like memory space and time
5	Fan & Jiang (2021)	ResNet and MobileNet	Can be applied to limited or small dataset	High computation overhead
6	Chavda *et al.* (2021)	CNN	Combination of various dataset to produce versatile and geologically bias-free data.	It used two different detectors that created complexity.
7	Nieto-Rodríguez *et al*. (2015)	LogitBoost	Real-time triggering of alarm system for people without mask	Any clothing near the mask area could give a false mask detection result because it is based on color filtering
8	Vinitha and Velantina (2020)	MobileNet	The model was tested with real-time images and real-time video streams	The dataset used, and the performance of the proposed system was not reported

Table 2. 1 (CONT)

S/N	AUTHOR(S) and YEAR	TECHNIQUE	STRENGTH	WEAKNESSES
9	Msigwa *et al.* (2022)	Generative Adversarial Networks and ResNet	Uniquely performed facemask detection from speech.	The model required high processing time.
10	Loey *et al.* (2021b)	CNN	The use of anchor boxes to improve the model accuracy.	The model could not identify masked faces from videos.
11	Bu *et al*. (2017)	CNN	The three-stage CNN makes the prediction stronger because each false detection is eliminated and is not calculated.	High computational cost
12	Khandelwal *et al.* (2020)	CNN-based MobileNetV2	High performance	Detection of partially overlapped faces cannot be done using this method. And the model cannot detect faces where the height of the camera exceeds 10 feet.
13	Ejaz and Islam (2019)	CNN and Support Vector Machine (SVM)	Achieved high accuracy result	The model performance was evaluated using accuracy without considering other performance measures such as time, precision, and f-score
14	Loey *et al.* (2021a)	ResNet50, ensemble, decision tree and SVM	Robust model as three datasets were used for training and testing	Concentrated on the accuracy of detecting masks, and the speed of detection was not properly addressed.
15	Jiang *et al*. (2021)	CNN	Made use of attention network to obtain the relationships among channels so that the network can focus more on the important features.	Dataset was significantly small, and deployment on portable devices was not taken into consideration

Table 2. 1 (CONT)

S/N	AUTHOR(S) and YEAR	TECHNIQUE	STRENGTH	WEAKNESSES
16	Alzu'bi *et al.*, (2021)	Review	In-depth review of the recent Masked Face Recognition (MFR) works based on deep learning techniques	The review only considered deep learning techniques for MFR without considering shallow learning methods for MFR
17	Dey *et al*. (2021)	MobileNet	Made use of diverse dataset	The model speed was not considered given that the proposed model will be used in real time.
18	Teboulbi *et al.* (2021)	MobileNet, ResNet, and VGG	The solution tracks the people with or without masks in a real-time scenario and ensures social distancing by generating an alarm if there is a violation in the scene or in public place	Execution time that was not accounted for.
19	Qin and Li (2020)	Super-resolution and classification networks (SRCNet)	Identification of incorrect facemask wearing	The dataset was relatively small and lower in attributes.
20	Joshi *et al*. (2020)	CNN	Detects face mask on both images and videos	The paper did not mention the various factors that may affect effective face detection on video streams like motion blur.

From the review of existing works in Face Mask detection, it is seen that most of the existing techniques did not address the speed of the model and execution time of feature extraction. Some that addressed the speed problem however did not consider computational complexities and resource management in terms of memory consumption when deployed on compact devices. The size of dataset was considerably in some studies small and a specific type of camera was needed in some others.

CHAPTER THREE

3.0 RESEARCH METHODOLOGY

This chapter presents the description of the methods used in carrying out this research. These techniques include data collection, image preprocessing, feature extraction, and data classification. The research methodology of this work is illustrated in Figure 3.1. The cascaded bi-level feature extraction algorithm is shown in Table 3.1.

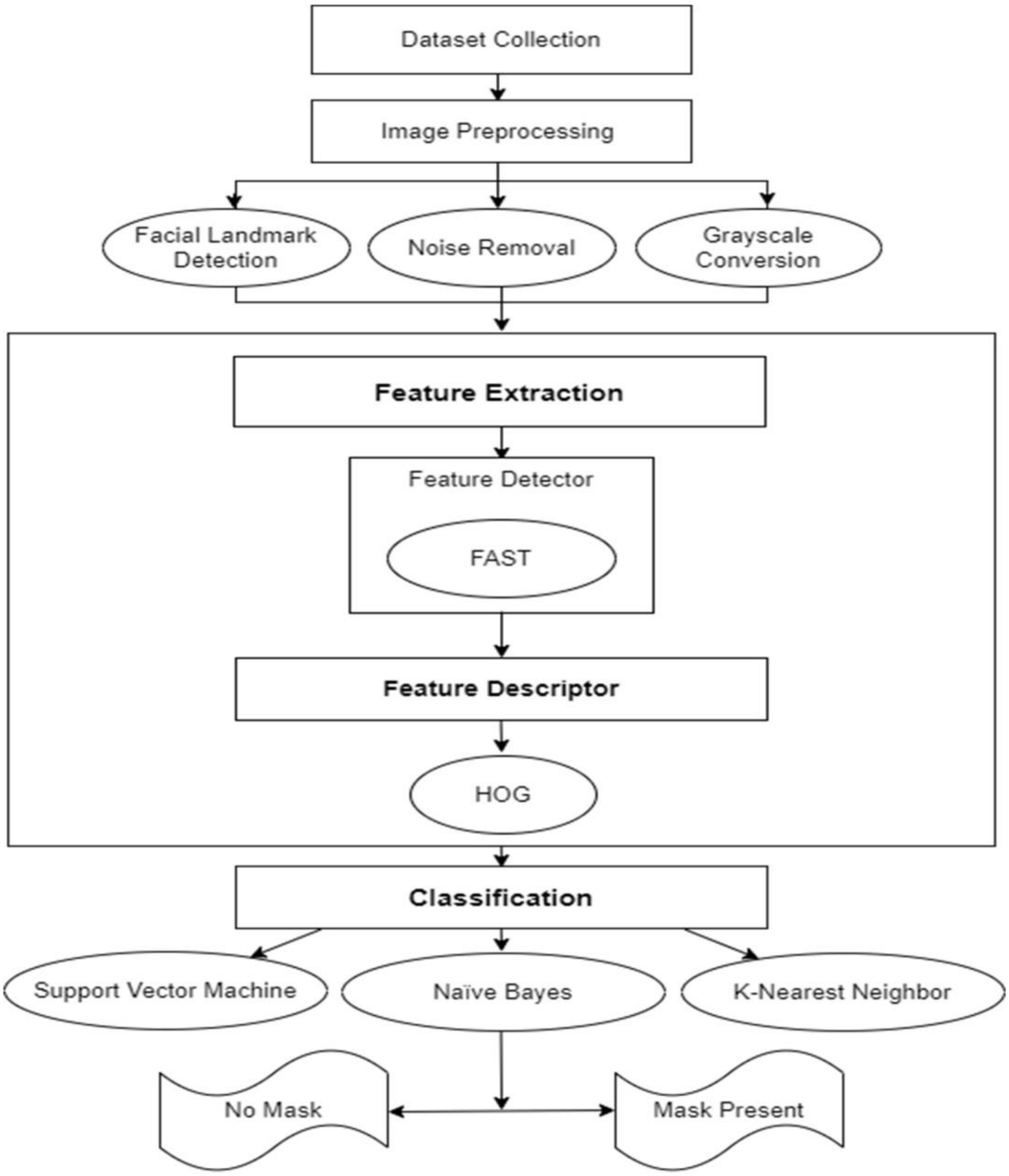

Figure 3. 1 Research Methodology Model

Table 3.1 Cascaded bi-level feature extraction algorithm

Cascaded bi-level feature extraction algorithm

```
Start
Require: Get images from online repository
  while dataset is imbalanced do
        treat imbalance in image dataset
   end while
    apply viola-jones to perform facial landmark identification
    convert images to greyscale images
    while images are noisy do
        apply median filter to remove noise
    end while
    for i=0 I < = images in dataset do
        if image is smoothed then
              pass images to FAST for corner detection
        else
            discard image
        end if
        pass detected corners to HOG for description pass features to SVM for classification
        if mask is present then
            output 1 and grant entrance
        else
            output 0 and refuse entrance
        end if
     end for
End
```

3.1 Tools and Materials

MATLAB R2020b was used to conduct the experimentation from the image preprocessing, feature extraction down to the classification and performance evaluation phase.

MATLAB is a proprietary multi-paradigm programming language and numeric computing environment developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs and libraries written in other languages like C++. It is one of the programing languages that is useful in the designing of AI models and AI-driven systems therefore, it is very suitable for Artificial Intelligence applications. MATLAB has a far reaching impact and explores a wide range of product capabilities that provide solutions to AI driven problems. Engineers and scientist around the globe use MATLAB to solve some of the biggest data and AI model driven challenges.

The choice of MATLAB for this thesis is due to the fact that the software was designed to handle problems in Machine Learning, Data analytics which is best for designing and testing, condition monitoring and predictive maintenance algorithms. The graphs were designed using Microsoft Word Graph designer for Office 2021 Pro.

3.2 Data Collection

The face mask detection task must identify whether a person wears a mask as required. The Face mask detection task requires masked face image samples. This research used the Real-world Masked Face Recognition Dataset (RMFRD) datasets to train and test the proposed model. The RMFRD dataset was selected in this study because it consists of real face mask images given that most available datasets are Synthetic and does not portray real life scenario. This dataset is present at https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset.

3.2.1 Real-world Masked Face Dataset (RMFD)

This dataset was developed by Wang *et al*. (2020). A python crawler tool was used to crawl the front-face images of public figures and their corresponding masked face images from massive Internet resources. Then, the unreasonable face images resulting from wrong correspondence were manually removed. Finally, the accurate face areas were cropped with the help of semi-automatic annotation tools, like LabelImg and LabelMe. The database contains 90,000 face images without masks, 5000 face images with masks belonging to 525 subjects. Figure 3.2 and 3.3 shows a pair of face images.

Where figure 3.2 is a face image without a mask, whereas figure 3.3 is a face image with a mask.

Figure 3. 2 Face image without mask

Figure 3. 3 Face image with mask

In this research, given that there are 90,000 images without a mask and only 5000 images with masks, the dataset is imbalance. To treat this data imbalance, under sampling was carried out on the majority class which represents the images without mask. Thus, some of the images without face mask were randomly removed leaving 6000 face images without masks. Meanwhile all the 5000 faces with masks were used for the model training and testing. This makes the used dataset a total of 11000 images. In the training phase, 80% of the total dataset was used for training, while the remaining 20% of the dataset were used to test the trained model.

3.2.2 Data imbalance treatment

Data imbalance is a scenario where there is a skew in the data classes represented. Data imbalance refers to a situation where the distribution of classes or labels in a dataset is not uniform, resulting in some classes being underrepresented compared to others. This can negatively impact the performance of machine learning models, as they may be biased towards the majority class and have reduced accuracy for minority classes. That is the class that has more members than the other class. For instance the RMFD is a good example of a skewed dataset. Here the class of data that has no mask is 90,000 images while the class of data that has mask is only 5,000. This is not suitable for the application of such performance metrices as accuracy, as the class with the larger number of members will always be classified by the classification model as true, which is not the true picture of the dataset. This is an issue in this work as Accuracy is one of the performance metrics that will be considered in measuring the effectiveness of the classification model.

There are several techniques that can be used to treat data imbalance in a dataset, however, it is important to note that there is no one-size-fits-all solution for treating data imbalance, and the choice of technique to use depends on the specific dataset and problem at hand.

Some of the techniques include:

i. Using different evaluation metrics

ii. Ensemble methods

iii. Cost-sensitive learning

iv. Using different algorithms

v. Collecting more data

vi. Resampling.

For the purpose of this research work, resampling method is used.

This technique involves either oversampling the minority class or under sampling the majority class to balance the class distribution. This method is selected for this research work because it is simple to implement and the removal of images without face masks will not affect the decision making of the classification model since all images either have mask on or no mask. The quality of the images is not affected in any way as all images are real images not synthetics or computer generated images.

Oversampling techniques include random oversampling, where instances of the minority class are duplicated, and synthetic oversampling, such as SMOTE (Synthetic

Minority Over-sampling Technique), which generates synthetic examples of the minority class.

Under sampling techniques include random under sampling, where instances of the majority class are randomly removed, and Tomek links, which identify and remove ambiguous instances from both classes. Under sampling operation will be carried out on the majority class sample to balance the dataset.

3.3 Image Preprocessing

In the Image preprocessing stage, three operations were performed. These operations include Facial landmark detection, grayscale conversation, and noise removal. Each of these processes is discussed in detail in the subsections below.

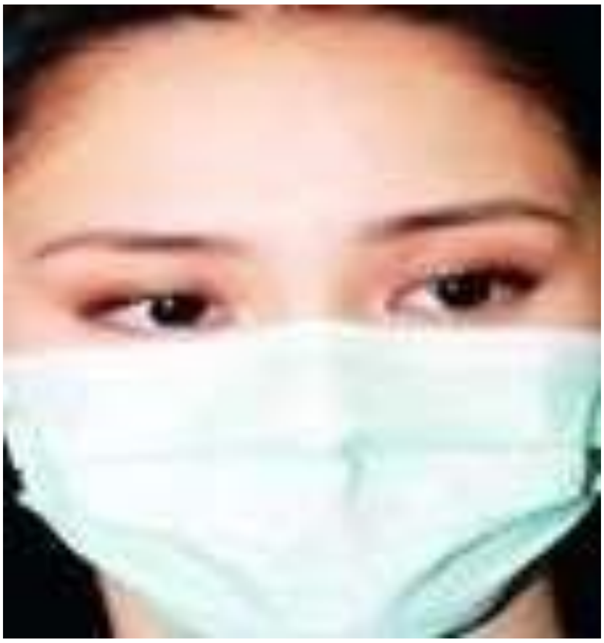

Figure 3.4 Masked Face image before image preprocessing steps

3.3.1 Facial landmark detection

Facial features, such as the nose, eyes, lips, brows, and jawline, are employed to restrict and signify significant regions of focus (Scherhag *et al.*, 2019; Wang, 2014). The Viola-

Jones approach was used to detect face features in this study. The Viola-Jones computation employs Haar-basis filtering, a scalar object in the middle of the image, and various Haar-like structures (Wang, 2014). Haar feature selection, integral photo screening, AdaBoost training, and a cascading classifier are the four phases of this approach for face recognition (Viola & Jones, 2004). The input image is first converted into an integral image via the Viola-Jones face detection algorithm. The integral image is a method for generating the pixel sum intensities in a square of a picture in an operational manner. Table 3.2 shows the Viola-Jones algorithm.

