



Facial Anthropometry-Based Masked Face Recognition System

Kennedy Okokpujie^{1,2,3*}, Imhade P. Okokpujie^{4,5}, Fortress Abigail Abioye¹, Roselyn E. Subair⁶,
Akingunsoye Adenugba Vincent⁷

¹ Department of Electrical and Information Engineering, Covenant University, Ota 112101, Ogun State, Nigeria

² Africa Centre of Excellence for Innovative & Transformative STEM Education, Lagos State University, Ojo 102101, Lagos State, Nigeria

³ Informatics & Communication African Centre of Excellence (CApIC-ACE), Covenant University, Ota 112101, Nigeria

⁴ Department of Mechanical and Mechatronics Engineering, Afe Babalola University, Ado Ekiti 360001, Nigeria

⁵ Department of Mechanical and Industrial Engineering Technology, University of Johannesburg, Johannesburg 2028, South Africa

⁶ University Librarian, Afe Babalola University, Ado-Ekiti 360001, Nigeria

⁷ Director at OVA Foundation, Millington YO42, Maryland, USA

Corresponding Author Email: kennedy.okokpujie@covenantuniversity.edu.ng

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.290302>

ABSTRACT

Received: 25 February 2023

Revised: 25 May 2023

Accepted: 8 October 2023

Available online: 20 June 2024

Keywords:

masked face recognition, unmasked face recognition, facial Anthropometry, COVID-19, facemask, facial landmarks, Local Binary Pattern Histogram, biometric, craniofacial plexus, face size, facial index, intercanthal index, orbital width index, nasal index

Different kinds of occlusion have proven to disrupt the accuracy of face recognition systems, one of them being masks. The problem of masked faces has become even more apparent with the widespread of the COVID-19 virus, with most people wearing masks in public. This brings up the issue of existing face recognition systems been able to accurately recognize people even when part of their face and the major identifiers (such as the nose and mouth) are covered by a facemask. In addition, most of the databases that have been curated in different organizations, countries are majorly of non-masked faces, and masked databases are rarely stored or universally accepted compared with conventional face datasets. Therefore, this paper aim at the development of a Masked Face Recognition System using facial anthropometrics technique (FAT). FAT is the science of calculating the measurements, proportion and dimension of human face and their features. A dataset of faces with individual wearing medical face mask was curated. Using the Facial anthropometry based technique a Masked Face Recognition System developed. This system was implemented using Local Binary Patterns Histogram algorithms for recognition. On testing the developed system trained with unmasked dataset, show a high recognition performance of 94% and 96.8% for masked and non-masked face recognition respectively because of the Facial anthropometry based technique adapted. On deployment, users were been recognized when they are wearing a mask with part of their face covered in real-time.

1. INTRODUCTION

Face recognition has advanced widely and become so significant in recent years with the rapid advancement in biometric identification technologies. It is a widespread concept that provides convenient and accurate identity verification in many areas such as military, finance, transportation, public security and daily life generally. The outbreak of the COVID-19 pandemic has also further increased the need for contactless identity verification to promote hygiene. However, with this outbreak and the need to keep the pandemic under control, the wide use of face masks has been recommended and even made compulsory in public. The identification of people while they are wearing masks has presented a sensitive issue on face recognition, as the mask covers a significant section of the face. This has brought up a challenge on the existing face recognition systems, making them less effective and therefore compromising security on many levels.

With the outbreak of COVID-19, not only do face masks have a strong and significant influence in preventing the spread of the disease, but they also have other observed important societal effects that gives room for quarantining, social distancing, strict isolation and such other precautionary steps to be a little relaxed, making them vital in these circumstances. This has led to many organizations recommending and requiring the use of facemasks that will fully cover the nose and mouth and even an increasing number of countries passing it as a law and making the proper use of facemasks when appearing in public mandatory, with penalties following for persons who flout it. In response to this, there are now a large and increasing proportion of persons using facemasks in public.

On the contrary, wearing of these masks has been observed to facilitate issues such as criminals. Whereby, taking advantage of the mask to steal or commit other crimes without being identified, wanted criminals being able to easily bypass security systems and surveillance. Couple with the fact that the

current systems for face recognition perform poorly when a person is using a facemask, and large portions of the face marks are covered.

Since the early 1990s, when face recognition became popular following the birth of the historical Eigenface approach [1, 2], there has been a record of high performance achieved in many applications, even though some factors like occlusions, facial expression, illumination, and poses still have their effects on face recognition methods [3-6].

As shown in Figure 1, during enrollment, the biometric trait (**B**) of a user (**Y**) is scanned by a sensor and a digital representation (**M**) of the features are acquired. To help with recognition later, a feature extractor is used to obtain a representation known as a feature set and is stored as a template in the system database (**D**) for later recognition. The concept of feature extraction is given by Eq. (1):

$$X = f_e(M) \quad (1)$$

When the user is to be recognized or identified, another input is taken and a new biometric data is acquired. Its features (X_R) are extracted and then contrasted (f_m) against the templates found in the system database. A match score is produced based on this comparison so as to decide the identity (\hat{Y}) that has been affiliated with that particular biometric feature set. The match score, S is given by Eq. (2).

$$S = f_m(X_R, D) \quad (2)$$

Since the goal in this case is to identify the user, the new extracted information is compared with all the templates available in the database to find a match. This is called a one-to-many mapping. The main aim of any biometric system is to accurately give a match, such that the predicted identity will be the same as the user identity; as shown in Eq. (3).

$$\hat{Y} = Y \quad (3)$$

Face occlusion has been an issue repeatedly considered in many face recognition solutions, and creating occlusion invariant face recognition solutions has been a challenge studied in many pieces of research [7]. However, it is good to note that a wide section of research on general occlusion focuses mainly on the wearing of glasses, hats, scarves or partial captures. Studies have shown that some elements of the face are more essential for face recognition than some others

and that covering the mouth [8, 9] affects the accuracy of the system compared to when it is open [10]. In the study by Dhamecha et al. [11], it is reported that the efficiency of face recognition and identification systems was compromised when the bottom half of the face was hidden.

Another important challenge contributing to the recognition of masked faces is the unavailability of masked faces datasets to train our systems with the study by Anwar et al. [12]. Most of the databases we have and use in many organizations and countries worldwide consist of just unmasked faces, and creating masked face datasets all over again for the purpose of recognition is another hectic process.

For better understanding of this problem, two different face recognition processes, namely Masked Face Recognition and Face Mask Recognition are briefly distinguished [13, 14]. Masked face recognition is the process of recognizing a face and its identity, while a mask is on, utilizing other uncovered features of the face. Facemask recognition, on the contrary, is one that checks whether the individual has a face mask on or not, and this is used mainly in environments where the use of a mask is mandatory. The focus here is on the first concept and to solve this, this work presents a novel face recognition solution that works with an unmasked-faces database to accurately verify the identity of people from their faces under circumstances when their face mask is worn or not. This method automatically crops off the masked area and uses only the available upper face features for recognition of any masked or unmasked individual. Three different algorithms are employed: the Local Binary Pattern Histogram (LBPH) [15], Principal Component Analysis (PCA) [16] and Fisherface [17] techniques. At first, we use the Haar-cascade classifier to detect faces from images acquired. The Haar-cascade classifier is regarded as the most efficient object detection machine learning-based technique and was proposed in 2001 by Jagtap et al. [18]. After face detection from the input images, any of the three algorithms can be used for feature extraction from the facial image, which are then used for classification and training. A statistical difference and analysis of the accuracy between masked face recognition and non-masked face recognition using the most efficient algorithm, Local Binary Patterns Histogram is presented in this paper.

