Enhancing and Securing a Real-Time Embedded Face Recognition System using Raspberry Pi

Dhafer Alhajim [a, *] iD **, Hassoon Salman FAHAMA** [b] iD **, Mohammed T.A** [c] iD

[a]Computer Center, University of Al-Qadisiyah, Al Diwaniyah, 58002, Iraq.Email: dhafer.alhajim@qu.edu.iq

[b]Najaf Technical Institute, Al-Furat Al-Awsat Technical University, Al-Najaf, 31001, Iraq.Email: hassoon.fahama@atu.edu.iq

[c]Department of Computer Technical Engineering, College of Technical Engineering, the Islamic University, Najaf 54001, Iraq.Email: muhammed.t.a@iunajaf.edu.iq

A R T I C L E I N F O

A B S T R A C T

The modern world is full of data of all kinds; however, the vast amount of video and image data available provides the data set needed for facial recognition. Facial recognition is crucial in safety and surveillance systems that analyze visual data and millions of images. Using facial recognition software, a person's identity can be verified through a variety of media images. For facial recognition, there is a variety of algorithms available. The article presents an approach to a face recognition framework using Haar Cascade, a biometric technology in safety and surveillance systems. It investigates combining standard machine learning techniques for face detection and identification with Raspberry Pi face detection, a cost-effective and easy-to-use embedded system. The system detects faces from indirect and direct images, achieving high speed using the latest Raspberry Pi 4 and Python libraries. The work demonstrates a machine-learning-based design method and a complete embedded system. The face detection accuracy is 92%, and the average time is 0.35 compared to the local binary model (LBP). Many facial recognition algorithms on the web and in literature reviews are vulnerable to image attacks. These methods are very effective in identifying faces in webcams, video streams, images, and videos. This system's use of the Raspberry Pi 4 and advanced Python libraries results in fast and accurate real-time face detection. This paper extends the work first presented at the 12th Iranian/Second International Conference on Machine Vision and Image Processing (MVIP).

MSc.

1. Introduction

The Raspberry Pi is a small computer that comes with a miniature mouse and keyboard and can be connected to a computer monitor or television [2]. The latest version of this tiny board has new features that allow it to replace desktop computers [3]. The one-size-fits-all approach of previous versions is no longer applicable to the Raspberry Pi 4, as it is available in one, two, and four gigabytes of RAM. This version is the first to offer an additional gigabyte of

---

∗Corresponding author Dhafer Alhajim

Email addresses: dhafer.alhajim@qu.edu.iq

Communicated by 'sub etitor'

RAM, which enhances the Pi's capabilities, particularly its capacity to run software. Despite this, the Raspberry Pi 4 remains an excellent small DIY device. It was initially created to serve as both an instructional and tinkering tool and has powered everything from small Mars missions to millions of scientific and hacking days in classrooms all over the world [6, 7]. The Raspberry Pi started as a hacker's dream: a low-cost, low power, highly extensible, and hacked PC in bare-board form [4, 5]. Additionally, it is worth noting that face detection, or facial recognition, is a computer technology that uses artificial intelligence (AI) to recognize and identify the faces of people in digital photographs. Face detection and recognition systems have recently gained popularity due to their increased security over finger and typed passwords. A smartphone most likely has a face-unlock feature that will make your life much easier. Face detection is used to keep people safe in many places, including airports, train stations, and roads. A security system's most important tool for protecting against both internal and external threats a number of security-related applications are being developed to protect employees and customers, such as one that can detect criminals and disgruntled ex-employees as they enter the workplace. Face detection and recognition systems can be used to improve the effectiveness of video surveillance by incorporating artificial intelligence into the system. The research goal is to improve the face detection and recognition capabilities of the building's security system, as well as image capture and storage in internal memory or the cloud. According to the findings, cloud storage offers the best performance features, capturing more than 75% of high-quality images [8, 9] Face recognition technology provides real-time tracking and monitoring of individuals in a wide range of fields, including safety [10] and personal security [11]. It has advanced from basic computer vision methods to machine learning (ML) innovations and associated technologies, resulting in ongoing performance improvements. It is currently used as a necessary first step in several critical applications, including facial tracking, analysis, and identification. Face recognition significantly enhances an application's capacity to perform successive tasks. Face detection analyzes facial expressions to determine what portions of a video or image should be focused on to assess one's gender, age, and emotions [12]. Face recognition software employs methods and machine learning to identify human faces in larger images, frequently including non-face elements such as scenery, buildings, and additional human body parts like feet and hands [13]. Face detection algorithms typically begin with human eyes, as they are one of the easiest features to recognize.