Table 3.2 Voila-Jones face detection algorithm

Voila-Jones face detection algorithm

```
Require: Selecting Haar-like features
Ensure: Creating an integral image
if Integral image created then
    Ensure: Run AdaBoost training
    Ensure: Create classifier cascades
end if
```

3.3.2 Grey-Scale conversion

A grey-scale image in digital image processing is one in which a single sample representing only a quantity of light is the value of each pixel; that is, it holds only intensity values. The pictures in grey scale, a kind of grey monochrome, are made entirely of shades of grey at the lowest intensity. The contrast varies from black to white at the highest (Saravanan, 2010). In this phase, the cropped RGB or coloured face images were converted to a grey-scale image to prepare the images for feature extraction. Figure 3.5 shows example of the image of the masked face in figure 3.4 converted to grey scale or monochrome image format. This is a necessary step for further image processing in this research work, although it is not a mandatory step in all image processing.

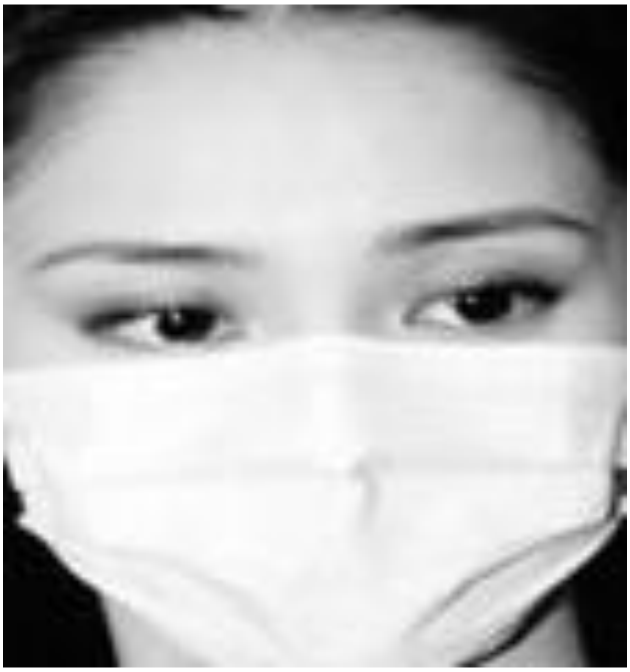

Figure 3.5 Masked Face image after Grey-Scale conversion

3.3.3 Noise removal

Image de-noising is a critical operation in picture processing for image analysis. For image de-noising, the median filter is utilized. The median filter is a non-linear filter that responds to the ranking of pixel values inside the filter zone (Verma *et al.*, 2013). The median filter is widely used to reduce certain types of noise. The median of the pixel values under the filter zone takes the place of the pixel's center value. For salt and pepper noise, the median filter is useful. These filters are commonly employed as image smoothers as well as in signal processing. The median filter has a significant benefit over linear filters in that it may eliminate the effect of input noise values with exceptionally large magnitudes (Iftikhar & Mohammed, 2011). The image in figure 3.6 shows the resultant image in figure 3.6 after the application of median blur to remove noise and therefore smoothing the image for further image processing activities.

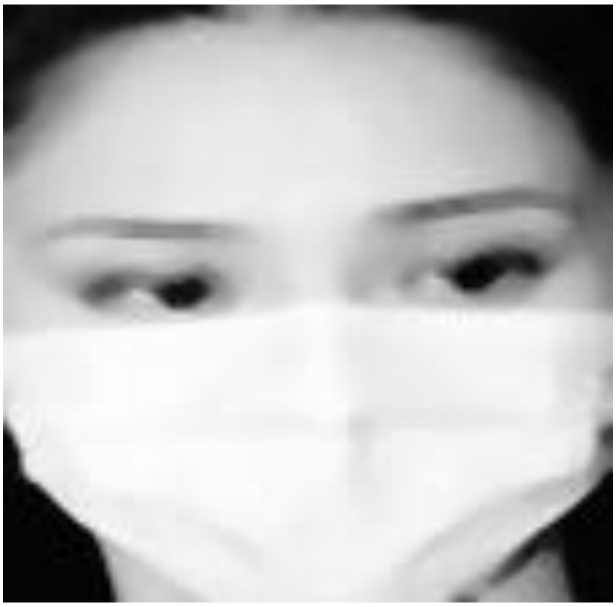

Figure 3.6 Masked face image after smoothing and noise removal

3.4 Feature Extraction

Image features contain essential information such as points and edges that are vital for image analysis. Images features are extracted using several techniques. In this research, the Feature from Accelerated Segment Test (FAST) techniques is used for corner detection and the Histogram of Oriented Gradient (HOG) method will be used for feature description.

3.4.1 Feature from Accelerated Segment Test (FAST)

FAST is an existing algorithm for the identification of interest points in an image originally introduced by Rosten and Drummond (Zhang *et al*., 2016). FAST uses one variable, which is the threshold of intensity between the middle pixel and the ones in a circular ring around the middle (Kulkarni *et al*., 2013). FAST is measured easily and quick to match. The precision is pretty good, too. FAST does not represent a scale-space detector, so the detection of the edges at the particular scale can produce much more than a scale-space technique like SIFT (Karami *et al*., 2017; Rublee *et al*., 2011).

FAST corner detector uses a circle of 16 pixels to classify whether a candidate point p is actually a corner. This is illustrated in figure 3.7. Each pixel in the circle is labelled from integer number 1 to 16 clockwise. If a set of N contiguous pixels in the circle are all brighter than the intensity of candidate pixel *p* (denoted by $\square_{P}$) plus a threshold value *t* or all darker than the intensity of candidate pixel *p* minus threshold value *t*, then *p* is classified as a corner (Ghahremani *et al*., 2020). The conditions can be written as:

- **Condition 1:** A set of N contiguous pixels S, $\forall \square \in \square$, the intensity of $\square > \square_{P} +$ threshold, or $\square_{S} > \square_{P} + \square$
- **Condition 2:** A set of N contiguous pixels S, $\forall \square \in \square$, $\square_{S} > \square_{P} - \square$

So, when either of the two conditions is met, candidate *p* can be classified as a corner. The FAST algorithm is given in Table 3.2.

Table 3.3 FAST algorithm

FAST algorithm

```
while There is an input Image do
Require: Select a pixel p in the image which is to be identified as an interest point or not. Let
   its intensity be Ip
Ensure: Select appropriate threshold value t
Ensure: Consider a circle of 16 pixels around the pixel under test. (This is a
   Bresenham circle of radius 3) initialization;
      if n contiguous pixels in the circle (of 16 pixels) which are all brighter than
   Ip + t then
            pixel p is a corner
      else pixel p is a not corner, discard p
      end if
      if n contiguous pixels in the circle (of 16 pixels) which are all darker than
   Ip - t then pixel p is a corner
      else pixel p is a not corner, discard p
end if
endwhile
```

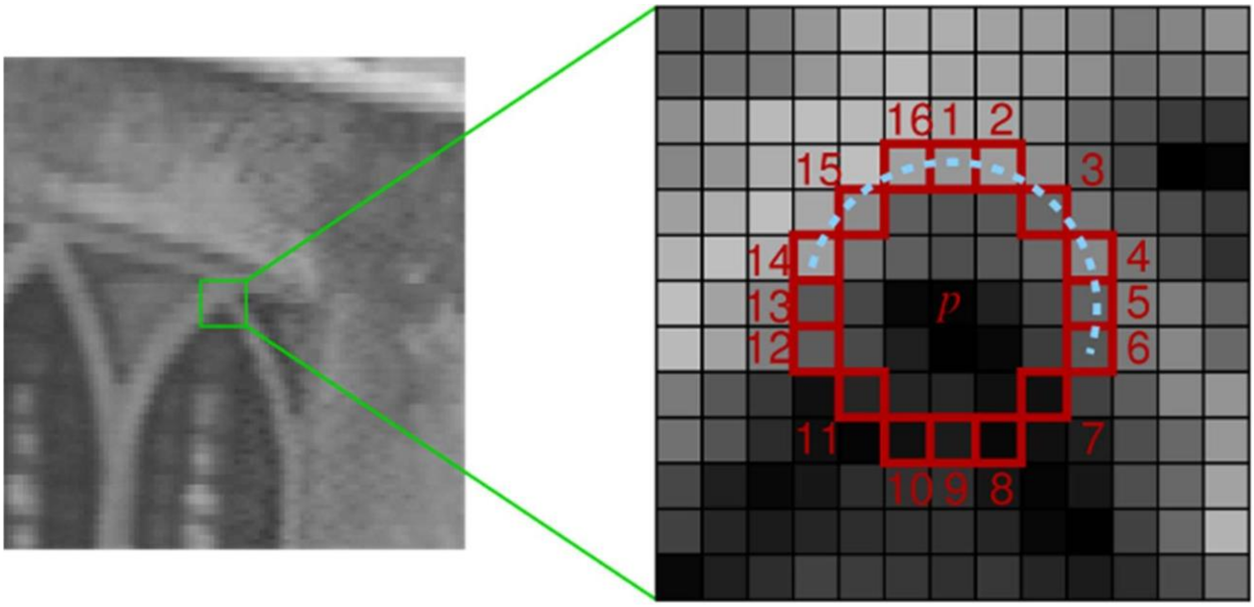

Figure 3. 7 Interest points under test and 16 pixels on the circle
(Rosten & Drummond, 2006)

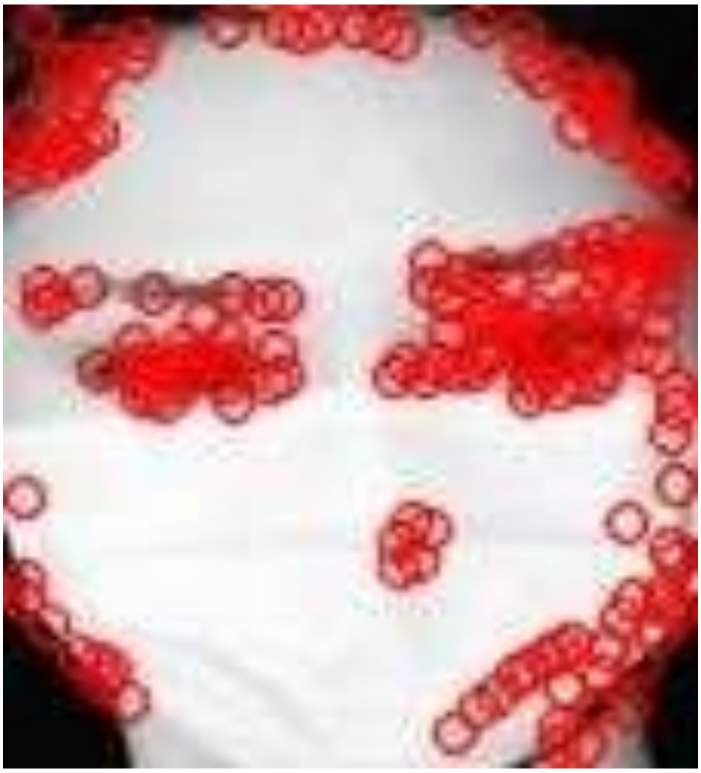

Figure 3.8 Masked face image showing corners detected by FAST

3.4.2 Binary Robust Independent Elementary Features (BRIEF)

Presented by (Calonder *et al*., 2010), BRIEF is a binary descriptor based on pair-wise intensity comparisons. A number of (128, 256, or 512) pairs are chosen randomly but on fixed locations in a patch around a keypoint. The BRIEF descriptor is computed by conducting the intensity difference test on these pairs. The test yields 1 if the intensity at one specific location is larger than at another. Otherwise, it gives 0. This descriptor is computationally faster than Scale Invariant Feature Transform (SIFT) or Speed Up Robust Feature (SURF), as it is based on binary comparisons. BRIEF is invariant to illumination changes, but not to scaling or rotation. With standard configuration, a BRIEF descriptor is 32-dimensional (Kashif *et al*., 2016).

3.4.3 Histogram of Oriented Gradient (HOG)

HOG is a feature extraction technique for object detection in image processing that is based on gradient estimation. The HOG extraction methodology counts gradient orientation occurrences in localized parts of an image discovery window (Surasak *et al*., 2018). This paper employed HOG to extract features since it is insensitive to geometric and photometric changes (Priyankha & Suresh, 2017). The following are key steps to generate HOG features: The variation in the x and y-axis for each pixel in the photo is computed after preprocessing and scaling the image, and the magnitude and orientation are estimated utilizing the methods in equations 3.1 and 3.2, respectively (Shu *et al*., 2011).

$$\text{Total Gradient Magnitude} = \overline{f(\square_s)^2 + (\square_y)^2} \qquad (3.1)$$

Where $\square_y$ is the gradient in the y-axis, and $\square_s$ is the gradient in the x axis.

$$\text{Orientation} = \square\square\square\ (\square) = \square_y \diagup \square_s \qquad (3.2)$$

The value of the angle ($\square$) is presented in equation 3.3

$$\square = \square\square\square\square\,(\square_y / \square_s) \tag{3.3}$$

The main advantage of using HOG in this study is that it captures edge and luminance structure, which is a key feature of local shape, in a regional representation with a high degree of invariance to local photometric and geometric alterations (Sreelekshmi & Mahesh, 2014). Translations make no difference if they are much narrower than the local spatial width or oriented bin dimensions (Surasak *et al*., 2018).