Some of the contributions of this research paper to the body of knowledge include:

- (i) Successful development of a Masked Face Recognition System that can accurately recognize individuals despite being masked, and when unmasked in a matter of seconds.

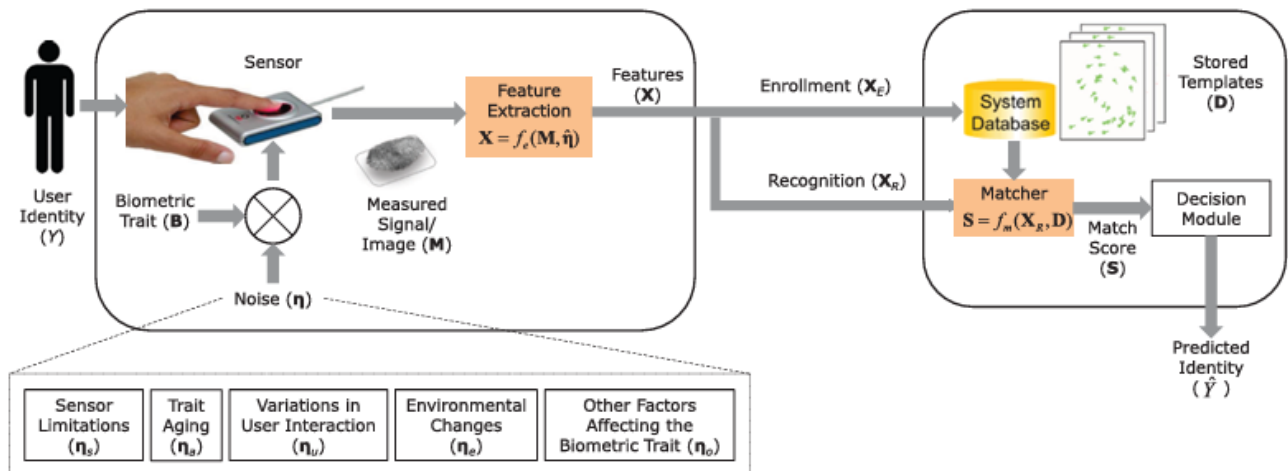


Figure 1. Operation of a typical biometric system [19]

(ii) Successful implementation of the system to work based on unmasked databases to recognize masked individuals.

(iii) Analysis of the developed model to know its reliability, efficiency and accuracy.

(iv) Successful implementation of the developed system on a Raspberry pi single-board computer, which can work with any mobile device.

(v) With the development of this prototype, the security risks involved with the use of face mask has been greatly reduced and therefore, facilitates the use of this precautionary measure to reduce the spread of the COVID-19 virus.

1.1 Related works

In the recent past years, some research works have been done and have expanded the field of masked face recognition, to find solutions to face mask occlusion of the face. A detailed study is done to check the proposed method(s), findings and results, relevance, strengths and weaknesses of some of these works.

Wang et al. [20] aimed to increase the present facial recognition system's performances on faces that are masked. Since the majority of the current systems for face recognition are founded in deep learning, which requires huge amounts of data to be effective, this study presents three different classes of masked-face databases: RMFRD- Real-world Masked Face Recognition Dataset, SMFRD- Simulated Masked Face Recognition Dataset and MFDD- Masked Face Detection Dataset. Leveraging these datasets, cutting-edge technology was developed and achieved an accuracy of 95%. The advantage of this is that these datasets can be used for both masked face identification and face mask detection, and it also integrates the benefits of the current publicly available facial recognition databases. Though this method gave a good result, it is not quite as accurate as traditional facial recognition technology, which has a 99% accuracy rate.

Anwar and Raychowdhury [12] used a freely available package known as MaskTheFace for retraining existing facial recognition systems to improve accuracy. This brought about a rise in the true-positive rate of the Facenet system by 38 per cent. This approach is resourceful as it uses existing facial databases and augments it with techniques that allow for the recognition of masked faces with low error rates and a great total performance without having to replicate the user's database by capturing new images for verification. However, The False Acceptance Rate (FAR) is averagely at 0.2%, and when tested on real-world photos, performance drops by 2 to 4%.

Boutros et al. [21] presents the use of Self-Restrained Triplet (SRT) and Embedding Unmasking Model (EUM), which for most experimental circumstances, significantly enhances the efficiency of masked face verification. The suggested method is developed to work in conjunction with present facial recognition technologies, thereby avoiding the demand that existent face recognition systems for non-masked faces be retrained. However, poorer results are obtained whenever the Embedding Unmasking Model (EUM) is not paired to use with the SRT on face recognition models.

Ye [22] studied the existing problem, collected relevant images, created a database, and offered a new solution for masked faces. The testing of the database reveals that by a minimum of 16.8 per cent, this suggested approach surpasses six current state-of-the-art technologies. This framework is

built on CNN (Convolution Neural Network) and consists of four different submodules that can deal with the problem robustly; however, the Average Precision over the training dataset achieves just up to 76.8% accuracy.

Ejaz et al. [23] used the PCA algorithm, to obtain an average accuracy of 72% on masked face image recognition and about 95% accuracy for unmasked faces. The strength of this method is that PCA is very good for regular facial recognition, and its weakness is that PCA gives a less accurate rate for recognizing faces that are masked as opposed to unmasked face images. The concern for the future should be to enhance the performance of recognition of masked faces using other refined ML techniques.

Li et al. [24] devised a technique called HGL in order to manage head pose classification by using line portrait and evaluation of the colour texture of images. This presented approach achieved a 93.64 per cent performance for front accuracy and an 87.17 per cent performance for the side. This method proved to be 0.8 per cent better above the performance of alternative approaches for the front and 2.28 per cent better for the side, with the downside that the side accuracy is lower compared to the front accuracy.

Mundial et al. [25] obtained a masked face database in order to train the SVM- Support Vector Machine classifier on cutting edge Facial Recognition Feature vector. When applied on masked faces, this technique produced a 97 per cent accuracy, and when applied on the LFW database, about 99 per cent performance was achieved. However, when only non-masked classifiers are used to train the network and then it is applied with masked faces, the performance of the system dropped to 79 per cent. This strategy can be applied for larger solutions, including video surveillance.

Li et al. [26] used a cropping-based strategy in conjunction with Convolutional Block Attention Module (CBAM). This integration produced an optimum performance rate for MFR, and the cropping-based technique can efficiently work with either the use of masked faces or unmasked faces as the training data to recognize both faces without masks and faces with masks, making this approach very flexible and suitable. Though the attention mechanism performs poorly in routine face recognition, this solution achieves an excellent accuracy when recognizing masked faces as opposed to the performance of other cutting-edge approaches.

Damer et al. [27] presents a carefully gathered dataset consisting of three modules, all of them with their own unique capture commands in order to test three of the most efficient facial recognition technologies in real-world scenarios: two academic solutions and one commercial off-the-shelf (COTS) system. It is observed that masked probe faces affect the ArcFace and SphereFace recognition accuracy, and on the other hand, the COTS technology retains a near-perfect accuracy rate. Nonetheless, in these three systems, a considerable change in their true values is still consistent.

Hariri [13] built a technique based on deep-learning features and on discarding the mask-occluded region so as to find a solution to the issue of recognition of masked faces. A best recognition rate of 91.3% was achieved. It is noted that this is the very first approach that dealt with the masked face recognition issue amidst the outbreak of COVID-19. However, the face recognition performance needs more improvement to the level of accuracy of regular FR systems as this issue and solution is not applied for only the COVID-19 era.