The algorithm can attempt to recognize the eyes, nose, mouth, and iris. After deciding that a facial region has been identified, the system performs additional tests to verify that it has recognized a face [4]. To achieve accuracy, the algorithms need to be trained on large datasets containing many images, both positive and negative. Training increases the algorithm's capacity to detect and locate faces in images [14]. Face detection in figures can be difficult due to a variety of factors such as expression, orientation, posture, position, pixel values, the color of the skin, the absence of spectacles or hair on the face, and changes in camera gain, lighting conditions, and image resolution. The use of deep learning for face recognition has advanced in recent years, outperforming traditional computer vision methods significantly [15]. In other words, facial recognition does more than just detect the existence of a human face; it also determines which face it is. The process employs a computer application that compares a digital image of a person's face, usually obtained from the video frame, to photos stored in a database. Photo attacks are a problem for the majority of facial recognition algorithms published online and in academia. These techniques significantly improve the ability to identify and recognize faces in photos, movies, and live webcam feeds. However, they are unable to distinguish between a live face and a photograph of a face. These algorithms cannot recognize faces as they only work on 2D frames. In this paper, we will learn how to perform facial recognition. An embedded system we will use OpenCV to build our face identification system, first detecting faces, and then using machine learning to extract face embedding is from each face, training a system for face recognition on the embedding, and finally using OpenCV to recognize faces in both images and video streams [1].

1.1. Contributions

- Increased Security: Face detection can help improve surveillance efforts and track down terrorists and criminals. Hackers cannot steal or change passwords, which improves security.

- Simple to Use: Quick and easy to use. Because most solutions are cross-platform, face detection and recognition software can be easily integrated into existing security systems.

- Automated Recognition: Previously, manual identification was the only option, but it was slow and error-prone. Face detection improves the efficiency and accuracy of the identification process.

- High Accuracy: Detection is a system weakness. Even though face detection systems are faster and more accurate than manual identification methods, changes in lighting or camera angles can easily fool them.

- Potential invasion of one's personal space: While face detection can help law enforcement catch criminals, the same monitoring can also be utilized to spy on regular citizens. Stringent laws must be implemented to guarantee that technology is used fairly and concerning human privacy rights.

2. Literature Review

The detection system has an extensive variety of applications [15], yet it is most frequently employed in airports to look for people accused of committing crimes and to compare passport scans to personal faces to provide identities. Law enforcement also uses facial recognition software to recognize and apprehend thieves, and such applications have been utilized by several states across the United States to prevent people from getting false identity cards or licenses to drive. Several foreign governments even employ software that recognizes faces to prevent voter fraud. This technology offers multiple benefits over other biometrics methods because it requires no physical contact. The recently developed technology may also record and analyze images of faces from a distance, without the need for contact with a human being or user. As a result, a user cannot effectively impersonate another person. It is also less expensive since it does not have constraints to therapy, unlike other biometrics methods. Although the invention has many benefits, we are going to find that it has a few flaws, which engineers across the world are trying to fix [16]. Individuals can only be identified when the lighting circumstances are favorable, such as when their face turns slightly black or when they appear in insufficient light, as this reduces the application's reliability in recognizing and recognizing the person. The future of recognizing faces appears bright. Surveillance and safety are the main sectors that will be greatly affected by technology. The use of facial recognition is being used to enhance administrative processes in educational institutions, colleges, and even healthcare organizations.

According to Shetty et al. (2021), recognizing faces is one of the most prevalent uses of identification methods [17]. A clear image and the right pose may improve accuracy.

In the face recognition. The Haar cascading classifier outperformed the Local Binary Patterns classifier. Face recognition may be used in this way in the future.