3.4.4 Convolutional Neural Network (CNN)

In the image processing and analysis field, the convolutional neural network has become a vastly proficient method of feature extraction and identification (Benkaddour & Bounoua, 2017). CNN is the most representative model of deep learning (Lecun *et al*., 2015). CNN is a multi-layer neural network; each layer consists of multiple 2D surfaces, and each plane consists of multiple independent neurons. CNNs have many connections, and their design is made up of many layers, such as pooling, convolution, and fully connected layers, that achieve some form of regularization (Ferreira & Giraldi, 2017). The standard model of CNN has a structure composed of the input layer, alternating convolutional layers, pooling layers and non-linear layer. The convolutional layer and the pooling layers are responsible for feature extraction, while the fully connected layers perform the classification of the features extracted by the convolutional and pooling layers(Benkaddour & Bounoua, 2017). The convolutional and pooling layers are discussed in the sub-sections below.

1. Convolutional Layers

CNN's basic building block is its convolutional layers. The main goal of convolution (Namatēvs, 2018). is to extract different features from the input. Each kernel is used to generate a feature map, and these layers are made up of a succession of filters that try to extract local features from the input. Low-level significant features such as edges, corners, textures, and lines are retrieved in the first convolutional layer. The convolutional layer after that extracts higher-order features, but the highest-level features are extracted in the final convolution layer (Hossain & Alam Sajib, 2019).

2. *Pooling Layer*

The pooling layer, also known as a subsampling layer, is used to decrease the resolution of prior feature maps by compressing features and lowering the network's computation cost (Sultana *et al.*, 2018). It frequently acts as a bridge between multiple convolutional layers (Liu, 2018). It fine-tunes the noise and disorder-resistant properties. In general, a pooling layer downsamples the input map and reduces the dimensionality of the feature maps utilised by subsequent layers (Ferreira & Giraldi, 2017). Pooling divides the inputs into R x R-sized areas, with each region producing one output. The average of a rectangle neighbourhood, max pooling, and pooling via down-sampling are among pooling algorithms used by CNN.

To produce one output from each zone, pooling splits the inputs into regions of size *N x N* . If an input of size *S x S* is fed to the pooling layer, the output size O may be calculated using equation 3.4:

$$\square = \square\square\square\square\square(\frac{S}{N}) \tag{3.4}$$

Each output map can mix convolution with numerous input maps. Which can be expressed as follows in equation 3.5 (Namatēvs, 2018):

$$\square_j^L = \square(\textstyle\sum_{iG_j} \square_j^{L-1} * \square_{ij}^L + \square_j^L) \tag{3.5}$$

Where :

L – The convolutional layer;

L−1 – the downsampling layer;

$\square_j^{L-1}$ – input features of *L* −1 convolutional layer;

$\square_{ij}^L$ – Kernel maps of *L* convolutional layer;

$\square_j^L$ – Additive bias of *L* convolutional layer;

Mj – represents a selection of input maps;

i-th – input;

j-th –output.

In general, feature extraction with CNNs is made up of several similar processes, each of which is made up of three cascading layers: the convolution layer, the activation layer, and the pooling function (Liu, 2018).

3.4.5 Raw pixel-based features

Images are represented by pixels, which means that the simplest way to create image features is to use these raw pixel values as separate features. The number of extracted features will be the same as the number of pixels in the image.

3.4.6 Binary Robust Invariant Scalable Keypoints (BRISK)

The BRISK algorithm is a feature point detection and description algorithm with scale invariance and rotation invariance (Kashif *et al.*, 2016). It constructs the feature descriptor of the local image through the grayscale relationship of random point pairs in the neighbourhood of the local image and obtains the binary feature descriptor (Liu *et al.*, 2018).

3.5 Classification

Machine learning capability lies in its ability to generalize by correctly classifying unknown information based on models developed using the training dataset (Andersson & Englund, 2016). Three machine learning classification models were used for classification in this research work: Support Vector Machine (SVM), Naïve Bayes

(NB), and K-Nearest Neighbor. Face images will be grouped into two classes in this research: No Mask (0) and Mask present (1).

3.5.1 Support Vector Machine (SVM)

The SVM algorithm is a supervised learning model (Walsh, 2019). This algorithm is based on the structural risk minimization principle, which allows it to compress an array of raw data into a support vector set and learn how to achieve a classification decision function (Cao *et al*., 2019). The SVM model iterates over a collection of labelled training samples to locate a hyper-plane that produces an optimal path cap by finding data points. The use of support vectors improves class differentiation (Ghosh, 2019). The decision function of a binary SVM in the input space is expressed in Equation 3.6.

$$\square = h(\square) = \square\square\square\square \left(\sum_{j=1}^{n} \square_j \square_j \, \square(\square, \square_j) + \square \right) \tag{3.6}$$

where x is the feature vector to be classified, j is the training instance index, n is the number of training example, and $\square_j$ is the training example label (1 or –1). $\square_j$ and v are fitted to the data to optimize the margin, and j, K (,) is the kernel function. Support vectors are training variables for which $\square_j \neq 0$ (Tang, 2013) .

3.5.2 Naïve Bayes (NB)

This is a supervised learning method and a statistical classification scheme are both demonstrated in the NB Model. It is based on an intrinsic probabilistic model and aids in measuring the results’ probabilities to obtain principled uncertainty about the model (Sopharak *et al*., 2010). The NB classifier is a probabilistic machine learning algorithm based on the Bayes theorem and the assumption of great feature independence. Learning involves numerous linear parameters in the number of problem functions, and NB classifiers are very scalable (Berrar, 2019; Harangi *et al*., 2012). The Bayes theorem provides a way to compute the posterior probability

$\square(\square|\square)$from$\square(\square)$,

$\square(\square)$ and$\square(\square|\square)$ in NB. Equation (3.7) and (3.8) presented the equation for posterior probability$\square(\square|\square)$.

$$\square(\square|\square) = \frac{P(\square|\square)\times P(s)}{P(y)} \qquad (3.7)$$

$$\square(\square|\square)=\frac{P(\square_1|\square)\times P(\square_2|\square)\times\ldots\times P(\square_n|\square)}{\times P(s)\; P(y_1,\ldots,y_n)} \qquad (3.8)$$

3.5.3 K-Nearest Neighbour (KNN)

In KNN an item is classified based on its “distance” from its neighbours, and it is allocated to the most common class of its k closest neighbours (Krishnaiah *et al*., 2015). If k = 1, the algorithm becomes the nearest neighbour algorithm, and the object is allocated to the nearest neighbour’s class. This number K indicates how many neighbours an object has (Kataria & Singh, 2013) .

The Euclidean distance is a linear distance between two points in Euclidean space (Kataria & Singh, 2013; Novakovic *et al*., 2016; Parvin *et al*., 2010). If two vectors y_i and y_j are given where yi =(y_{i1}, y_{i2}, y_{i3}, …, y_{in}) And y_j =(y_{j1}, y_{j2}, y_{j3}, …, y_{jn}), Then the Euclidean distance between y_i and y_j is given in equation (3.9):

$$\square(\square_i, \square_j = J\overline{\sum_{k=1}^{n}(\square_{ik} - \square_{jk})^2} \qquad (3.9)$$

The following is a description of the K-NN algorithm:

- Step 1: Assigns a positive integer k to each new sample.

- Step 2: In the database, select k entries that are closest to the new case.

- Step 3: The most common category is found for such entries.
- Step 4: We assign a category to the new sample.

3.6 Performance Metrics

The methodology's performance is measured using Precision, Execution Time, Recall, F-Score, and Accuracy performance evaluation metrics. Each of these metrics is very crucial to the measurement of the performance of this work.

3.6.1 Precision

Precision is a metric that measures how many correct positive forecasts have been made. That is of all the positive predictions made by the classification model, how often was the model correct. In the case of this study, when the model classified a sample as having a Face mask, how often was it correct. The proportion of accurately forecasted positive instances divided by the total number of positive instances predicted is used to compute it. The formula below considers Falsepositives as well because even though the class is negative, the classification model classified it as positive. The formula in equation 3.10 can be used to calculate the precision. It is worthy of note however that a high precision value does not necessarily mean that the model has performed well. This is because, the higher the precision the higher the chances that the classification model has left out some of the positives it should have accepted.

$$\text{Precision} = \text{Truepositives}/(\text{Truepositives} + \text{Falsepositives}) \qquad (3.10)$$

3.6.2 Recall/Sensitivity

Recall is an indicator that shows how many correct positive predictions were produced out of all possible positive predictions. Recall is considered in this study because the study focuses primarily on Faces that has a Mask. The recall is calculated using the formula in equation 3.11. The high value of recall may mean a lower value of Precision, because recall considers everything it perceives as positive prediction, this may certainly contain false positives too.

$$\text{Recall} = \text{Truepositives}/(\text{Truepositives} + \text{FalseNegatives}) \qquad (3.11)$$

3.6.3 F-score

The harmonic average of recall and precision is known as the F-measure. Equation 3.12 represents this definition numerically. This metric balances the trade between Precision and Recall. If the decision threshold is moved such that it favours Recall, then precision suffers a reduction in value, likewise when the decision threshold is adjusted to favour Precision, then recall surfers reduction in value, this F-score is meant to balance these trade offs.

$$\text{F} - \text{measure} = 2 * \frac{\text{precision} * \text{ recall}}{\text{precision+recall}} \qquad (3.12)$$

3.6.4 Accuracy

The rate of correctly classified instances can be used to define accuracy. The Accuracy of a classification model simply put indicates that of all the classification made by the classification model, how many did the model get right. Accuracy alone cannot be used to measure the performance of a classification model, especially if the data set is acutely imbalance. If the target class contains more members than the other class, the accuracy will always be 100%, which does not represent the correctly classified dataset. Equation 3.13 represents this accuracy definition numerically:

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True negative}}{\text{True Positive} + \text{True negative} + \text{False Positive} + \text{False negative}} \qquad (3.13)$$

3.6.5 Execution time

Execution time measures the time taken for the model to perform feature extraction. This a very important metric for this research work, as the system, will run in real-time and fast too. Therefore, the time it takes for the model to extract features ready for matching must be considerably small.

CHAPTER FOUR

4.0 RESULTS AND DISCUSSION

In this work, experiments were conducted on five different feature descriptor algorithms: the Raw pixel features, BRIEF, BRISK, HOG and CNN feature descriptors with respect to face mask detection and classification. These extracted features were fed to the SVM, KNN and NB classifiers for classification.

4.1 Support Vector Machine Results

The results of the different feature descriptor techniques using SVM classifier are shown in Table 4.1.

Table 4. 1 Face Mask Classification Result using SVM classifier

FEATURE EXTRACTORS	ACCURACY (%)	PRECISION (%)	RECALL (%)	F1-SCORE (%)
Raw Pixel-based	95	95	95	95
FAST + BRIEF	95.54	96.57	96.57	96.57
FAST + BRISK	91.40	99.32	84.88	91.54
FAST + HOG	99.46	99.41	98.83	99.12
CNN	99.12	100	98.32	99.15

From Table 4.1, it can be inferred that all the extracted features produced a good accuracy value. However, the FAST+HOG produced the highest classification accuracy with a value of 99.46% as compared to Raw Pixel-based, FAST+BRIEF, FAST+BRISK and CNN with a classification accuracy of 95.00%, 95.54%, 91.40% and 99.12%, respectively. Based on the precision metric, CNN produced a higher precision value of 100% than Raw Pixel-based, FAST+BRIEF, FAST+BRISK and FAST+HOG with 95%, 96.57%, 99.32% and 99.41 %, respectively. Looking at the obtained recall values, the FAST+HOG has a high recall value of 98.83%, which shows that the number of correct positive predictions made from all the positive predictions is better than the positive prediction made by Raw Pixel-based, FAST+BRIEF, FAST+BRISK and CNN with a recall of 95%, 96.57%, 84.488% and 98.32% respectively. From the recall value, it can also be seen that FAST+BRISK had the lowest recall value. Evaluating from the F1-score perspective, CNN produces the highest F1-Score of 99.15%, followed by FAST+HOG with an F1-Score of 99.12%. Figure 4.1 shows a visual representation of SVM performance on all the feature descriptors.