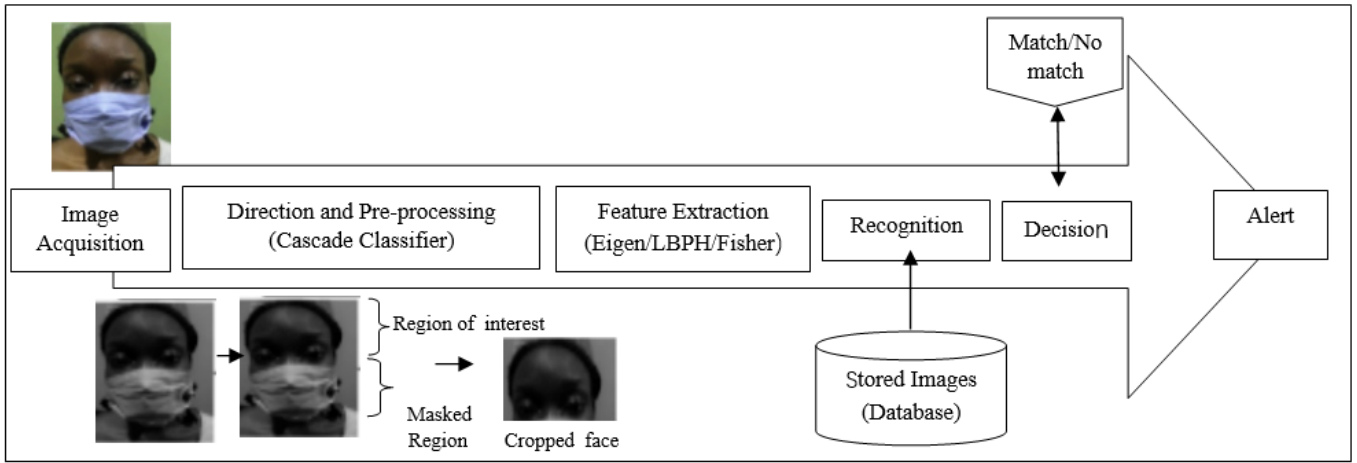


Figure 3. Conceptual framework of the Masked Face Recognition System

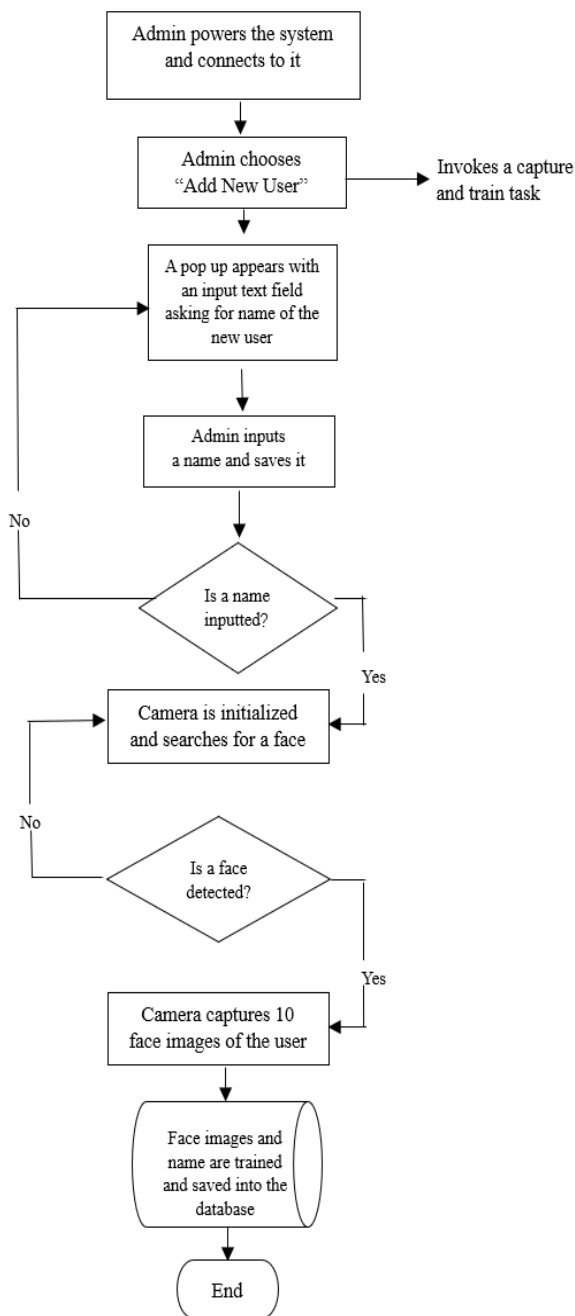


Figure 4. Activity flow diagram for enrollment phase

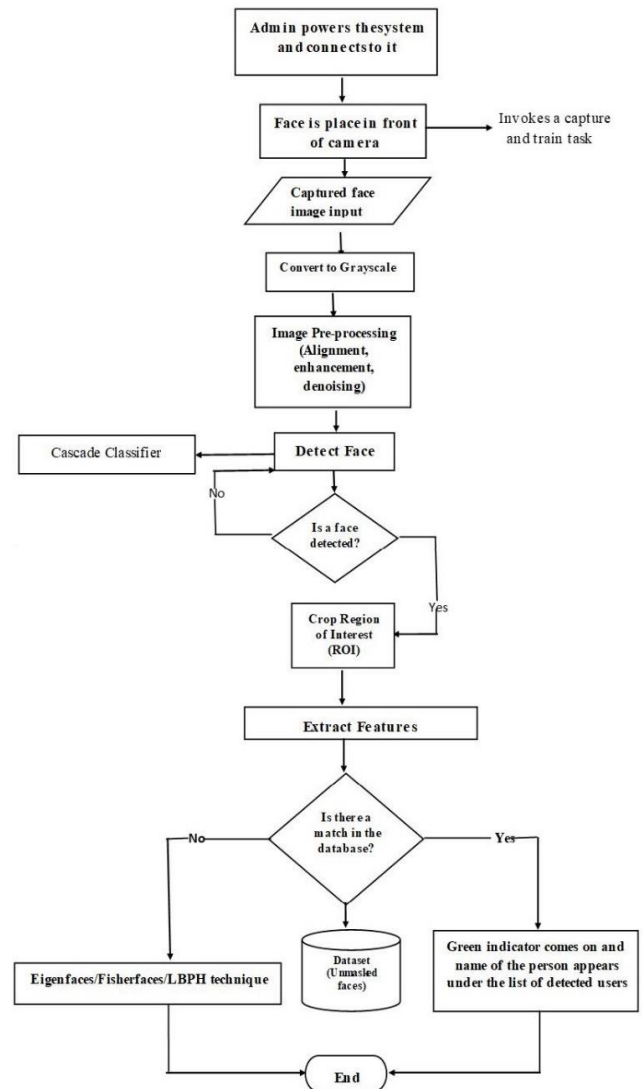


Figure 5. Activity flow diagram for recognition phase

For enrollment, the face to be captured and trained is placed in front of the device camera, which grabs frames when a face is detected. Ten pictures of the face are taken, and every picture is stored in a jpg format. A system database is created for this purpose for the storage of these images and the extracted features. Labels/identifiers are assigned to every set of images that relate to a particular person to enable the system

to call or identify the individual.