Mayur Surve al. published a method for gathering live images via a camera in 2020 [18]. It then uses a variety of algorithms to identify and distinguish between faces. They also created a GUI that uses a single tap to capture and instill images from the dataset. They used the cascaded Haar method to identify faces in images.

Palanivel N al., [19] developed a module that detects people's faces and generates tendency data. Face Recognition is most reliable when it comes to changes in brightness, posture, expression, and occlusion. The K-means method was used to examine facial characteristics. There is a push to remove the biometric aspect of facial features. The face features are gathered using the K-mean clustered algorithm. SVM is then used to determine the photo's characteristics. It may be highly identifiable while showing fewer characteristics.

Jenifer D. Souza, W.S., al., proposed an image analysis approach for recognizing faces in 2019 [20]. The refined imagery follows a comparison to the successfully battled catalog. To start the process, four modules were employed: image capture, group photo splitting, recognizing a face comparison, and recognition. In 2019, AZM Ehtesham Chowdhury al. proposed a novel webcam prototype for a much more effective attendance analysis [21]. The technique has been extensively evaluated to produce a model that is both vigorous and consistent. The attendance of pupils will be tracked utilizing a method (computerized system). Facial recognition and recognition are used. The most important consideration in determining which modus is best is average accuracy.

In 2019, Nandhini R., Duraimurugan N, et al. [22] created technologies that can identify and recognize students' faces in photos or videos, which are then captured with a camera. Several additional approaches and algorithms have been developed to improve the performance of recognizing faces using Deep Learning. In 2018, Omar Abdel Rahman Salim and others presented an alternative technique for implementing a fully implanted attendance process to face detection. To use this method, you will need a device called a Raspberry Pi installing Raspbian, which is on your computer. A 5-inch screen is linked to a Raspberry Pi using the Raspberry Pi. The digital camera's data is transmitted into the Raspberry Pi for processing. In addition to creating the LBPs, it is internally configurable for handling face recognition. If the face from, for example, the image that was taken matches the face in the model's dataset image, the same doorway will be chosen to open, and attendance will be recorded positively.

In the year 2017, Xiang-Yu et al. [24] proposed a face identification methodology that supports the fast reasoning study to address the same issue of not having a reliable guide to recognizing the squat face in a limitless situation. A

purge primal dataset containing a usual Haar-feature model of classification was utilized. In addition, the procedure is employed to remove the characteristic. Existing monitoring technologies show that Facial recognition could be utilized to achieve personal identification.

C.B. Yuvaraj et al. in (2017) [25] proposed a method for tracking attendance through image processing.

The piece of writing followed standards when depicting the user's face. The Haar feature improved the Viola-Jones method. It will show a specific pupil's face in the picture. It provides database file solutions along with a separated coaching schedule. The Haar cascade improves the look of facial features dramatically. The journey would have been possible if the camera's number were up to date.

Sujay Patole al., (2017) [26], for example, created hardware that uses techniques like justification examination.

It is based on the extraction of features and uses Voila-Jones for identifying faces. PCA is employed to identify faces. Initially, human beings were saved in a database to help with image identification. A viable solution has been discovered, and it will be able to detect the changes that take place within the human face over time.

Haar Cascading is an approach to machine learning that involves training a classifier with a large number of images that are both positive and negative. According to both Viola and Jones [27, 28], here corresponds to the proposed algorithm. Cascade classifiers based on Haar features are used to detect objects. This classifier uses machine learning and a cascade procedure derived from images to look into items in subsequent photographs. Image identification of faces and emotions is additionally successful. The final step is to present both positive and negative pictures to the classifier. The image's characteristics may then be deduced. To find the unique quantity of each feature, subtract the sum of dots in the white box from the sum of cells in the black box. It recognizes the faces of an array of human beings in a variety of settings. Integral images allow you to calculate a Haar-like feature of any dimension in a particular period [29].

Sharma and Savakis (2011) introduced a method for acquiring an effective eye detector by integrating the histogram of oriented gradients (HOG) features with support vector machine (SVM) classifiers [30]. The primary goal of the presented article is to enhance the security platform's face recognition system by identifying individuals and collecting images for archiving.

3. Strategy and Execution

The process for the proposed framework's design has been covered first. The proposed architecture is well suited to the core smart home functionalities of face analysis, recognition, and tracking. The camera on the Raspberry Pi

captures footage as it happens. The video is shown in the yielding window, and a green jump box is drawn until a face is recognized.