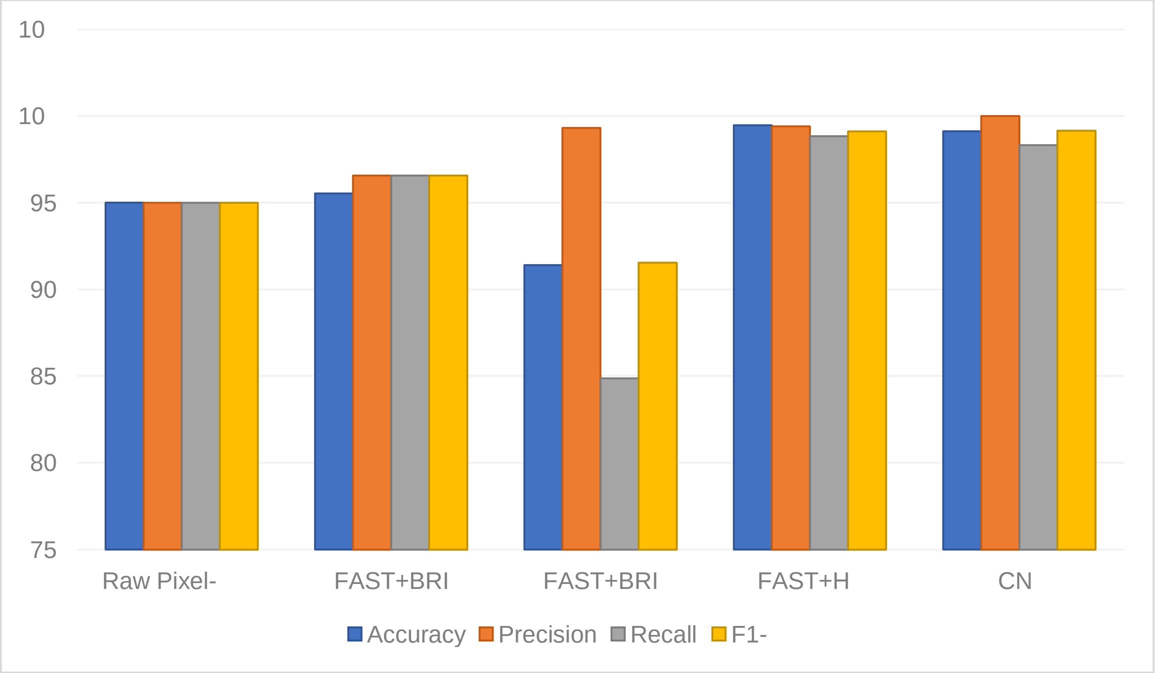

Figure 4. 1 Performance Comparison of CNN, HOG, BRISK, BRIEF and Raw Pixel for SVM classification

Figure 4.1 gives a clear visualization of the accuracy, recall, precision, and f-score for CNN, FAST + HOG, FAST + BRISK, FAST + BRIEF and Raw Pixel-based. The accuracy, precision, recall, and f1-score values represented in the chart are displayed in Table 4.1.

The confusion matrix for SVM classification using the different five feature vectors (CNN, FAST + HOG, FAST + BRISK, FAST + BRIEF AND Raw Pixel-based) are shown in figure 4.2, 4.3, 4.4, 4.5 and 4.6 respectively.

Actual: Positive (1)	1184	42
Actual: Negative (0)	42	932
n = 2200	Predicted: Positive (1)	Predicted: Negative (0)

Figure 4. 2 Confusion Matrix of BRIEF + SVM technique

Figure 4.2 shows the confusion matrix of BRIEF + SVM which consist of the true positive, true negative values, false positive and false negative values. Out of the two thousand, two hundred (2200) examples used to test the model, 1184 examples were correctly classified as face mask while 42 were wrongly classified as face mask. 932 examples were correctly classified as no face mask while 42 examples were wrongly classified as no face mask images. Figure 4.3 is a confusion matrix of the FAST + HOG + SVM technique.

Actual: Positive (1)	1191	7
Actual: Negative (0)	7	995
n = 2200	Predicted:Positive (1)	Predicted: Negative (0)

Figure 4. 3 Confusion Matrix of FAST + HOG + SVM technique

Figure 4.3 displays the confusion matrix of FAST + HOG + SVM technique which resulted in an accuracy of 99.46%. 1191 examples were correctly classified as face mask out of the 2200 examples used to test the model, while 7 were wrongly classified as face mask. 995 examples were correctly classified as no face mask while 7 examples were wrongly classified as no face mask. Figure 4.4 is a confusion matrix of the Decision tree classification.

Actual: Positive (1)	1023	182
Actual: Negative (0)	7	988
n = 2200	Predicted: Positive (1)	Predicted: Negative (0)

Figure 4. 4 Confusion Matrix of FAST + BRISK + SVM technique

Figure 4.4 displays the confusion matrix of FAST + BRISK + SVM which resulted in an accuracy of 91.40%. The confusion in figure 4.4 shows that 1023 examples were correctly classified as face mask out of the 2200 examples used to test the model, whereas 7 were wrongly classified as face mask. Furthermore, 988 examples were correctly predicted as no face mask, whereas 182 examples were wrongly classified as no face mask images. Figure 4.5 is a confusion matrix of the CNN + SVM technique.

Actual Positive (1)	1233	0
Actual Negative (0)	21	946
n = 2200	Predicted: Positive (1)	Predicted: Negative (0)

Figure 4. 5 Confusion Matrix of CNN + SVM technique

Figure 4.5 displays the confusion matrix of CNN + SVM technique. After testing the model, 1233 examples were correctly categorized as face mask, while 21 were wrongly classified as face mask. 946 examples were correctly classified as no face mask while no example was wrongly classified as no face mask images. Figure 4.6 is a confusion matrix of the Raw Pixel-based + SVM technique.

Actual Positive (1)	1117	47
Actual Negative (0)	65	971
n = 2200	Predicted Positive (1)	Predicted Negative (0)

Figure 4. 6 Confusion Matrix of Raw Pixel-based + SVM technique

The confusion matrix of the Raw Pixel-based + SVM is represented on figure 4.6. Out of the 2200 examples used to test the model, 1117 examples were correctly classified as face mask though 65 were wrongly classified as face mask. 971 examples were correctly classified as no face mask while 47 examples were wrongly classified as no face mask.

4.2 Naïve Bayes (NB)

The results of the five different feature descriptor techniques using NB classifier are shown in Table 4.2.

Table 4. 2 Face Mask Classification Result using Naive Bayes (NB)

FEATURE EXTRACTORS	**ACCURACY (%)**	**PRECISION (%)**	**RECALL (%)**	**F1-SCORE (%)**
Raw Pixel-based	91.40	92.22	92.74	92.74
FAST + BRIEF	92.35	94.54	92.51	93.51
FAST + BRISK	82.80	87.13	82.32	84.66
FAST + HOG	96.83	97.16	97.16	97.16
CNN	93.97	92.61	96.45	94.49

It is clear from Table 4.2 that each of the retrieved image features generated a satisfactory accuracy value. Nevertheless, FAST+HOG provided the highest classification accuracy with a value of 96.83% as opposed to raw pixel-based, FAST+BRIEF, FAST+BRISK, and CNN with a classification accuracy of 91.40%, 92.35%, 82.80%, and 93.97%, respectively. Raw pixel, FAST+BRIEF, FAST+BRISK, and CNN all had lower precision values than FAST+HOG (97.16% vs. 92.22%, 94.54%, 87.13%, and 92.61%, respectively). When comparing the obtained recall values, the FAST+HOG has a high recall value of 97.16%, demonstrating that the proportion of correct positive predictions made from all the positive predictions is higher than that of the raw pixel-based, FAST+BRIEF, FAST+BRISK, and CNN, with recalls of 92.74%, 92.51%, 82.32%, and 96.45%, respectively. It is also clear from the recall value that FAST+BRISK had the least recall value. In terms of evaluation using the F1-score, CNN comes in second with an F1-score of 94.99%, followed by FAST+HOG with a score of 97.16%. A visual representation of NB's performance on each of the feature descriptors is provided in Figure 4.7.

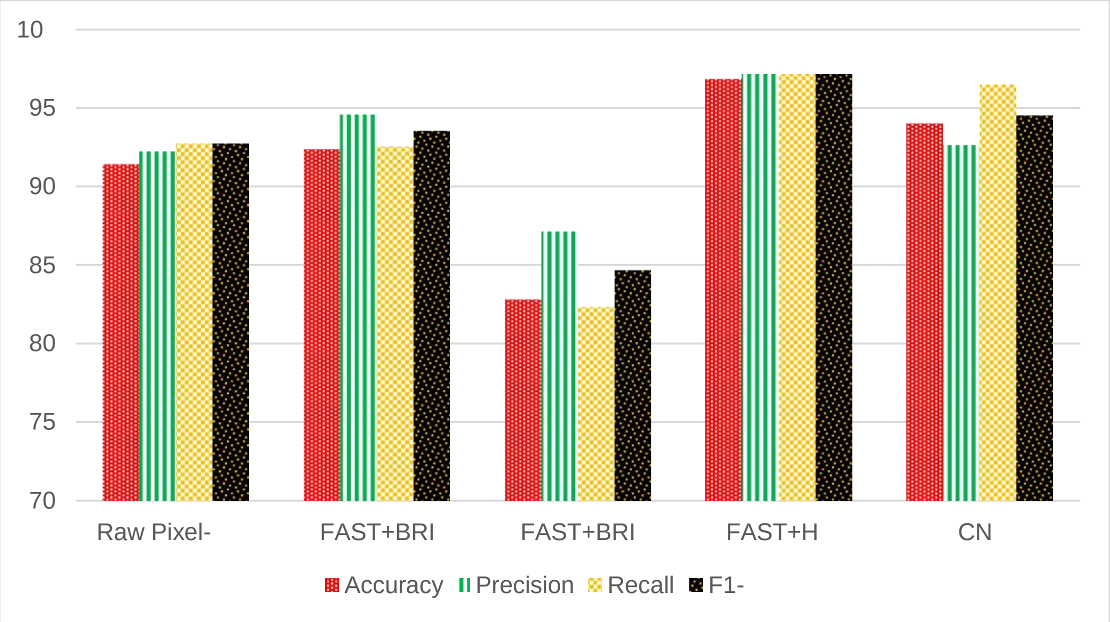

Figure 4. 7 Comparison of CNN, HOG, BRISK, BRIEF and Raw Pixel for NB classification

The accuracy, recall, precision, and f-score for CNN, FAST + HOG, FAST + BRISK, FAST + BRIEF, and raw pixel-based using NB classifier are clearly displayed in Figure 4.1. Table 4.2 results show the values for the chart's accuracy, precision, recall, and f1-score.

4.3 K-Nearest Neighour (KNN)

Table 4.3 displays the outcomes of the five various feature descriptor methods using the KNN classifier.

Table 4. 3 Face Mask Classification Result using K-Nearest Neighour (KNN)

FEATURE EXTRACTORS	**ACCURACY (%)**	**PRECISION (%)**	**RECALL (%)**	**F1-SCORE (%)**
Raw Pixel-based	92.99	97.60	90.06	93.68
FAST + BRIEF	93.94	94.80	95.04	94.92
FAST + BRISK	87.89	86.49	92.49	89.39
FAST + HOG	98.10	99.43	97.22	98.31
CNN	97.46	99.89	95.65	97.77

The accuracy values that each of the retrieved picture characteristics produced were satisfactory, as shown in Table 4.3. In contrast to raw pixel-based, FAST+BRIEF, FAST+BRISK, and CNN, which had classification accuracy values of 92.99%, 93.94%, 87.89%, and 97.46%, respectively, FAST+HOG had the best classification accuracy with a value of 98.10%. In comparison to CNN, the precision values of raw pixel, FAST+BRIEF, FAST+BRISK, and FAST+HOG were all lower (99.89% vs. 97.60%, 94.80%, 86.49%, and 99.43%, correspondingly). When recall values are compared, the FAST+HOG has a high recall value of 97.22%, indicating that the ratio of correct positive assumptions made from all the true positives is higher than that of the raw pixel-based, FAST+BRIEF, FAST+BRISK, and CNN, with recalls of 90.06%, 95.04%, 92.49%, and 95.65%, respectively. The accuracy, precision and recall values also make it evident that FAST+BRISK had the lowest performance. FAST+HOG is ranked first with a F1-score of 98.31%, followed by CNN with an F1-score of 97.77% in terms of evaluation. Figure 4.8 offers a graphical representation of KNN's performance on each of the feature descriptions.

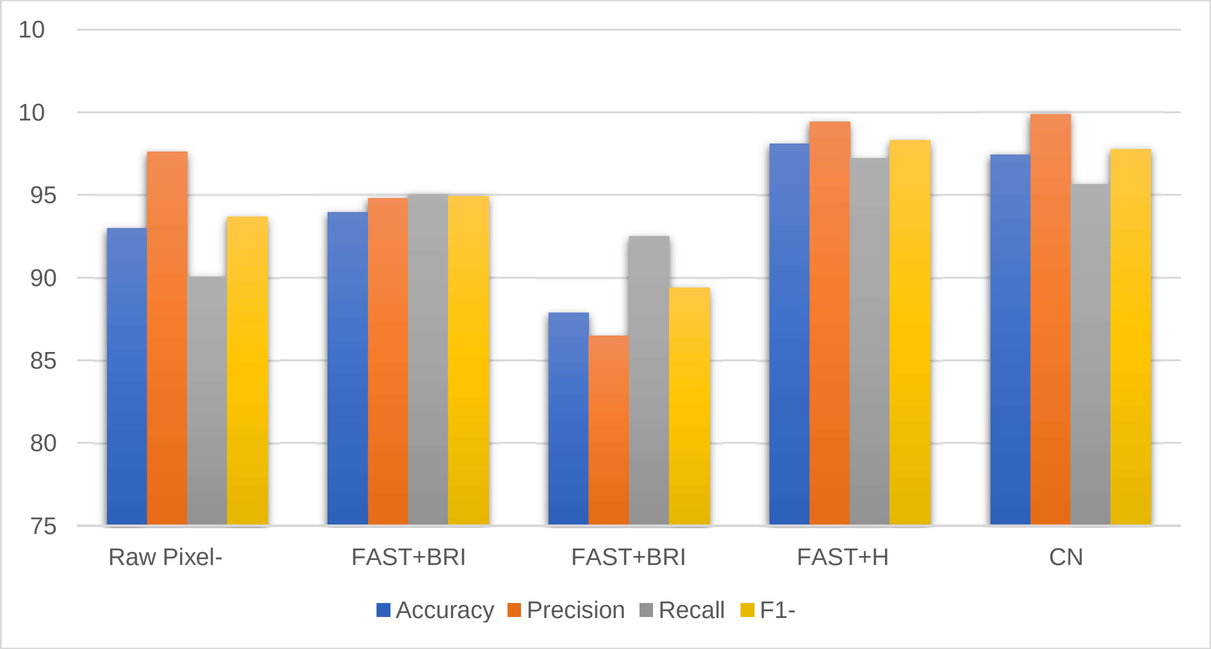

Figure 4. 8 Comparison of CNN, HOG, BRISK, BRIEF and Raw Pixel for KNN classification

Figure 4.1 clearly shows the accuracy, recall, precision, and f-score for CNN, FAST + HOG, FAST + BRISK, FAST + BRIEF, and raw pixel-based utilizing KNN classifier. The accuracy, precision, recall, and f1-score values represented in the chart are displayed in Table 4.3.