For recognition, either a masked or unmasked face can be used as an image input. This image will be processed and features extracted for comparison with the database in order to make a decision as shown in Figure 5.

2.2.2 Image pre-processing and face detection

The image acquired as an input into the system is first converted to grayscale before processing can be done. Face detection is done to verify that a face is present in the image and if so, the image is cropped to only the face and then cropped again to only the upper region of the face, as this is the region of interest (ROI). The ROI is defined by the use of a rectangle taking the total face width and half of the face height. Other parts of the image are discarded and only the upper facial features are used for extraction and recognition. For face detection, the Haar-cascade classifier is employed.

2.2.3 Feature extraction

Three different face recognition algorithms have been adopted for feature extraction and matching to accurately get the user's identity. They include Eigenface, Fisherface and Local Binary Pattern Histogram (LBPH). These techniques have their different methods for recognition. They have been programmed for use on the system, and only one can be used at a time, based on the administrator's preference. Testing has shown that the LBPH method is the most accurate for testing but is slower, while the Eigenface is the fastest, though it does not reach the level of accuracy of LBPH [29, 30]. As a result of need for the system to more accurately the LBPH algorithm only was adopted in this research.

2.2.4 Training and classification

During the enrollment phase, the extracted facial features of the acquired face are trained and stored into the database with an ID name, which is used to identify and call that face when it is time for recognition. During the recognition phase, after the masked/unmasked face to be recognized is processed and its features extracted, the set of extracted features are compared with those that are in the system database to either classify the face as a match or not. If a match is found, the face is recognized, and the user ID associated with the face is displayed under the list of detected users. If no match is found, it means the individual is a stranger and could either be enrolled into the database or any other action can be taken, based on the administrator's choice.

2.3 System development

The Masked Face Recognition System (MFRS) is developed having both the hardware and software part.

2.3.1 System hardware

The whole process of enrollment and recognition is implemented using just the Raspberry Pi 3 Model B+ and its camera. An external memory card is also added for the storage of the system database.

2.3.2 Software elements

With the aim of this prototype in mind, the elements used for the development of the system software are as follows:

(1) C# (C Sharp): This programming language was developed by Microsoft and runs on the .NET framework. It is based on other languages like C, C++ and is used for the

development of the database, web services and various other applications. C# is designed for Common Language Infrastructure (CLI), which allows executable code and runtime environment. This is the main language used for the programming of the software aspect of this system.

(2) EmguCV library: This is a platform that allows functions from OpenCV image-processing library to be called from .NET languages, which includes C#. It is the OpenCV binding for C#. It can run on various OS and is compiled in this case by Visual Studio.

(3) Blazor: This is a free and open-source framework created by Microsoft. It is a UI framework with the ability to create great web user interfaces (UI) using just HTML, CSS and C#. It is used here for developing the administrator or user interface.

(4) CSS: The CSS language is mainly used for designing or styling a web page. It describes how the different elements of the page will be laid out and displayed on screen or any other way. It is used here for UI styling.

(5) JavaScript: This is a dynamic and the most popular programming language for the web. Very detailed codes/instructions can be written through this, and it is used to design how the user/admin will interact with the webpage for in-browser code. It is employed here for those purposes.

(6) VSCode: Visual Studio Code is a source-code editor used for developing and debugging various cloud or web applications. It runs on macOS, Windows and Linux, comes with in-built functions and is very flexible, easy and customizable. It is used here as the IDE.

(7) Haar-Cascade: Haar-Cascade is classifier which is used for face detection. The trained Haar-Cascade file can be found here [28, 31].

3. RESULTS AND DISCUSSION

3.1 Hardware implementation

The Raspberry Pi is the major hardware module used for the hardware implementation, while some other components are connected to it for full operation of the system. The Raspberry pi comes with its own RPI camera, which captures real-time events as they happen and then spots faces and captures them for either enrollment or detection. An external microSD card is inserted into the Pi to carry the system's database, which can carry as much as 91,428 different enrolled persons at once, given the current storage size. Three LEDs are also connected, one internal red LED to signify stability, and two external red and green LEDs. The green LED comes on when a face is detected and recognized, and if not, the red LED comes on. A USB Type-A charger is used to power the Raspberry Pi, and a network cable can be used to connect the Raspberry Pi via SSH to a computer for backend access and control.

Figure 6 shows the internal connection of the Raspberry Pi to other components such as the camera and LEDs, while Figure 7 shows the final packaging of the hardware device and how it looks when powered on and in operation.

3.2 Software implementation

Figure 8 depicts the system deployment and implementation. Some libraries and programming elements were utilized for the software implementation process of the project. The prototype is designed for the Raspberry Pi to connect to a

mobile device and be interacted with via a web server. C# was used for writing many of the codes on the VS Code IDE. EmguCV library was used to bind the C# language to OpenCV. Blazor, a UI framework, was used to design the webpage user/admin interface. JavaScript was used to design how every

interaction between the admin and the web server will happen. Finally, CSS was used for the user interface styling. All the necessary libraries were installed, the code development workspace and the setup were used to run the initial code, compile and debug the errors.