3.1. Experimental design

3.1.1. Hardware Design

A video sensor built into the device may use a Logitech webcam to identify and categories faces. When this occurred, the data source was a micro SD card, and a Raspberry Pi 4 version A+ interface was connected to it via a USB cable. A network cable connection is required for the Raspberry Pi, which may be used as a wireless router. The Raspberry Pi has also used the Raspbian stretched operating system, which is present on the 64 GB micro SD card. The connection of the Raspberry Pi, the screen display, and the Logitech webcam is illustrated in Figure 2 The hardware implementation for Model A+ is finished.
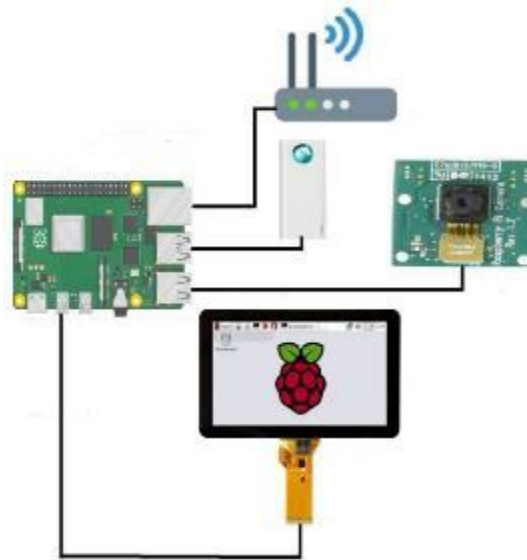

Fig. 1 : Project Schematic [1]

3.1.2. Software Design

With interfaces in C++, Python, and Android, OpenCV (the open-source Computer Vision Library) is cross-platform and available for use on Mac, Linux, Windows, and Android devices. The framework in this project uses OpenCV version 4.5.5, which is more recent than its predecessor is. Face detection algorithm: Faces, pedestrians, and similar objects can all be located using Haar Cascade. emotions captured in an image The Haar Cascade algorithm is trained using a wide range of images, including both positive and negative depictions of human faces in different contexts, along with negative depictions of human bodies in different circumstances. Pictures that do not show humans but can show things like grass and trees. Concurrent feature selection and classifier training are carried out using the ad boost and integral pictures. We retrained Ad boost's classifier. The accuracy of prior training is used to iteratively choose the training set. Figures 2 and 4 indicate that the project's Haar cascade approach makes use of Python 3.9.2 and OpenCV 3.0 for artificial intelligence and image processing. Video streaming makes use of the Haar Cascades method for facial recognition.
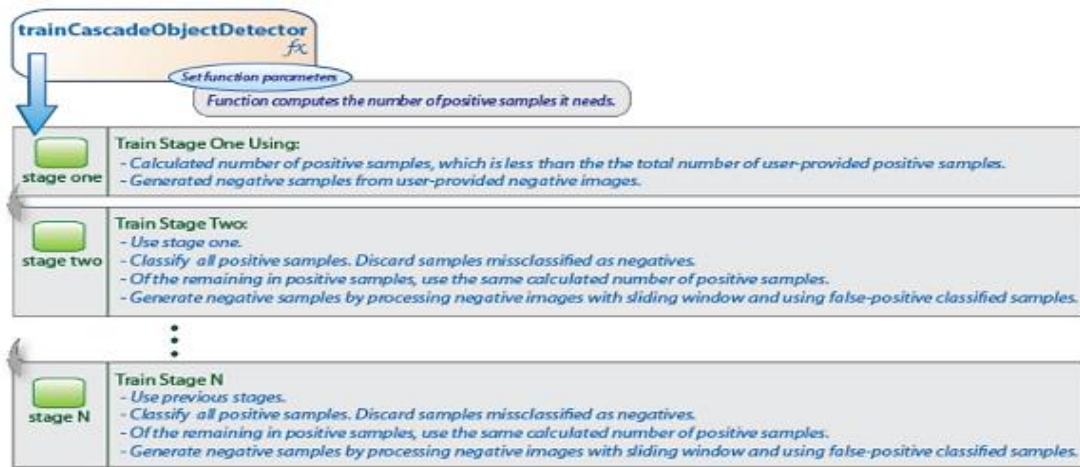
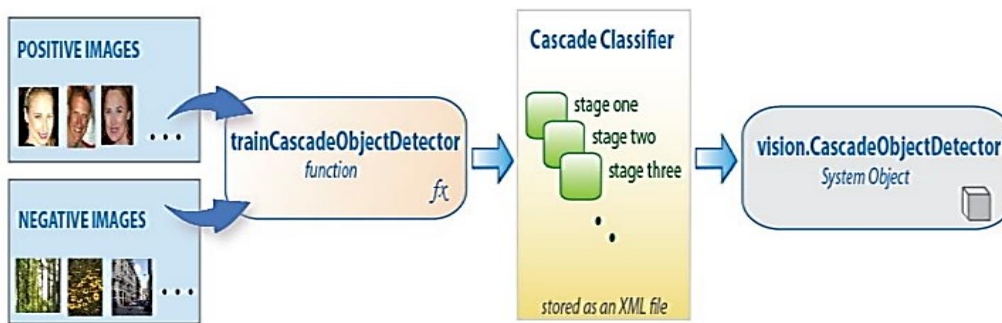**Fig. 2   : The Workings of the Haar Cascade [10].**