4.4 Execution Time

During the feature extraction the time taken for each of the five descriptors to extract the image features are displayed in Table 4.4.

Table 4. 4 Comparison of feature descriptors based on Execution Time

FEATURE EXTRACTORS	EXECUTION TIME (Seconds)
Raw Pixel-based	52.36
FAST + BRIEF	46.04
FAST + BRISK	313.10
FAST + HOG	48.00
CNN	162.92

Visual representation of the execution time for each of the description is shown in figure 4.9.

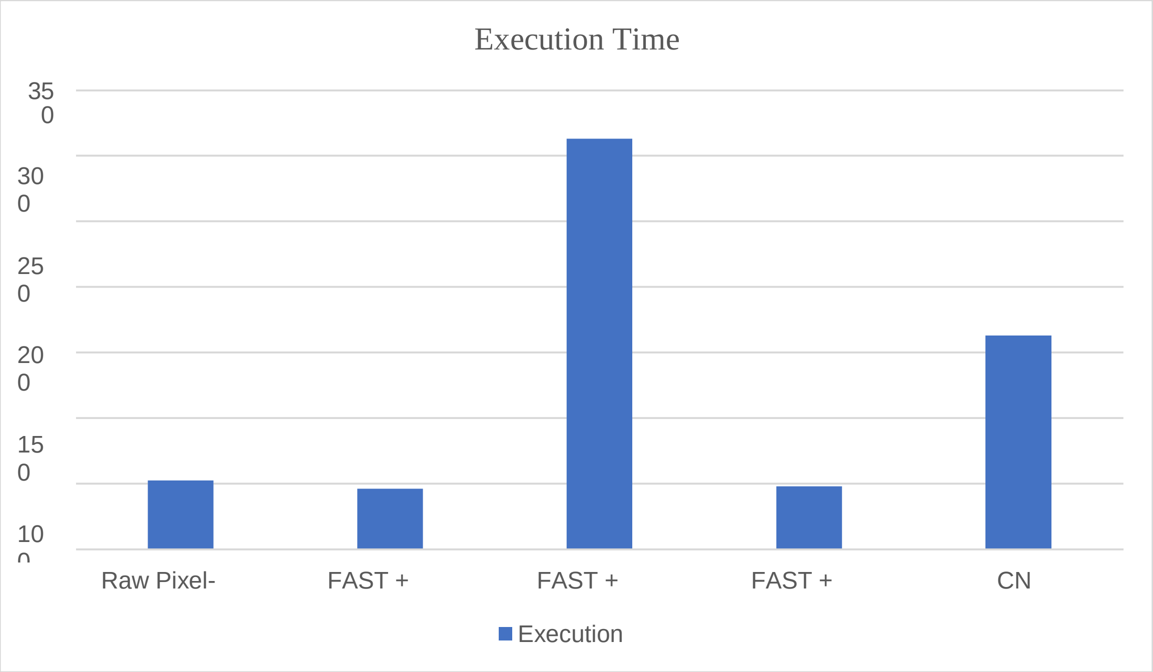

Figure 4. 9 Execution time for each descriptor

Comparing the various feature descriptors based on the execution, the FAST+BRIEF descriptor had the least execution time of 46.04 seconds, followed by FAST+HOG with 48.00 seconds. The FAST+BRISK descriptor took longer than the other descriptors when describing and extracting the image features. From the Accuracy, Recall, precision, F1-Score, and execution time obtained, it can be concluded that the FAST+HOG descriptor is more appropriate for a reliable Face mask identification than the other four descriptors.

Table 4.5 shows a comparison of the proposed method with existing related works.

Table 4. 5 Comparison with Existing Works

METHODS	**ACCURACY (%)**	**PRECISION (%)**	**RECALL (%)**	**F1-SCORE (%)**	**EXECUTION TIME (Seconds)**
Loey *et al.*, (2021a)	99.27	99.27	99.27	99.27	**-**
Wang *et al.*, (2020)	95	-	-	-	-
Method used	99.46	99.41	98.83	99.12	48.00

Based on the values of accuracy presented in Table 4.5, the proposed FAST+HOG has indicated the best performance with 99.46%. While the method proposed by Loey *et al*. (2021a) and (Wang *et al*. (2020) had 99.27% and 95% accuracy, respectively.

CHAPTER FIVE

5.0 CONCLUSION AND RECOMMENDATION

5.1 Conclusion

In this research five different feature descriptor algorithms: the Raw pixel features, BRIEF, BRISK, HOG and CNN feature descriptors were used to extract features from the face images. For speedy classification the FAST corner detector was used to detect the interest point in the images and the features of these interest points were extracted using the BRISK, HOG and BRIEF. The extracted features using the Raw pixel features, BRIEF, BRISK, HOG and CNN feature descriptors were fed as input to the SVM, NB, and KNN classification models. To train the SVM, KNN and NB models to identify if an image contains face mask or just bare face the RMFD real world masked dataset was used. The main goal of this research is to develop a method that will detect face masks in face images in an accurate and timely manner and this was achieved with the FAST+HOG+SVM method.

In conclusion, a method for face mask detection was developed based on a cascade of FAST and HOG feature extraction techniques. In this research work, prediction using the FAST + HOG + SVM model achieved a better performance than classification using any of the other models like FAST + BRIEF + NB, raw pixel + KNN, CNN + KNN, FAST + BRISK + SVM, CNN + SVM, CNN + NB and Raw pixel + SVM.

The first objective of processing the data obtained from the online repository was accomplished by under sampling of the majority class. The challenges of execution time and computational complexity was overcome using two feature extraction models; FAST and HOG for detecting and describing the important image features. The extracted features from the cascade of the FAST and HOG features were fed to the SVM, NB and KNN for classification.

The cascading of the FAST and HOG model is an accomplishment of the second objective. The FAST + HOG model was evaluated against FAST + BRIEF, FAST + BRISK, CNN and Raw pixel models and against related works on face mask detection. In accomplishment of the third and fourth objectives of this work, the accuracy, precision, recall, f-score and execution time performance measurements were used to conduct this evaluation. The FAST+HOG technique obtained an accuracy of 99.46%, a precision of 99.41%, a recall of 98.83%, f-score of 99.12% for SVM classification and an execution time of 48 seconds. These findings show that the FAST+HOG technique is beneficial in improving real-time face mask detection. This accomplishes the aim of this work which to reduce to further spread of the deadly Covid-19 virus.

5.2 Recommendation

In this research, the SVM, KNN and NB classification models were used. Hence for future work, more classification models such as Decision tree, and discriminate analysis can be used or combined to improve the system's robustness. In this work, only the Real-world Masked Face Dataset (RMFD) was considered for analysis given that it consists of real face mask images, not synthetics. The dataset suffers from the problem of imbalanced data. For future work, the number of datasets used can be increased, and more face mask images can be added to the RMFD to reduce the imbalanced nature of the dataset. In addition, most face mask datasets consist of synthetic masks, hence it is recommended that more real face mask datasets should be created as it represents the real-world scenario properly.

5.3 Contribution to Knowledge

This study contributed to knowledge by developing of a cascaded bi-level model with feature detection and feature description algorithms for categorizing face photos into no face mask and face mask categories.

REFERENCES

Alguzo, A., Alzu'Bi, A., & Albalas, F. (2021). Masked Face Detection using Multi-Graph Convolutional Networks. *2021 12th International Conference on Information and Communication Systems, ICICS 2021*, 385–391. https://doi.org9/

Alzu'bi, A., Albalas, F., Al-Hadhrami, T., Younis, L. B., & Bashayreh, A. (2021). Masked face recognition using deep learning: A review. *MDPI Open Access Journals* (Vol. 10, Issue 21). *MDPI.* https://doi.org/10.3390/electronics10212666

Andersson, E., & Englund, R. (2016). Machine Learning for Technical Information Quality Assessment. *Open Access Theses and Dissertations*. https://publi34989.pdf

Attallah, B., Serir, A., Chahir, Y., & Boudjelal, A. (2017). Histogram of gradient and binarized statistical image features of wavelet subband-based palmprint features extraction. *Journal of Electronic Imaging, 26*(06), 1. https://doi.org/10.1117/1.jei.

Azamgarhi MPharm, T., Maynard-Smith FRCPath, L., Bysouth, H. R., & Warren FRCPath, S. (2022). Coronavirus disease 2019 (COVID-19) polymerase chain reaction (PCR) screening of asymptomatic healthcare workers in a low-prevalence setting: A single-center UK cohort study. *Antimicrobial Stewardship & Healthcare Epidemiology*, *2*(1), e84. https://doi.org/10.1017/ASH.2022.44

Baker, J. O., Martí, G., Braunstein, R., Whitehead, A. L., & Yukich, G. (2020). Religion in the Age of Social Distancing: How COVID-19 Presents New Directions for Research. *Sociology of Religion*, *81*(4), 357–370. https://doi.org/CREL/SRAA039

Benedict, S. R., & Kumar, J. S. (2017). Geometric shaped facial feature extraction for face recognition. *2016 IEEE International Conference on Advances in Computer Applications, ICACA 2016*, 275–278. https://doi.org/0.1109/ICACA.2016.7887965

Benkaddour, M. K., & Bounoua, A. (2017). Feature extraction and classification using deep convolutional neural networks, PCA and SVC for face recognition. *Traitement Du Signal*, *34*(1–2), 77–91. https://doi.org/10.3166/TS.34.77-91

Bernheim, A., Mei, X., Huang, M., Yang, Y., Fayad, Z. A., Zhang, N., Diao, K., Lin, B., Zhu, X., Li, K., Li, S., Shan, H., Jacobi, A., & Chung, M. (2020). Chest CT Findings in Coronavirus Disease-19 (COVID-19): Relationship to Duration of Infection. *Radiology*, *295*(3), 685–691. https://doi.org/.1148/RADIOL.2020200463

Berrar, D. (2019). Bayes ' Theorem and Naive Bayes Classifier Bayes ' Theorem and Naive Bayes Classifier. *Encyclopedia of Bioinfor- Matics and Computational Biology, January 2018*, 0–18. https://doi.org/10.1016/B978-0-12-809633-8.204731

Blundell, R., Costa Dias, M., Joyce, R., & Xu, X. (2020). COVID-19 and Inequalities. *Fiscal Studies - The Journal of Applied Public Economics*, *41*(2), 291–319. https://doi.org/10.1111/1475-5890.12232

Boulos, M. M. (2021). Facial Recognition and Face Mask Detection Using Machine Learning Techniques. *Montclair State University*, *728*. https://digitalcolair.edu/cgi/

Bu, W., Xiao, J., Zhou, C., Yang, M., & Peng, C. (2017). A cascade framework for masked face detection. *2017 IEEE International Conference on Cybernetics and Intelligent Systems, CIS 2017 and IEEE Conference on Robotics, Automation and Mechatronics, RAM 2017 - Proceedings*, *2018-January*, 458–462. https://doi.org/9

Bulat, A., & Tzimiropoulos, G. (2017). How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks). *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 1021-1030.* www.adrianbulat.com/face-alignment/

Cabani, A., Hammoudi, K., Benhabiles, H., & Melkemi, M. (2021). MaskedFace-Net – A dataset of correctly/incorrectly masked face images in the context of COVID-19. *Smart Health*, *19*. https://doi.org/10.1016/j.smhl.2020.100144

Cai, X. F., Chen, J., Hu, J. li, Long, Q. X., Deng, H. J., Liu, P., Fan, K., Liao, P., Liu, B. Z., Wu, G. C., Chen, Y. K., Li, Z. J., Wang, K., Zhang, X. L., Tian, W. G., Xiang, J. L., Du, H. X., Wang, J., Hu, Y., … Wang, D. Q. (2020). A Peptide-Based Magnetic Chemiluminescence Enzyme Immunoassay for Serological Diagnosis of Coronavirus Disease 2019. *The Journal of Infectious Diseases*, *222*(2), 189–195. https://doi.org/10.1093/INFDIS/JIAA243

Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features. *Springer*. https://link.springer.com/content/pdf/-3-642-15561-1_56.pdf

Cao, J., Wang, M., Li, Y., & Zhang, Q. (2019). Improved support vector machine classification algorithm based on adaptive feature weight updating in the Hadoop cluster environment. *PLoS ONE*, *14*(4). https://doi.org/.1371/journal.pone.0215136

Cao, X., Wei, Y., Wen, F., & Sun, J. (2013). Face Alignment by Explicit Shape Regression. *International Journal of Computer Vision 2013 107:2*, *107*(2), 177–190. https://doi.org/10.1007/S11263-013-0667-3

Centers for Disease Control and Prevention. (2020). *Guidance for SARS-CoV-2 Rapid Testing Performed in Point-of-Care Settings | CDC*. https://www.cdc.gov/coronavirus.html

Centers for Disease Control and Prevention. (2021a). *Nucleic Acid Amplification Tests (NAATs)*. CDC. https://www.cdc.gov/coronavirus/2019-ncov/lab/naats.html

Centers for Disease Control and Prevention. (2021b). *Requirement for Face Masks on Public Transportation Conveyances and at Transportation Hubs* . https://www.cdc.gov/quarantine/masks/face-masks-public-transportation.html?CDC_AA_refVal=https%3A%2F%2Fwww.cdc.gov%2Fcoronavirus%2F2019-ncov%2Ftravelers%2Fface-masks-public-transportation.html