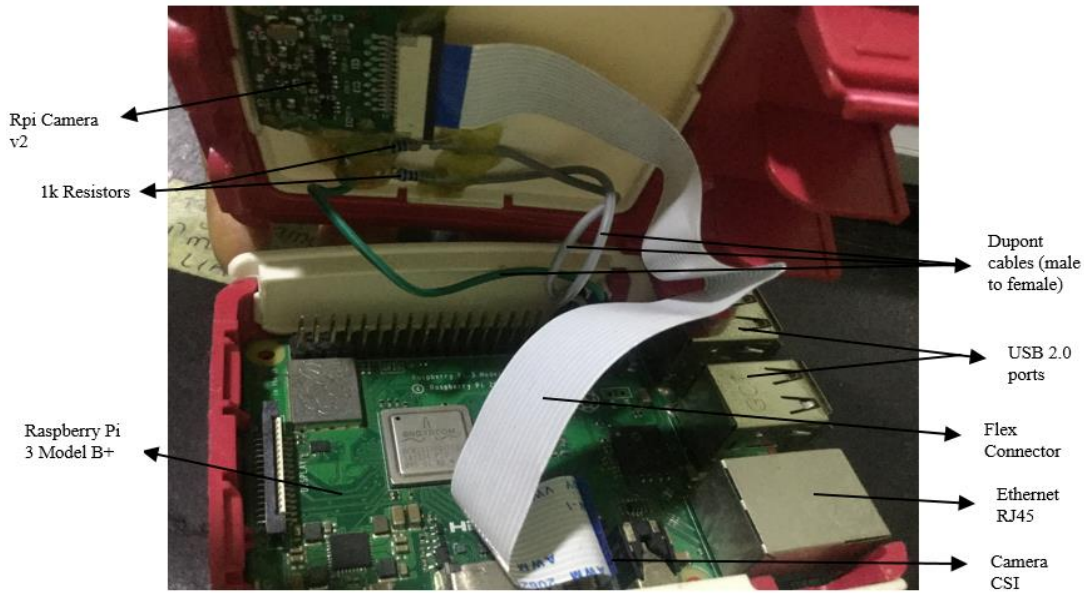


Figure 6. Internal connection of the hardware raspberry Pi to other components

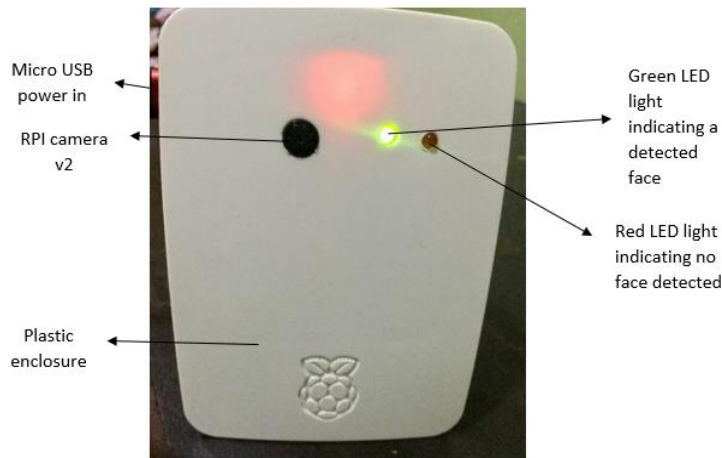


Figure 7. Final packaging of the system device

The Raspberry Pi, after being powered on needs the following to be connected for communication and control: A network connection (Ethernet or Wi-Fi), a mobile device connected to the same network and the Pi’s IP address. In this research, the Raspberry Pi is designed to connect to a mobile device via the hotspot. For security and access control purposes, the hardware can only connect to a specific hotspot username and password, which will belong to the administrator. After the connection is made to the right device, the Pi can be connected to the internet by using its IP address.

This will lead to a web server that will show in real-time what the Pi’s camera is seeing. The web server is the interface with which the admin can see what is going, make settings on the type of detector to be used, the recognition threshold and the detection size. The system is designed with a recognition threshold and detection size in the range of 0-1000, and the desired setting can be made by the admin. However, the threshold is set at 300 and the detection size at 500 automatically when the connection is made been the best

setting during design/testing stage of the prototype. Also, the system is designed to be most efficient for recognition when the threshold number is low, and the detection size is large.

The web server shows a real-time visualization of the system operation, shows the list of all registered users and the list of detected users in the timeframe it has been connected. Every activity that is in the range of the camera is shown, and once a face appears before the camera, whether it is masked or unmasked, is recognized or unrecognized by the system, depending on if that face has been enrolled or registered in the system database. If a face is recognized, it appears under the list of detected users, and if another face appears and is recognized, it is also placed in the list and so on. However, if a face is not recognized, the raspberry pi’s red light comes on, indicating that the face is not recognized. The webpage can be conveniently refreshed whenever the admin wants to do so, which will clear the list of detected users and starts recognizing and detecting afresh.