Fig. 3   : Training Function with Haar Cascade [11].

OpenCV, a pre-trained image technique, uses pre-trained images using the OpenCV platform. The method known as the cascade approach is based on a detector, and a trainer is part of it, together with pre-trained classifiers like a smile, eyes, and face.

The OpenCV folder contains the Haar Cascade files, which are a kind of classifier that may be employed to identify particular items in a given image. Among OpenCV's many capabilities is the Haar cascade, which was specifically developed for frontal face detection. One machine learning system that may identify objects in pictures or movies is the Haar Cascade. The system needs a dataset that can identify faces before it can train on images; each image should have the faces encoded into 128-dimensional vectors using a histogram of directed gradients.

Used for detection purposes. The flow of the suggested system is shown in figure 4. Figure 4 shows the completed face recognition system, which makes use of a Raspberry Pi 4 and a Raspberry Pi USB camera to record and transmit video to an HDMI LCD screen, a power bank, and an external display. While neither Wi-Fi nor a computer are necessary for face detection in the system, a Wi-Fi connection is necessary for connecting to a private network. Next, the user can decide whether to apply facial recognition to a single frame of video or to save the findings to either a flash drive or the cloud. When the user chooses facial recognition, the stream of video is made available so that the video frame can show not only the names of the building's workers but also their faces. Moreover, the user can identify a suspicious individual in the building because the system cannot recognize them and will display "unknown" in a video frame. When faced with after the user chooses "face detection with storage," the camera takes a picture and saves it if it detects a person.

Preventing photo attacks is the main objective of this research, which aims to create a face-liveness identification method based on eye-blink detection. The algorithm is activated in real-time through a webcam and only shows the name if the subject blinks. The program's operation, explained in simple terms, is as follows:

- Recognize faces in every webcam shot.

- For every face that has been detected, find its eyes.

- Verify the openness or closeness of each eye that has been observed.

If the eyes go from open to close and back to open again, we infer that the user is blinking, and the program shows their name (or, in the instance of a facial recognition door opener, grants them access).

Most helpful are the face encodings, compare faces, and face locations tools. The face location approach uses a convolutional neural network (CNN) and the histogram of orientation gradients (HoG) in combination to identify faces. The HoG approach was selected because of the time restrictions. An image can be encoded into a vector of 128 features using the face encoding function, which is a pre-trained neural network. There ought to be sufficient data in this embedding vector to differentiate between two distinct individuals. Finally yet importantly, the compare faces function measures the separation of two embedding vectors. This will enable the algorithm to identify a face in a camera frame by comparing its embedding vectors to all the faces in our dataset that have been encoded. If possible, have the nearest vector stand in for the same individual [30].