Chavda, A., Dsouza, J., Badgujar, S., & Damani, A. (2021). Multi-Stage CNN Architecture for Face Mask Detection. *2021 6th International Conference for Convergence in Technology, I2CT 2021*. https://doi.org//I2CT51068.2021.9418207

Cleveland Clinic. (2021). PCR Test for COVID-19: What It Is, How Its Done, What The Results Mean. https://my.clevelandclinic.org/health/diagnostics/21462-covid-19-and-pcr-testing

Dagnes, N., Vezzetti, E., Marcolin, F., & Tornincasa, S. (2018). Occlusion detection and restoration techniques for 3D face recognition: a literature review. *Machine Vision and Applications 2018 29:5*, *29*(5), 789–813. https://doi.org/10.1007/S00138-018-0933-Z

Deidre, G. S., Ned, L., & Bysiewicz, S. (2020). COVID-19 Testing: Recommendations for the Use of Nucleic Acid Tests to Diagnose Persons with Current Infection with SARS-CoV-2. *Virus* . www.ct.gov/dph

Deng, J., Guo, J., Ververas, E., Kotsia, I., & Zafeiriou, S. (2020). Retinaface: Single-shot multi-level face localisation in the wild. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 5202–5211. https://doi.org/10.1109/CVPR42600.2020.00525

Deshpande, N. T., & Ravishankar, S. (2016). Face Detection and Recognition using Viola-Jones algorithm and fusion of LDA and ANN. *IOSR Journal of Computer Engineering*, *18*(6), 1–6. https://pdfs.semanticscholar.org1d016adb4cba20ce4e.pdf

Deshpande, N. T., & Ravishankar, S. (2017). Face Detection and Recognition using Viola-Jones algorithm and Fusion of PCA and ANN. *Advances in Computational Sciences and Technology*, *10*(5), 1173–1189. http://www.ripublication.com

Dey, S. K., Howlader, A., & Deb, C. (2021). Mobilenet mask: A multi-phase face mask detection model to prevent person-to-person transmission of sars-cov-2. *Advances in Intelligent Systems and Computing*, *1309*, 603–613. https://doi.org/10.1007/978-981-33-4673-4_49

Ejaz Sabbir, & Islam Rabiul. (2019). Masked Face Recognition Using Convolutional Neural Network. *International Conference on Sustainable Technologies for Industry 4.0 (STI).*

Fan, X., & Jiang, M. (2021). RetinaFaceMask: A Single Stage Face Mask Detector for Assisting Control of the COVID-19 Pandemic. *IEEE*. http://arxiv.org/abs/05.03950

Fan, X., Jiang, M., & Yan, H. (2021). A Deep Learning Based Light-Weight Face Mask Detector with Residual Context Attention and Gaussian Heatmap to Fight against COVID-19. *IEEE Access*, *9*, 96964–96974. https://doi.org/ACCESS.2021.3095191

Fang, Y., Zhang, H., Xie, J., Lin, M., Ying, L., Pang, P., & Ji, W. (2020). Sensitivity of chest CT for COVID-19: Comparison to RT-PCR. *National Library of Medicine* (Vol. 296, Issue 2, pp. E115–E117). *Radiological Society of North America Inc.* https://doi.org/10.1148/radiol.2020200432

Feng, W., Newbigging, A. M., Le, C., Pang, B., Peng, H., Cao, Y., Wu, J., Abbas, G., Song, J., Wang, D. B., Cui, M., Tao, J., Tyrrell, D. L., Zhang, X. E., Zhang, H., &

Le, X. C. (2020). Molecular Diagnosis of COVID-19: Challenges and Research Needs. *Analytical Chemistry*, *92*(15), 10196–10209. https://doi.org/HEM.0C02060

Ferreira, A., & Giraldi, G. (2017). Convolutional Neural Network approaches to granite tiles classification. *Expert Systems with Applications*, *84,* 1–11. https://doi.org/10.1016/j.eswa.2017.04.053

Floriano, I., Silvinato, A., Bernardo, W. M., Reis, J. C., & Soledade, G. (2021). Accuracy of the polymerase chain reaction (PCR) test in the diagnosis of acute respiratory syndrome due to coronavirus: A systematic review and meta-analysis. *Revista Da Associacao Medica Brasileira*, *66*(7), 880–888. https://doi.org/10.1590/1806-9282.66.7.880

Foudeh, A. M., Fatanat Didar, T., Veres, T., & Tabrizian, M. (2012). Microfluidic designs and techniques using lab-on-a-chip devices for pathogen detection for point-of-care diagnostics. *Lab on a Chip, 12*(18), 3249–3266. https://doi.org/.1039/C2LC40630F

Garg, M., Prabhakar, N., Bhalla, A., Irodi, A., Sehgal, I., Debi, U., Suri, V., Agarwal, R., Yaddanapudi, L., Puri, G., & Sandhu, M. (2021). Computed tomography chest in COVID-19: When & why? *The Indian Journal of Medical Research*, *153*(1 & 2), 86–92. https://doi.org/10.4103/IJMR.IJMR_3669_20

Ghahremani, M., Liu, Y., & Tiddeman, B. (2020). FFD: Fast Feature Detector. *IEEE Transactions on Image Processing*, *10*(10). https://doi.org/1109/TIP.2020.3042057

Ghosh, S. (2019). A Study on Support Vector Machine based Linear and Non-Linear Pattern Classification. *Iciss*, 24–28. https://doi.org/10.1007/999-3-000-58529-7

Guo, J., Zhu, X., Yang, Y., Yang, F., Lei, Z., & Li, S. Z. (2020). Towards Fast, Accurate and Stable 3D Dense Face Alignment. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12364 LNCS*, 152–168. https://doi.org/10.1007/978-3-030-58529-7_10

Harangi, B., Antal, B., & Hajdu, A. (2012). Automatic exudate detection with improved naïve-Bayes classifier. *Proceedings - IEEE Symposium on Computer-Based Medical Systems*. https://doi.org/10.1109/CBMS.2012.6266341

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-December*, 770–778. https://doi.org/2016.90

Hossain, Md. A., & Alam Sajib, Md. S. (2019). Classification of Image using Convolutional Neural Network (CNN). *Global Journal of Computer Science and Technology*, *19*(2), 13–18. https://doi.org/10.34257/gjcstdvol19is2pg13

Hwang, S. S., Lim, J., Yu, Z., Kong, P., Sefik, E., Xu, H., Harman, C. C. D., Kim, L. K., Lee, G. R., Li, H. B., & Flavell, R. A. (2020). Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. *Science (New York, N.Y.), 367*(6483), 1255–1260. https://doi.org/10.1126/SCIENCE.ABB2507

Iftikhar, F., & Mohammed, J. (2011). Algorithm for Image Processing Using Improved Elimination of Gaussian Noise from FPGA Based Co-Processors IJRES Journal An Improved Median Filter Based on Efficient Noise Detection for High Quality Image Restoration. *International Journal of Soft Computing and Engineering (IJSCE)* (Issue 1). https://ijres.org/papers/v1-i2/D122733.pdf

Jassim, S., & Asaad, A. (2018). Automatic detection of image morphing by topology-based analysis. *European Signal Processing Conference*, *2018-Septe*, 1007–1011. https://doi.org/10.23919/EUSIPCO.2018.8553317

Jiang, X., Gao, T., Zhu, Z., & Zhao, Y. (2021). Real-time face mask detection method based on yolov3. *Electronics (Switzerland), 10*(7). https://doi.org/tronics10070837

Joshi, A. S., Joshi, S. S., Kanahasabai, G., Kapil, R., & Gupta, S. (2020). Deep Learning Framework to Detect Face Masks from Video Footage. *IEEE explore* https://doi.org/10.1109/CICN49253.2020.9242625

Karami, E., Prasad, S., & Shehata, M. (2017). Image Matching Using SIFT, SURF, BRIEF and ORB*:* Performance Comparison for Distorted Images. *ArXiv*. http://arxiv.org/0.02726

Kashif, M., Deserno, T. M., Haak, D., & Jonas, S. (2016). Feature description with SIFT, SURF, BRIEF, BRISK, or FREAK? A general question answered for bone age assessment. *Computers in Biology and Medicine*, *68*, 67–75. https://doi.org/10.1016/j.compbiomed.2015.11.006

Kataria, A., & Singh, M. D. (2013). A Review of Data Classification Using K-Nearest Neighbour Algorithm. *International Journal of Emerging Technology and Advanced Engineering*, *3*(6), 354–360. https://www.researchgate.net/

Kenneth, O. M., Bashir, S. A., Abisoye, O. A., & Mohammed, A. D. (2021). Face Morphing Attack Detection in the Presence of Post-processed Image Sources Using Neighborhood Component Analysis and Decision Tree Classifier. *Communications in Computer and Information Science, 1350*, 340–354. https://doi.org/10.1007/978-3-030-69143-1_27

Khandelwal, P., Khandelwal, A., Agarwal, S., Thomas, D., Xavier, N., & Raghuraman, A. (2020). Using Computer Vision to enhance Safety of Workforce in Manufacturing in a Post COVID World. *ArXiv*. https://doi.org/10.48550/arxiv.1412.69990

Kingma, D. P., & Ba, J. L. (2014). Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. https://doi.org/10.48550/arxiv.1412.6980

Kobayashi, Y., & Mitsudomi, T. (2013). Management of ground-glass opacities: should all pulmonary lesions with ground-glass opacity be surgically resected? *Translational Lung Cancer Research*, *2*(5), 354–363. https://doi.org//J.ISSN.2218-6751.2013.09.03

Krishnaiah, V., Narsimha, G., & Chandra, S. N. (2015). Heart Disease Prediction System Using Data Mining Technique by Fuzzy K-NN Approach. *Advances in Intelligent Systems and Computing*, *1*(1), 371–384. https://doi.org/10.1007/978-3-319-137285

Kulkarni, A. v, Jagtap, J. S., & Harpale, V. K. (2013). Object recognition with ORB and its Implementation on FPGA. *International Journal of Advanced Computer Research*, *3*. https://doi.org/12.4490/2217114

Kumar, T. A., Rajmohan, R., Pavithra, M., Ajagbe, S. A., Hodhod, R., & Gaber, T. (2022). Automatic Face Mask Detection System in Public Transportation in Smart Cities Using IoT and Deep Learning. *MDPI Open Access Journals*, *11*(6). https://doi.org/10.3390/electronics11060904

Larxel. (2021). Face Mask Detection. *Kaggle*. https://www.kaggle.com/

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lee, E. Y. P., Ng, M. Y., & Khong, P. L. (2020). COVID-19 pneumonia: what has CT taught us? *The Lancet. Infectious Diseases*, *20*(4), 384–385. https://doi.org/10.1016/S1473-3099(20)30134-1

Li, C., Ge, S., Zhang, D., & Li, J. (2020). Look through Masks: Towards Masked Face Recognition with De-Occlusion Distillation. *MM 2020 - Proceedings of the 28th ACM International Conference on Multimedia*, 3016–3024. https://doi.org/413960

Li, D., Chen, X., Zhang, Z., & Huang, K. (2017). Learning deep context-Aware features over body and latent parts for person re-identification. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-January*, 7398–7407. https://doi.org/10.1109/CVPR.2017.782

Li, W., Zhu, X., & Gong, S. (2018). Harmonious Attention Network for Person Re-identification. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2285–2294. https://doi.org/2018.00243

Lim, Y. X., Ng, Y. L., Tam, J. P., & Liu, D. X. (2016). Human Coronaviruses: A Review of Virus–Host Interactions. *Diseases*, *4*(3), 26. https://doi.org//DISEASES4030026

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-January*, 936–944. https://doi.org/10.1109/CVPR.2017.106

Liu, Y., Gayle, A. A., Wilder-Smith, A., & Rocklöv, J. (2020). The reproductive number of COVID-19 is higher compared to SARS coronavirus. *Journal of Travel Medicine*, *27*(2). https://doi.org/10.1093/JTM/TAAA021

Liu, Y. H. (2018). Feature Extraction and Image Recognition with Convolutional Neural Networks. *Journal of Physics: Conference Series*, *1087*(6). https://doi.org/88/1742-6596/1087/6/062032

Liu, Y., Zhang, H., Guo, H., & Xiong, N. N. (2018). A FAST-BRISK feature detector with depth information. *MDPI Open Access Journals*, *18*(11). https://doi.org/13908

Loey, M., Manogaran, G., Taha, M. H. N., & Khalifa, N. E. M. (2021a). A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. *Measurement: Journal of the International Measurement Confederation*, *167*. https://doi.org/016/j.measurement.2020.108288

Loey, M., Manogaran, G., Taha, M. H. N., & Khalifa, N. E. M. (2021b). Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection. *Sustainable Cities and Society*, *65*. https://doi.org/10.1016/J.SCS.2020.102600

Lu, R., Zhao, X., Li, J., Niu, P., Yang, B., Wu, H., Wang, W., Song, H., Huang, B., Zhu, N., Bi, Y., Ma, X., Zhan, F., Wang, L., Hu, T., Zhou, H., Hu, Z., Zhou, W., Zhao, L., … Tan, W. (2020). Genomic characterisation and epidemiology of 2019 novel coronavirus: implications for virus origins and receptor binding. *Lancet (London, England)*, *395*(10224), 565–574. https://doi.org/10.1016/S0140-6736(20)30251-8

Lv, H., Wu, N. C., Tsang, O. T. Y., Yuan, M., Perera, R. A. P. M., Leung, W. S., So, R. T. Y., Chan, J. M. C., Yip, G. K., Chik, T. S. H., Wang, Y., Choi, C. Y. C., Lin, Y., Ng, W. W., Zhao, J., Poon, L. L. M., Peiris, J. S. M., Wilson, I. A., & Mok, C. K. P. (2020). Cross-reactive Antibody Response between SARS-CoV-2 and SARS-CoV Infections. *Cell Reports*, *31*(9). https://doi.org/10.1016/j.celrep.2020.107725