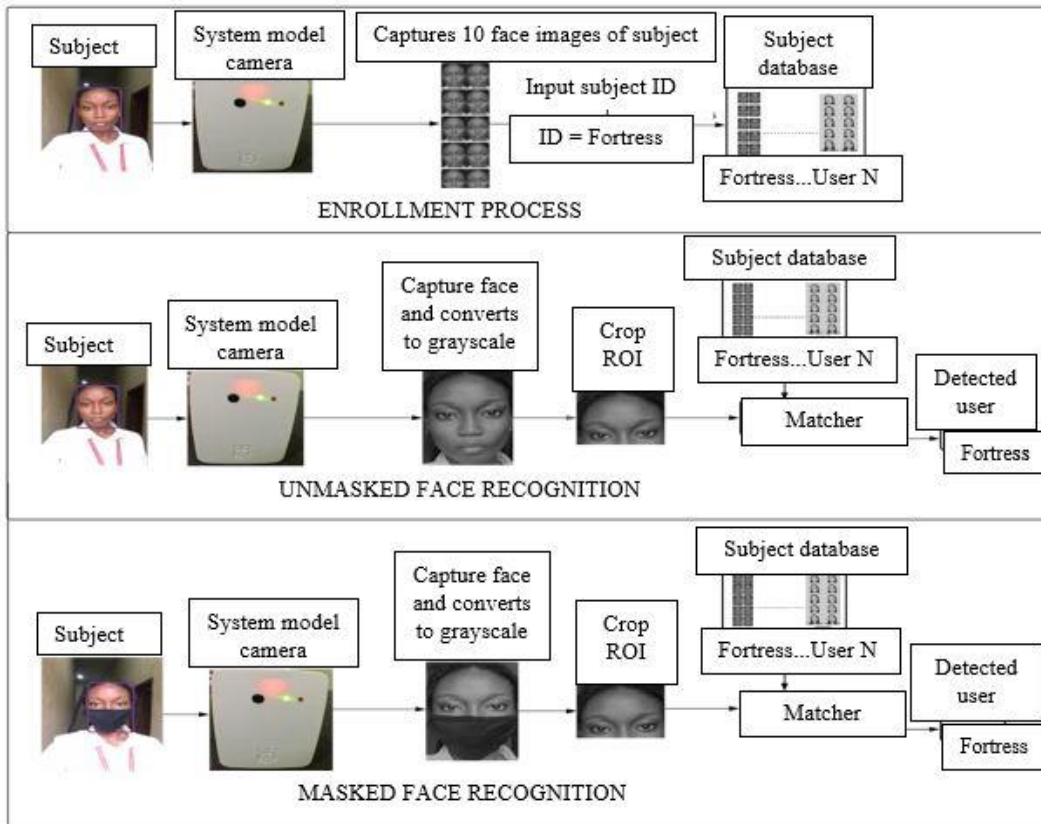


Figure 8. System deployment / implementation

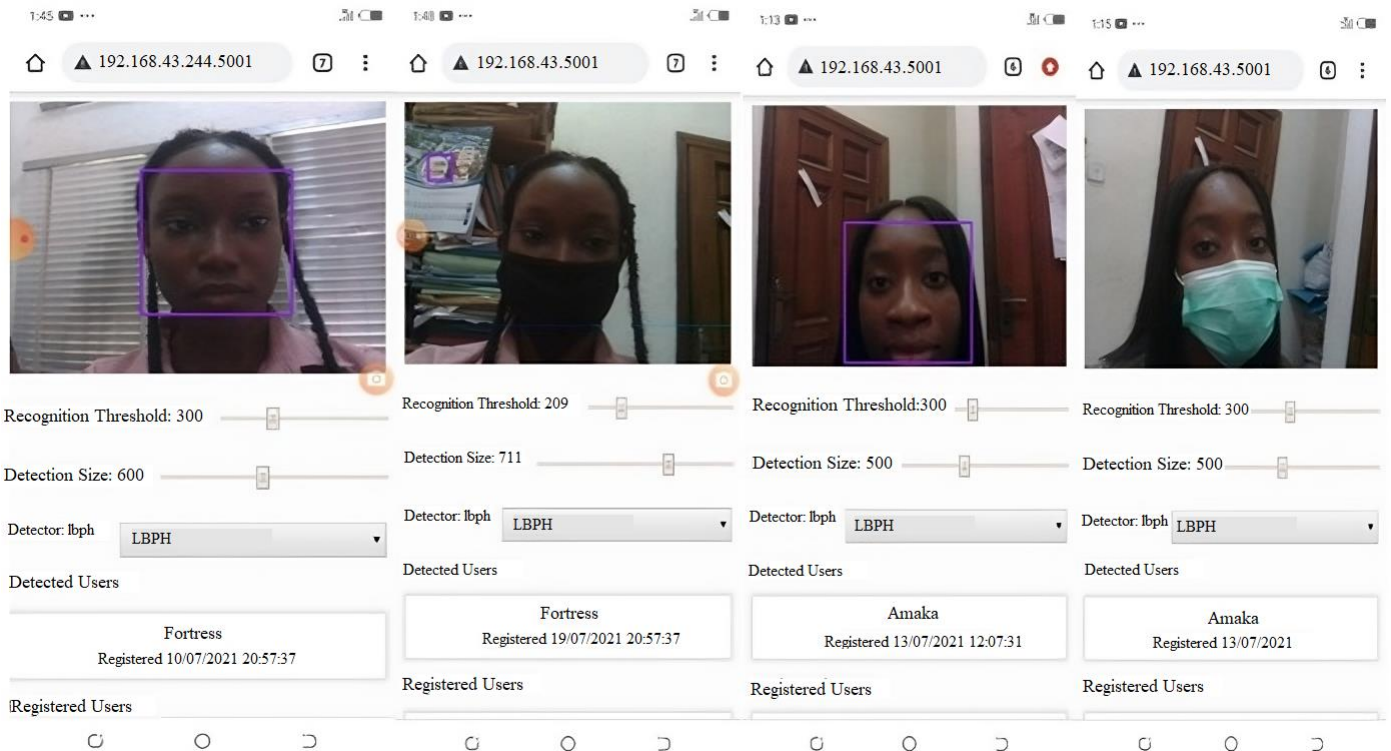


Figure 9. Samples of subject masked and unmasked faces correctly recognized

Figure 9 shows screenshots of the web server of the system displaying the system interface, camera output, recognition threshold, detection size, type of detector being used, the list of registered users, and a screenshot of the web server of the system when it recognizes a user and his/her name appears under the list of detected users.

3.3 System testing

The dataset used for training the system composed of 2500 colored unmasked face images of individuals with 10 different images per person, amounting to 250 unmasked subjects and user IDs in the database. Following the pipeline for

recognition, these images were first converted to grayscale, cropped and resized to the region of interest, preprocessed and then stored.

For each of the 250 unmasked subjects enrolled with a user ID into the database, their faces were used to test the system and check its recognition performance for both when the subjects were masked and when they were unmasked. The average and total training/enrollment time and the average and total recognition time for both masked and unmasked faces were measured and recorded.

Tests were carried out with the developed prototype to ensure its efficient performance. As the research work has two main parts, which are enrollment and recognition, the system was tested to check its performance under both functions. The reasons why the system is tested are to measure its performance, check its reliability and know its accuracy. For testing this system, we only reported the LBPH detector on this paper, because it is generally the most effective technique amongst the three which have been implemented for on this

system. The recognition threshold and detection were alternated to achieve the best setting.

Table 2. Recognition result of sample faces

S/N	Parameter	Value
1	Number of images in the training set	2500
2	Number of masked and unmasked subjects tested	250
3	Average Enrollment Time	1.11 secs
4	Total Enrollment Time	277.50 secs
5	Number of unmasked subjects recognized	242
6	Number of masked subjects recognized	235
7	Average Unmasked Face Recognition Time	0.9 secs
8	Total Unmasked Face Recognition Time	225 secs
9	Average Masked Face Recognition Time	1.4 secs
10	Total Masked Face Recognition Time	350 secs
11	Unmasked Recognition rate	96.80%
12	Masked Recognition rate	94.00%

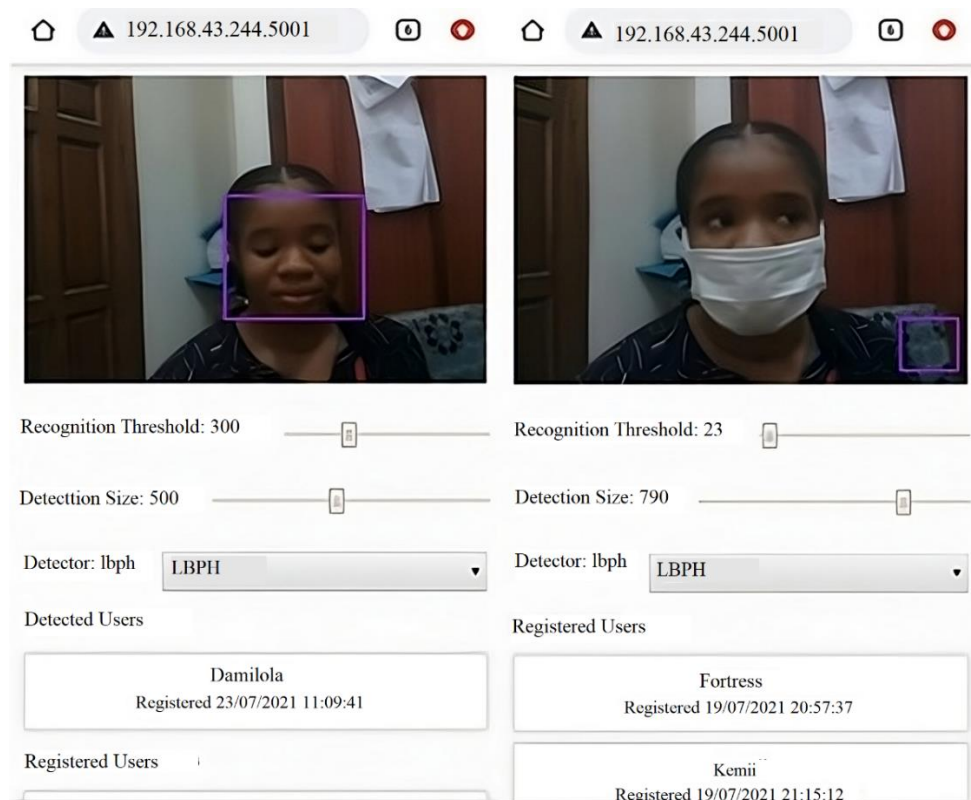


Figure 10. Sample of masked face subject not recognized

Two hundred fifty (250) user facial images were taken and enrolled along with their user ID. After enrolling these individuals, the system was tested to check if it could recognize the individuals while they are masked and also while they are unmasked. Results were then taken so that performance can be measured. Table 2 shows the result of these samples of faces that were taken and enrolled.

The samples of the 250 individuals tested for this research are listed, and after testing, it was observed that 235 individuals were recognized while masked and 15 were not recognized. When unmasked, 242 individuals were recognized, and 8 were not recognized. The causes of these outliers are attributed to poor network, poor lighting or also the contrast of the color of the mask against the face as shown in Figure 10. The final results as depicted in Figures 11 to 13.

Masked Face Recognition

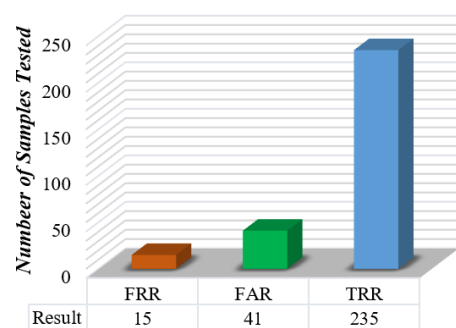


Figure 11. Graphical analysis of masked face recognition

Unmasked Face Recognition

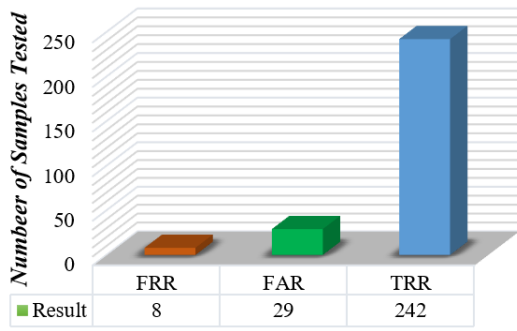


Figure 12. Analysis of unmasked face recognition

3.4 System backend control

The administrator can have full control of the system, view the database or make additional changes to the program and function of the raspberry pi. From the backend, pictures can be added manually to the Pi by transfer. Since the database contains a folder for every user ID that has been created along with their enrolled pictures, the administrator can manually create these folders and transfer the processed user pictures into the database. This will add the user and enable the system to recognize him/her; although, this is not so recommended as accuracy is much higher when enrollment into the database is done through the raspberry pi’s camera.

where,

$FRR = \text{False Reject Rate}$, $FAR = \text{False Accept Rate}$,
and $TRR = \text{True Recognition Rate}$.