We already established that eventually, we want to be able to detect a liveness in a face by looking for an open-closed-open eye pattern. A convolutional neural network (CNN) was taught to differentiate between open and closed eyes. The LeNet-5 algorithm was selected due to its training on the CEW dataset. About 4,800 24x24-eye photos make it up. We record the eye status of individuals and use our algorithm to forecast their eye health whenever we detect an eye. Therefore, it is quite easy to detect an eye blink using the following code, which searches for a closed-open-closed pattern in the eyes' state history. Our "real" facial recognition algorithm is nearly complete. Our only requirement is a real-time face and eye detection system. We employed a pre-trained Haar-cascade classifier from openCV to do these tasks.

To identify faces, face detectors use a Haar-Cascade face classifier. Classifier for closed or open left and right eyes based on the Haar cascade model. An encyclopaedia of famous names and encodings that includes the eye status history for every name. After resizing the camera feed to speed up calculations, a frame finds faces in the feed and compresses them into 128-dimensional features. We determine the number of matches by comparing this feature with the known face encodings. Find out what the name is. The one with the most matches is selected. The first step is to look for eyeballs in the face boxes. The open-eye detectors are used to identify open eyes first. Since an open-eye detector is unable to identify closed eyes, a '1' is appended to the eye state history if the detector succeeds in determining that both eyes are open. If the first classifiers do not work (for whatever reason, like the eyes being closed or not being recognised), the left and right eye detectors are employed. The detectors are classified by dividing the face into the right and left sides. Finds the eye region and uses the trained model to determine if both eyes are closed or not. If we only see one closed eye, we will assume they are both closed and add a '0' to their status history. The eyes are considered open unless otherwise stated. Finally yet importantly, the blinking feature can tell when someone is blinking their eyes and will show their name if it is true.
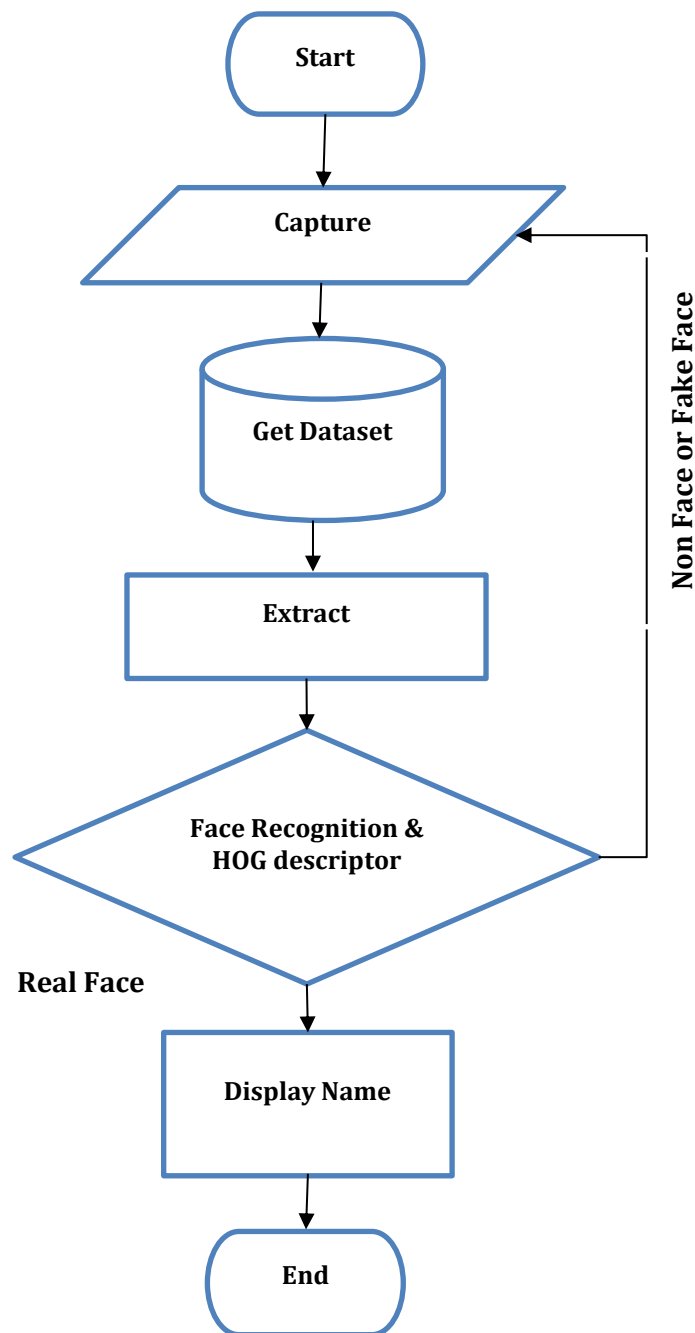
Fig. 4   : The process of facial recognition

Viewed from a different angle, this process illustrates the suggested approach and pseudocode shown in Figure 4 and Algorithm1.