McKiernan, H. E., & Danielson, P. B. (2017). Molecular Diagnostic Applications in Forensic Science. In Molecular Diagnostics: Third Edition (pp. 371–394). *Elsevier Inc*. https://doi.org/10.1016/B978-0-12-802971-8.00021-3

Militante, S. v., & Dionisio, N. v. (2020). Real-Time Facemask Recognition with Alarm System using Deep Learning. *2020 11th IEEE Control and System Graduate Research Colloquium, ICSGRC 2020 - Proceedings*, 106–110. https://doi.org/2610

Msigwa, C., Baek, S., Bernard, D., & Yun, J. (2022). Are You Wearing a Mask? Detecting If a Person Wears a Mask Using a Wristband. *Sensors (Basel, Switzerland)*, *22*(5). https://doi.org/10.3390/S22051745

Mubarak, N., & Zin, C. S. (2020). Religious tourism and mass religious gatherings — The potential link in the spread of COVID-19. Current perspective and future implications. *Travel Medicine and Infectious Disease*, *36*, 101786. https://doi.org/10.1016/J.TMAID.2020.101786

Namatēvs, I. (2018). Deep Convolutional Neural Networks: Structure, Feature Extraction and Training. *Information Technology and Management Science*, *20*(1), 40–47. https://doi.org/10.1515/itms-2017-0007

Nanni, L., Lumini, A., & Brahnam, S. (2010). Local binary patterns variants as texture descriptors for medical image analysis. *Artificial Intelligence in Medicine*, *49*(2), 117–125. https://doi.org/10.1016/j.artmed.2010.02.006

National Human Genome Research Institute. (2022). Understanding COVID-19 PCR Testing. https://www.genome.gov/about-genomics/fact-sheets/Understanding-COVID-19-PCR-Testing

Neuman, B. W., Kiss, G., Kunding, A. H., Bhella, D., Baksh, M. F., Connelly, S., Droese, B., Klaus, J. P., Makino, S., Sawicki, S. G., Siddell, S. G., Stamou, D. G., Wilson, I. A., Kuhn, P., & Buchmeier, M. J. (2011). A structural analysis of M protein in coronavirus assembly and morphology. *Journal of Structural Biology*, *174*(1), 11–22. https://doi.org/10.1016/J.JSB.2010.11.021

Nieto-Rodríguez, A., Mucientes, M., & Brea, V. M. (2015). System for Medical Mask Detection in the Operating Room Through Facial Attributes. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9117*, 138–145. https://doi.org/10.1007/978-3-319-19390-8_16

Novakovic, J. Đ., Veljovic, A., & Ilic, S. S. (2016). Experimental Study of using the K-Nearest Neighbour Classifier with Filter Methods. *Computer Science and Technologies*, *451*, 90–99. https://www.researchgate.net/publication/324918782

Oumina, A., el Makhfi, N., & Hamdi, M. (2020, December 2). Control the COVID-19 Pandemic: Face Mask Detection Using Transfer Learning. *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science, ICECOCS 2020*. https://doi.org/10.1109/ICECOCS50124.2020.9314511

Oyallon, E., & Rabin, J. (2015). An Analysis of the SURF Method. *Image Processing On Line*, *5*(2004), 176–218. https://doi.org/10.5201/ipol.2015.69

Pan, F., Ye, T., Sun, P., Gui, S., Liang, B., Li, L., Zheng, D., Wang, J., Hesketh, R. L., Yang, L., & Zheng, C. (2020). Time Course of Lung Changes at Chest CT during Recovery from Coronavirus Disease 2019 (COVID-19). *Radiology*, *295*(3), 715–721. https://doi.org/10.1148/RADIOL.2020200370

Parvin, H., Alizadeh, H., & Minati, B. (2010). A Modification on K-Nearest Neighbor Classifier. *Global Journal of Computer Science and Technology*, *10*(14), 37–41. https://globaljournals.org/GJCST_Volume10/7-A-Modification-on-K-Nearest-Neighbor-Classifier.pdf

Priyankha, J. J., & Suresh, K. (2017). Crop Disease Identification Using a Feature Extraction HOG Algorithm. *Asian Journal of Applied Science and Technology (AJAST)*, *1*(3), 35–39. https://doi.org/16.6690/P4457663-20185236

Qin, B., & Li, D. (2020). Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19. *Sensors (Switzerland)*, *20*(18), 1–23. https://doi.org/10.3390/S20185236

Raghavendra, R., Venkatesh, S., Raja, K., & Busch, C. (2020). Detecting Face Morphing Attacks with Collaborative Representation of Steerable Features. *Proceedings of 3rd International Conference on Computer Vision and Image Processing*, 255–265. https://doi.org/10.1007/978-981-32-9088-4_22

Ramesh, N., Siddaiah, A., & Joseph, B. (2020). Tackling Corona Virus Disease 2019 (COVID 19) in Workplaces. *Indian Journal of Occupational and Environmental Medicine*, *24*(1), 16. https://doi.org/10.4103/IJOEM.IJOEM_49_20

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-January*, 6517–6525. https://doi.org/10.1109/CVPR.2017.690

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., & Savarese, S. (2019). Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression. *ARXIV*. http://arxiv.org/abs/1902.09630

Rosten, E., & Drummond, T. (2006). Machine Learning for High-Speed Corner Detection. *European Conference on Computer Vision*, 430–443. https://doi.org/10.1109/ICCV.0.778954

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. *Proceedings of the IEEE International Conference on Computer Vision*, *May*, 2564–2571. https://doi.org/10.1109/ICCV.2011.126544

Sanjaya, S. A., & Rakhmawan, S. A. (2020). Face Mask Detection Using MobileNetV2 in the Era of COVID-19 Pandemic. *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy, ICDABI 2020*. https://doi.org/10.1109/ICDABI51230.2020.9325631

Saponara, S., Elhanashi, A., & Gagliardi, A. (2021). Implementing a real-time, AI-based, people detection and social distancing measuring system for Covid-19. *Journal of Real-Time Image Processing*, *18*(6), 1937–1947. https://doi.org/10.1007/S11554-021-01070-6

Saravanan, C. (2010). Color image to grayscale image conversion. *2010 2nd International Conference on Computer Engineering and Applications, ICCEA 2010*, *2*(April 2010), 196–199. https://doi.org/10.1109/ICCEA.2010.192

Satya, B. V., & Saravanan, C. (2016). Analysis of SIFT and SURF Feature Extraction in Palmprint Verification System. *2016 IEEE International Conference on computing, communication and control technology, luknov, India, 298-301.* . https://www.researchgate.net/publication/310240326_Analysis_of_SIFT_and_SURF_Feature_Extraction_in_Palmprint_Verification_System

Scherhag, U., Rathgeb, C., Merkle, J., Breithaupt, R., & Busch, C. (2019). Face Recognition Systems under Morphing Attacks: A Survey. *IEEE Access*, *7*, 23012–23026. https://doi.org/10.1109/ACCESS.2019.2899367

Sexton, N. R., Smith, E. C., Blanc, H., Vignuzzi, M., Peersen, O. B., & Denison, M. R. (2016). Homology-Based Identification of a Mutation in the Coronavirus RNA-Dependent RNA Polymerase That Confers Resistance to Multiple Mutagens. *Journal of Virology*, *90*(16), 7415–7428. https://doi.org/10.1128/JVI.00080-16

Shahid, F., Zameer, A., & Muneeb, M. (2020). Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM. *Chaos, Solitons and Fractals*, *140*, 110212. https://doi.org/10.1016/j.chaos.2020.110212

Shi, Y., Wang, G., Cai, X. peng, Deng, J. wen, Zheng, L., Zhu, H. hong, Zheng, M., Yang, B., & Chen, Z. (2020). An overview of COVID-19. In *Journal of Zhejiang University: Science B* (Vol. 21, Issue 5, pp. 343–360). *Zhejiang University Press*. https://doi.org/10.1631/jzus.B2000083

Shretash, W. (2020). *Medical Mask Dataset | Kaggle*. Kaggle. https://www.kaggle.com/datasets/shreyashwaghe/medical-mask-dataset

Shu, C., Ding, X., & Fang, C. (2011). Histogram of the oriented gradient for face recognition. *Tsinghua Science and Technology*, *16*(2), 216–224. https://doi.org/10.1016/S1007-0214(11)70032-3

Šochman, J., & Matas, J. (2004). AdaBoost with totally corrective updates for fast face detection. *Proceedings - Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, 445–450. https://doi.org/10.1109/AFGR.2004.1301573

Song, L., Gong, Di., Li, Z., Liu, C., & Liu, W. (2019). Occlusion robust face recognition based on mask learning with pairwise differential siamese network. *Proceedings of the IEEE International Conference on Computer Vision*, *2019-October*, 773–782. https://doi.org/10.1109/ICCV.2019.00086

Song, Q., Sun, X., Dai, Z., Gao, Y., Gong, X., Zhou, B., Wu, J., & Wen, W. (2021). Point-of-care testing detection methods for COVID-19. *Lab on a Chip*, *21*(9), 1634–1660. https://doi.org/10.1039/D0LC01156H

Sopharak, A., Dailey, M. N., Uyyanonvara, B., Barman, S., Williamson, T., Nwe, K. T., & Moe, Y. A. (2010). Machine learning approach to automatic exudate detection in retinal images from diabetic patients. *Journal of Modern Optics*, *57*(2), 124–135. https://doi.org/10.1080/09500340903118517

Spengler, M., Adler, M., & Niemeyer, C. M. (2015). Highly sensitive ligand-binding assays in pre-clinical and clinical applications: Immuno-PCR and other emerging techniques. In *The Royal Society of Chemistry* (Vol. 140, Issue 18, pp. 6175–6194). Royal Society of Chemistry. https://doi.org/10.1039/c5an00822k

Sreelekshmi, K. J., & Mahesh, T. Y. (2014). Human Identification Based on the Histogram of Oriented Gradients. *International Journal of Engineering Research & Technology (IJERT), 3*(7), 1611–1614. https://www.researchgate.net

Srinivasan, S., Rujula Singh, R., Biradar, R. R., & Revathi, S. A. (2021). COVID-19 monitoring system using social distancing and face mask detection on surveillance video datasets. *2021 International Conference on Emerging Smart Computing and Informatics, ESCI 2021*, 449–455. https://doi.org/.1109/ESCI50559.2021.9396783

Sultana, F., Sufian, A., & Dutta, P. (2018). Advancements in image classification using convolutional neural network. *Proceedings - 2018 4th IEEE International*

Conference on Research in Computational Intelligence and Communication Networks, ICRCICN 2018, 122–129. https://doi.org/1109/ICRCICN.2018.8718718

Surasak, T., Takahiro, I., Cheng, C. H., Wang, C. E., & Sheng, P. Y. (2018). Histogram of oriented gradients for human detection in video. *Proceedings of 2018 5th International Conference on Business and Industrial Research: Smart Technology for Next Generation of Information, Engineering, Business and Social Science, ICBIR 2018*, 172–176. https://doi.org/10.1109/ICBIR.2018.8391187

Susanto, S., Putra, F. A., Analia, R., & Suciningtyas, I. K. L. N. (2020). The face mask detection for preventing the spread of COVID-19 at politeknik negeri batam. *Proceedings of ICAE 2020 - 3rd International Conference on Applied Engineering*. https://doi.org/10.1109/ICAE50557.2020.9350556

Sutskever, Ii., Martens, J., Dahl, G., & Hinton, G. (2013). On the importance of initialization and momentum in deep learning. *ICML'13: Proceedings of the 30th International Conference on International Conference on Machine Learning*, 1139–1147. https://dl.acm.org/doi/10.5555/3042817.3043064

Swapnali, B. B., & Vijay, K. S. (2014). Feature Extraction Using Surf Algorithm for Object Recognition. *International Journal of Technical Research and Applications*, *2*(4), 197–199. www.ijtra.com

Tang, X., Wu, C., Li, X., Song, Y., Yao, X., Wu, X., Duan, Y., Zhang, H., Wang, Y., Qian, Z., Cui, J., & Lu, J. (2020). On the origin and continuing evolution of SARS-CoV-2. *National Science Review*, *7*(6), 1012. https://doi.org/1093/NSR/NWAA036

Tang, Y. (2013). Deep Learning using Linear Support Vector Machines. *ARXIV*. http://arxiv.org/abs/1306.0239

Teboulbi, S., Messaoud, S., Hajjaji, M. A., & Mtibaa, A. (2021). Real-Time Implementation of AI-Based Face Mask Detection and Social Distancing Measuring System for COVID-19 Prevention. *Scientific Programming*, *2021*. https://doi.org/10.1155/2021/8340779

To-Kelvin, K.-W., Tsang, O. T. Y., Leung, W. S., Tam, A. R., Wu, T. C., Lung, D. C., Yip, C. C. Y., Cai, J. P., Chan, J. M. C., Chik, T. S. H., Lau, D. P. L., Choi, C. Y. C., Chen, L. L., Chan, W. M., Chan, K. H., Ip, J. D., Ng, A. C. K., Poon, R. W. S., Luo, C. T., … Yuen, K. Y. (2020). Temporal profiles of viral load in posterior oropharyngeal saliva samples and serum antibody responses during infection by SARS-CoV-2: an observational cohort study. *The Lancet. Infectious Diseases*, *20*(5), 565–574. https://doi.org/10.1016/S1473-3099(20)30196-1

Tolba, A. S., El-Baz, A. H., & El-Harby, A. A. (2014). Face Recognition: A Literature Review. *International Journal of Signal Processing*, *2*(2), 88–103.