Backend access is established by connecting the powered-

on Pi device to a computer via a network cable. Once it is well connected, software or application can be used to view the raspberry pi on the computer monitor. To connect and display, the Pi’s host, username and password details are needed, given below:

- Username: Pi
- Password: raspberry
- Host: raspberrypi.local

A software called FileZilla FTP Client is used here to connect the raspberry pi and display its content. This software allows changes and configurations to be made and files to be transferred from the local computer to the pi. Connection is made over port 22, which is the port used for SSH- Secure Shell communication and allows the administrator to connect to the machine.

Figure 14 shows a screenshot of the backend display when the raspberry pi is connected to a computer and its contents are visible and accessible.

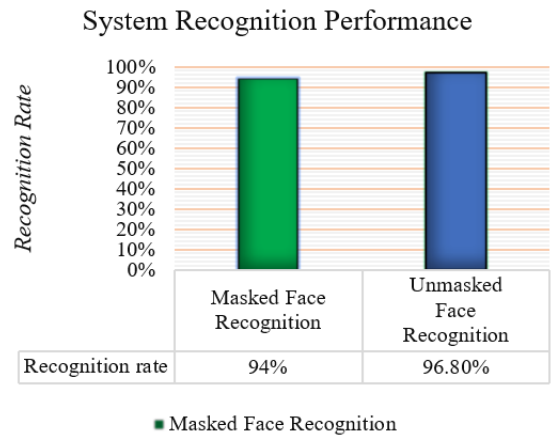


Figure 13. Graphical representation of masked and unmasked face recognition accuracy

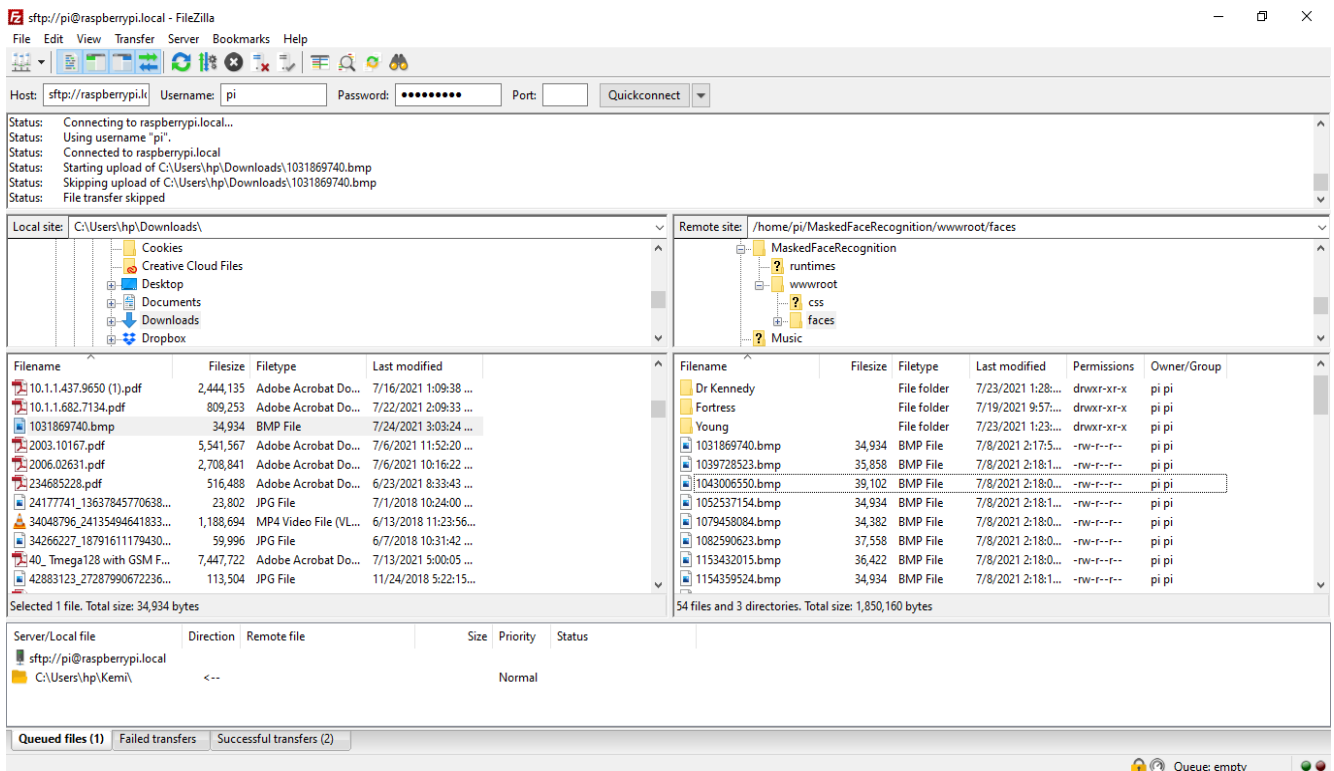


Figure 14. System backed access control

4. CONCLUSIONS

This research was motivated by the arrival and widespread of the COVID-19 virus, which caused countries and governments to mandate the use of facemasks by their population while they are in public. This brought up the challenge of occluding the face and making it hard for face recognition systems to accurately identify who is behind the mask. Asking people to remove their masks while in public so that they can be recognized is unethical and unlawful, as it has been passed on as a law in many countries that face masks must be worn in public. This research aims to solve that problem by developing a face recognition model that can identify and recognize people effectively even when they are masked or unmasked on the already existing unmasked face database that is globally available.

This system is also designed to work with databases of unmasked faces, as there are hardly any organized masked face datasets. Detailed research was done to check the works that have been done in this area, methodologies, results and challenges. It included analysis and comparisons of the different methods and outcomes. Based on this study, a methodology, which was summarized in a framework, was chosen.

This prototype architecture involves both the software and the hardware parts. A software was developed to carry out the required processes from the first to the last, as shown by the project framework and stages. Thereafter, the designed system was implemented on a raspberry pi hardware module for use in any application. Finally, tests were carried out to accurately measure the performance of the designed system

ACKNOWLEDGMENT

This paper is partly sponsored by Covenant University Center of Research, Innovation, and Discovery (CUCRID) Covenant University, Ota, Ogun State, Nigeria.

REFERENCES

- [1] Turk, M., Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1): 71-86. <https://doi.org/10.1162/jocn.1991.3.1.71>
- [2] Kortli, Y., Jridi, M., Al Falou, A., Atri, M. (2020). Face recognition systems: A survey. *Sensors*, 20(2): 342. <https://doi.org/10.3390/s20020342>
- [3] Bassili, J.N. (1979). Emotion recognition: The role of facial movement and the relative importance of upper and lower areas of the face. *Journal of Personality and Social Psychology*, 37(11): 2049-2058. <https://doi.org/10.1037/0022-3514.37.11.2049>
- [4] Okokpujie, K., Apeh, S. (2020). Predictive modeling of trait-aging invariant face recognition system using machine learning. In: Kim, K., Kim, H.Y. (eds) *Information Science and Applications. Lecture Notes in Electrical Engineering*, vol 621. Springer, Singapore. https://doi.org/10.1007/978-981-15-1465-4_43
- [5] Okokpujie, K., John, S., Ndujiuba, C., Badejo, J.A., Noma-Osaghae, E. (2021). An improved age invariant face recognition using data augmentation. *Bulletin of Electrical Engineering and Informatics*, 10(1): 179-191. <https://doi.org/10.11591/eei.v10i1.2356>
- [6] Kennedy Okokpujie, S.J., Donald, U., Nwagu, M., Noma-Osaghae, E., Ndujiuba, C., Okokpujie, I.P. (2020). Development of an illumination invariant face recognition system. *International Journal*, 9(5): 9215-9220. <https://doi.org/10.30534/ijatcse/2020/331952020>
- [7] Song, L., Gong, D., Li, Z., Liu, C., Liu, W. (2019). Occlusion robust face recognition based on mask learning with pairwise differential Siamese network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Seoul, Korea (South), pp. 773-782. <https://doi.org/10.1109/ICCV.2019.00086>
- [8] Kotsia, I., Buciu, I., Pitas, I. (2008). An analysis of facial expression recognition under partial facial image occlusion. *Image and Vision Computing*, 26(7): 1052-1067. <https://doi.org/10.1016/j.imavis.2007.11.004>
- [9] McKelvie, S.J. (1976). The role of eyes and mouth in the memory of a face. *The American Journal of Psychology*, 89(2): 311-323. <https://doi.org/10.2307/1421414>
- [10] Davies, G., Ellis, H., Shepherd, J. (1977). Cue saliency in faces as assessed by the 'Photofit' technique. *Perception*, 6(3): 263-269. <https://doi.org/10.1068/p060263>
- [11] Dhamecha, T.I., Singh, R., Vatsa, M., Kumar, A. (2014). Recognizing disguised faces: Human and machine evaluation. *PloS One*, 9(7): e99212. <https://doi.org/10.1371/journal.pone.0099212>
- [12] Anwar, A., Raychowdhury, A. (2020). Masked face recognition for secure authentication. *arXiv preprint arXiv:2008.11104*. <https://doi.org/10.48550/arXiv.2008.11104>
- [13] Hariri, W. (2022). Efficient masked face recognition method during the covid-19 pandemic. *Signal, Image and Video Processing*, 16(3): 605-612. <https://doi.org/10.1007/s11760-021-02050-w>
- [14] Kennedy, O., Chiamaka, A.O., Princess, O.I., Julius-Olatunji, O. (2022). Implementation of an embedded Masked Face Recognition System using huskylens system-on-chip module. In *2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON)*, Lagos, Nigeria, pp. 1-7. <https://doi.org/10.1109/NIGERCON54645.2022.9803092>
- [15] Abuzneid, M.A., Mahmood, A. (2018). Enhanced human face recognition using LBPH descriptor, multi-KNN, and back-propagation neural network. *IEEE Access*, 6: 20641-20651.
- [16] Mundial, I.Q., Hassan, M.S.U., Tiwana, M.I., Qureshi, W.S., Alanazi, E. (2020). Towards facial recognition problem in COVID-19 pandemic. In *2020 4rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)*, Medan, Indonesia, pp. 210-214. <https://doi.org/10.1109/ELTICOM50775.2020.9230504>
- [17] Okokpujie, K., Noma-Osaghae, E., John, S., Grace, K.A., Okokpujie, I. (2017). A face recognition attendance system with GSM notification. In *2017 IEEE 3rd International Conference on Electro-Technology for national Development (NIGERCON)*, Owerri, Nigeria, pp. 239-244. <https://doi.org/10.1109/NIGERCON.2017.8281895>
- [18] Jagtap, A.M., Kangale, V., Unune, K., Gosavi, P. (2019). A study of LBPH, eigenface, Fisherface and Haar-like features for face recognition using OpenCV. In *2019*

- International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, pp. 219-224. <https://doi.org/10.1109/ISS1.2019.8907965>
- [19] Jain, A.K., Nandakumar, K., Ross, A. (2016). 50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79: 80-105. <https://doi.org/10.1016/j.patrec.2015.12.013>
- [20] 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. arXiv preprint arXiv:2003.09093. <https://doi.org/10.48550/arXiv.2003.09093>
- [21] Boutrosa, F., Damera, N., Kirchbuchner, F., Kuijper, A. (2021). Unmasking face embeddings by self-restrained triplet loss for accurate masked face recognition. <http://arxiv.org/abs/2103.01716>.
- [22] Ye, Q. (2018). Masked face detection via a novel framework. In 2018 International Conference on Mechanical, Electronic, Control and Automation Engineering (MECAE 2018), pp. 238-243. <https://doi.org/10.2991/mecae-18.2018.137>
- [23] Ejaz, M.S., Islam, M.R., Sifatullah, M., Sarker, A. (2019). Implementation of principal component analysis on masked and non-masked face recognition. In 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, pp. 1-5. <https://doi.org/10.1109/ICASERT.2019.8934543>
- [24] Li, S., Ning, X., Yu, L., Zhang, L., Dong, X., Shi, Y., He, W. (2020). Multi-angle head pose classification when wearing the mask for face recognition under the COVID-19 coronavirus epidemic. In 2020 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), Shenzhen, China, pp. 1-5. <https://doi.org/10.1109/HPBDIS49115.2020.9130585>
- [25] Mundial, I.Q., Hassan, M.S.U., Tiwana, M.I., Qureshi, W.S., Alanazi, E. (2020). Towards facial recognition problem in COVID-19 pandemic. In 2020 4rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM), Medan, Indonesia, pp. 210-214. <https://doi.org/10.1109/ELTICOM50775.2020.9230504>
- [26] Li, Y., Guo, K., Lu, Y., Liu, L. (2021). Cropping and attention based approach for masked face recognition. *Applied Intelligence*, 51: 3012-3025. <https://doi.org/10.1007/s10489-020-02100-9>
- [27] Damer, N., Grebe, J.H., Chen, C., Boutros, F., Kirchbuchner, F., Kuijper, A. (2020). The effect of wearing a mask on face recognition performance: An exploratory study. arXiv preprint arXiv:2007.13521. <https://doi.org/10.48550/arXiv.2007.13521>
- [28] Ramanathan, N., Chellappa, R. (2006). Modeling age progression in young faces. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, pp. 387-394. <https://doi.org/10.1109/CVPR.2006.187>
- [29] Olayiwola, J.O., Badejo, J.A., Okokpujie, K., Awomoyi, M.E. (2023). Lung-related diseases classification using deep convolutional neural network. *Mathematical Modelling of Engineering Problems*, 10(4): 1097. <https://doi.org/10.18280/mmep.100401>
- [30] Isife, O.F., Okokpujie, K., Okokpujie, I.P., Subair, R.E., Vincent, A.A., Awomoyi, M.E. (2023). Development of a malicious network traffic intrusion detection system using deep learning. *International Journal of Safety & Security Engineering*, 13(4): 587. <https://doi.org/10.18280/ijss.130401>
- [31] Vpisarev (2013). `haarcascade_frontalface_default.xml`. https://github.com/opencv/opencv/blob/master/data/haarcascades/haarcascade_frontalface_default.xml, accessed on August 13, 2022.