Tomás, J., Rego, A., Viciano-Tudela, S., & Lloret, J. (2021). Incorrect Facemask-Wearing Detection Using Convolutional Neural Networks with Transfer Learning. *Healthcare*, *9*(1050). https://doi.org/10.3390/healthcare908105

Tran, A. T., Hassner, T., Masi, I., Paz, E., Nirkin, Y., & Medioni, G. (2018). Extreme 3D Face Reconstruction: Seeing Through Occlusions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3935–3944. https://doi.org/10.1109/CVPR.2018.00414

Tripathi, M. K., & Maktedar, D. D. (2019). A role of computer vision in fruits and vegetables among various horticulture products of agriculture fields : A survey. *Information Processing in Agriculture*, *xxxx*. https://doi.org/10.1016/j.inpa.2019.07.003

Udugama, B., Kadhiresan, P., Kozlowski, H. N., Malekjahani, A., Osborne, M., Li, V. Y. C., Chen, H., Mubareka, S., Gubbay, J. B., & Chan, W. C. W. (2020). Diagnosing COVID-19: The Disease and Tools for Detection. In *ACS nano* (Vol. 14, Issue 4, pp. 3822–3835). NLM (Medline). https://doi.org/10.1021/acsnano.0c02624

Verma, R., Rohit Verma, M., & Ali, J. (2013). A comparative study of various types of image noise and efficient noise removal techniques. In *International Journal of Advanced Research in Computer Science and Software Engineering* (Vol. 3, Issue 10). www.ijarcsse.com

Vinitha, V., & Velantina, V. (2020). COVID-19 Facemask detection with deep learning and computer vision. *International Research Journal of Engineering and Technology*, *07*(08). www.irjet.net

Viola, P., & Jones, M. (2004). Rapid object detection using a boosted cascade of simple features. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *1*. https://doi.org/10.1109/cvpr.2001.990517

Walsh, P. (2019). Support Vector Machine Learning for ECG Classification. *CEUR Workshop Proceedings*, *2348*(10), 195–204. https://ceur-ws.org/Vol-8/paper14.pdf

Wandzik, L. , Kaeding, G. , & Garcia, R. v. (2018). Morphing Detection Using a General-Purpose Face Recognition System. *2018 26th European Signal Processing Conference (EUSIPCO)*, 1017–1021. https://www.eurasip.org/Pro/1570437948.pdf

Wang, C., Zhang, Q., Huang, C., Liu, W., & Wang, X. (2018). Mancs: A Multi-task Attentional Network with Curriculum Sampling for Person Re-Identification. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11208 LNCS*, 384–400. https://doi.org/10.1007/978-3-030-01225-0_23

Wang, H., Li, G., Zhao, J., Li, Y., & Ai, Y. (2021). An Overview of Nucleic Acid Testing for the Novel Coronavirus SARS-CoV-2. *Frontiers in Medicine*, *7*, 1063. https://doi.org/10.3389/FMED.2020.571709/BIBTEX

Wang, J., Yuan, Y., & Yu, G. (2017). Face Attention Network: An Effective Face Detector for the Occluded Faces. *Computer Vision and Pattern Recognition* . http://arxiv.org/abs/1711.07246

Wang, Y.-Q. (2014). An Analysis of the Viola-Jones Face Detection Algorithm. *Image Processing On Line, 4*, 128–148. https://doi.org/10.5201/ipol.2014.104

Wang, Z., Wang, G., Huang, B., Xiong, Z., Hong, Q., Wu, H., Yi, P., Jiang, K., Wang, N., Pei, Y., Chen, H., Miao, Y., Huang, Z., & Liang, J. (2020). *Masked Face Recognition Dataset and Application*. http://arxiv.org/abs/2003.09093

Whiting, P., Singatullina, N., & Rosser, J. H. (2015). Computed tomography of the chest: I. Basic principles. *Continuing Education in Anaesthesia Critical Care & Pain, 15*(6), 299–304. https://doi.org/10.1093/BJACEACCP/MKU063

WorkSafe. (2020). *Managing coronavirus (COVID-19) risks: Face masks in workplaces - WorkSafe*. https://www.worksafe.vic.gov.au/managing-coronavirus-covid-19-risks-face-masks-workplaces

World Health Organization. (2020a, January 29). *Advice on the use of masks the community, during home care and in health care settings in the context of the novel coronavirus (2019-nCoV) outbreak*. https://www.who.int/docs/default-source/documents/advice-on-the-use-of-masks-2019-ncov.pdf

World Health Organization. (2020b). *Advice on the use of point-of-care immunodiagnostic tests for COVID-19 Rapid diagnostic tests based on antigen detection*. https://doi.org/10.1101/2020.03.26.20044883

Wu, A., Peng, Y., Huang, B., Ding, X., Wang, X., Niu, P., Meng, J., Zhu, Z., Zhang, Z., Wang, J., Sheng, J., Quan, L., Xia, Z., Tan, W., Cheng, G., & Jiang, T. (2020). Genome Composition and Divergence of the Novel Coronavirus (2019-nCoV) Originating in China. *Cell Host & Microbe*, *27*(3), 325–328. https://doi.org/10.1016/J.CHOM.2020.02.001

Xiang, J., Yan, M., Li, H., Liu, T., Lin, C., Huang, S., & Shen, C. (2020). Evaluation of Enzyme-Linked Immunoassay and Colloidal Gold-Immunochromatographic Assay Kit for Detection of Novel Coronavirus (SARS-Cov-2) Causing an Outbreak of Pneumonia (COVID-19). *MedRxiv*, 2020.02.27.20028787. https://doi.org/0028787

Xie, X., Zhong, Z., Zhao, W., Zheng, C., Wang, F., & Liu, J. (2020). Chest CT for Typical 2019-nCoV Pneumonia: Relationship to Negative RT-PCR Testing. *Radiology*, *296*(2), E41–E45. https://doi.org/10.1148/RADIOL.2020200343

Xiong, X., & de La Torre, F. (2013). Supervised Descent Method and its Applications to Face Alignment. *IEEE*. https://doi.org/10.1109/CVPR.2013.75

Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). WIDER FACE: A face detection benchmark. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-December,* 5525–5533. https://doi.org/10.1109/CVPR.2016.596

Yang, S., & Rothman, R. E. (2004). PCR-based diagnostics for infectious diseases: uses, limitations, and future applications in acute-care settings. *The Lancet. Infectious Diseases*, *4*(6), 337. https://doi.org/10.1016/S1473-3099(04)01044-8

Yuan, X., & Park, I. K. (2019). Face de-occlusion using 3D morphable model and generative adversarial network. *Proceedings of the IEEE International Conference on Computer Vision*, *2019-October*, 10061–10070. https://doi.org/CV.2019.01016

Zhang, H., Wohlfeil, J., & Grießbach, D. (2016). Extension and evaluation of the AGAST feature detector. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *III–4*, 133–137. https://doi.org/10.5194/isprsannals-iii-4-133-2016

Zhou, P., Yang, X. lou, Wang, X. G., Hu, B., Zhang, L., Zhang, W., Si, H. R., Zhu, Y., Li, B., Huang, C. L., Chen, H. D., Chen, J., Luo, Y., Guo, H., Jiang, R. di, Liu, M. Q., Chen, Y., Shen, X. R., Wang, X., … Shi, Z. L. (2020). A pneumonia outbreak associated with a new coronavirus of probable bat origin. *Nature*, *579*(7798), 270–273. https://doi.org/10.1038/S41586-020-2012-7

Zhou, Z.-H. (2021). Semi-Supervised Learning. In Machine Learning (pp. 315–341). *Springer, Singapore*. https://doi.org/10.1007/978-981-15-1967-3_13

Zhu, N., Zhang, D., Wang, W., Li, X., Yang, B., Song, J., Zhao, X., Huang, B., Shi, W., Lu, R., Niu, P., Zhan, F., Ma, X., Wang, D., Xu, W., Wu, G., Gao, G. F., & Tan, W. (2020a). A Novel Coronavirus from Patients with Pneumonia in China, 2019. *New England Journal of Medicine*, *382*(8), 727–733. https://doi.org/MOA2001017

Zohuri, B. (2020). Deep Learning Limitations and Flaws. *Modern Approaches on Material Science*, *2*(3). https://doi.org/10.32474/mams.2020.02.000138

Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. v. (2018). Learning Transferable Architectures for Scalable Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 8697–8710. https://doi.org/10.1109/CVPR.2018.00907

APPENDIX A

Source Code for Image Preprocessing, Feature Extraction and Classification

```
clc; clear; tic;
for n=1:6000
   im{n} = imread(sprintf('no_mask (%d).jpg',n));
   figure
   imshow(im{n})
   % Detect facial Landmark using Viola Jones
   faceDetector = vision.CascadeObjectDetector;
   bboxes = faceDetectorim{n};
   IFaces = insertObjectAnnotation(I,'rectangle',bboxes,'Face');
   figure
   imshow(IFaces)
   title('Detected faces');

   %Crop face image
   I2 = imcrop(im{n},bboxes);

   %Convert cropped face image to grayscale image
   img=I2;
   gray_scale = rgb2gray(im{n});
   %Image enhancement
   J = imadjust(gray_scale);

   %Median Filter and image resizing
   J = medfilt2(J);
   J = imresize(J, [350 350]);
     %Feature Extractions
   corners = detectFASTFeatures(J,'MinContrast',0.1); % detect features using FAST
   [features,validPoints] = extractFeatures(J,corners, 'Method', 'BRISK'); %extract Brisk
   [hog1,visualization]     = extractHOGFeatures(J,corners,'CellSize',[32  32]);%extract
HOG
   [Brief1,visualize]         =  extractFeatures(I,points,'Method','ORB');    %extract ORB
points
   [rawpixels,visual]     = extractFeatures(I,points,'Method','Block'); %extract raw pixels
   Brisks1     = features.Features;
   Brief1      = Brief1.Features;
   raw1        = rawpixels.Features;
end
toc;
%save CSV files to disk
base_path = 'C:\Users\Documents\ projects\Mtech\datasets';
myFiles = fullfile(base_path,sprintf('Brisk1%d.csv',n));
csvwrite(myFiles,Brisk1)
myFiles1 = fullfile(base_path,sprintf('Brief1%d.csv',n));
csvwrite(myFiles1,Brief1)
myFile2 = fullfile(base_path,sprintf('raw1%d.csv',n));
csvwrite(myFiles2,raw1)
```

```
myFiles3 = fullfile(base_path,sprintf('hog1%d.csv',n));
csvwrite(myFiles3,hog1)
```

Feature Extraction using CNN

```
tic;
%Load face images;
imds =
imageDatastore('faceMask','IncludeSubfolders',true,'LabelSource','foldernames');
[imdsTrain,imdsTest] = splitEachLabel(imds,0.8,'randomized');
 %Load Alexnet
net = alexnet;
net.Layers
inputSize = net.Layers(1).InputSize
augimdsTrain = augmentedImageDatastore(inputSize(1:2),imdsTrain);
augimdsTest = augmentedImageDatastore(inputSize(1:2),imdsTest);
%Alexnet
layer = 'fc7';
featuresTrain = activations(net,augimdsTrain,layer,'OutputAs','rows');
featuresTest = activations(net,augimdsTest,layer,'OutputAs','rows');
YTrain = imdsTrain.Labels;
YTest = imdsTest.Labels;
toc;
```

Classification using SVM, KNN, NB

```
X = data(1:11000,:);
Y = labels1(1:1100,:);
%80:20
rand_num = randperm(11000);
Xtrain2 = X(rand_num(1:8800),:);
ytrain2 = Y(rand_num(1:8800),:);
Xtest2 = X(rand_num(8801:end),:);
ytest2 = Y(rand_num(8801:end),:);
% SVM model fit
% SVM classification with Gaussian kernel
svmMdl2=fitcsvm(Xtrain2,ytrain2,'KernelFunction','gaussian','Standardize',true,`Kernel
Scale','auto');
%SVM accuracy and confusion matrix
Svm_acc1=sum(predict(svmMdl2,Xtest2)== ytest2)/length(ytest2)*100
te2 = predict(svmMdl2,Xtest2);
Con2 = confusionmat(ytest2,te2);
%SVM precision, recall and f1-score
precision2 = Con2(1,1) / (Con2(1,1)+Con2(2,1))
recall2 = Con2(1,1) / (Con2(1,1)+Con2(1,2))
F_measure2 = (2*precision2*recall2)/(precision2+recall2)

%NB model fit
nb_model = fitctree(Xtrain2,ytrain2);

%Naive Bayes accuracy and confusion matrix
nb_acc=sum(predict(nb_model,Xtest2)== ytest2)/length(ytest2)*100
```

```
te3 = predict(nb_model,Xtest2);
confMatrix = confusionmat(ytest2,te3);

%NB precision, recall and f1-score
nb_precision = confMatrix(1,1) / (confMatrix(1,1)+confMatrix(2,1))
nb_recall = confMatrix(1,1) / (confMatrix(1,1)+confMatrix(1,2))
nb_F_measure = (2*nb_precision*nb_recall)/(nb_precision+nb_recall)

%KNN model fit
knn_model = fitcknn(Xtrain2,ytrain2,'NumNeighbors',3);

%KNN accuracy and confusion matrix
kn_acc=sum(predict(knn_model,Xtest2)== ytest2)/length(ytest2)*100
te4 = predict(knn_model,Xtest2);
knn_Con5 = confusionmat(ytest2,te4);

%KNN precision, recall and f1-score
knn_precision1 = knn_Con5(1,1) / (knn_Con5(1,1)+knn_Con5(2,1))
knn_recall1 = knn_Con5(1,1) / (knn_Con5(1,1)+knn_Con5(1,2))
knn_F_measure1 = (2*knn_precision1*knn_recall1)/(knn_precision1+knn_recall1
```