There are several processing steps in the design of this system. An example of a phased operation is the gathering of customer data. We used the Raspberry Pi to build a bonding system for facial recognition. While existing users have access to some services that apply to their personal information, regular users are given access to a handful of basic capabilities. In contrast, video recognition features grant authorized users entry to some premium features.

Algorithm 1: The facial recognition process

1 Gather face data (images) of the persons to be identified from camera (phase1) G
2 Train the recognizer to recognize faces in real time (phase2) T
3 Face recognition versus a new face captured in the future (phase3) F
4 HOG descriptor for face liveness (phase4) H
5 for each G ∈ F & H do
6   Find Haar Cascade and HOG descriptor procedures
7   if G = F & H
8   Real Face
9   else
10   Non Face or Fake Face
11   return G;
12   Display Name of Real-Face
13 End

To identify gender, age, and emotional state from facial expressions, face detection tells face analysis algorithms which parts of a photograph (or video) to prioritize. To create face prints from an image or video and compare them to previously recorded ones, face detection is a necessary component of facial recognition algorithms. In its most fundamental form, face recognition is a tool for identifying people. Our focus here will be on face recognition. To train the classifier, the technique needs a large number of both positive (pictures with faces) and negative (images without faces) images. Then the face is recognised as face-liveness using the HOG approach (phase 4). The next step is to extract its characteristics to distinguish between a familiar face (Phase 1) and a new one (Phase 5). We first captured a picture of the former (Phase 1). To this day, the "Haar Cascade classifier" remains the gold standard for facial recognition algorithms. Our focus here will be on face recognition. The next step was to compile a set of grayscale images for each ID, including the face detection component. The subsequent phase utilised this data set. We collect ten to thirty samples for each ID. Secondly, we "trained" the OpenCV Recognizer using all of the user information in our dataset. The application of a particular OpenCV function achieves this. We have caught a new face on film as we approach the end of our tour. If we have previously recorded and trained the face, we will generate a "prediction" with the person's ID and an index indicating the recognizer's confidence in the match.

4. Equation

The Average time per image can be calculated using the total time processing images (Tav) in data and the total number of images (N), it can be expressed as (see 1):

$$Tav = \frac{Ttotal}{N} \tag{1}$$

To calculate Ttotal , record the start and end times of each image's processing and sum all of the individual processing times (Ti) (see 2).

$$Ttotal = \sum_{i=1}^{N} Ti \tag{2}$$

Detection accuracy is calculated by considering both true positive (TP) means correctly identified faces and false negative (FN) detections, mean faces that the algorithm failed to detect across all tested images. (see 3).

$$Accuracy\ (\%) = \left(\frac{TP}{TP + FN}\right) * 100\% \qquad (3)$$

5. RESULTS

The facial diagnosis is particularly relevant in many fields because of the recent artificial intelligence revolution and the urgent need for many projects to have a facial examination advantage due to the rapid technology. This article describes an established integrated development project that uses a machine learning method to detect faces. Facial recognition is more accurate than fingerprints and biology only through the camera. To address facial accuracy, facial simulation is done directly on a group that picks up the same person, and that data is retrained to accurately identify faces using Haar cascade's algorithm to obtain a facial recognition model through its programming on the Raspberry Pi 4, which has facial detection and recognition and when the embedded system is linked to a standard camera.

From each ID, (80-100) samples are taken, and then all the user data from the dataset is trained using OpenCV Recognizer. In this testing level, the person who used to test is four, and the results are shown below (see Fig 5), which describes a real dataset tested in natural environments from a real-time video Raspberry Pi camera.
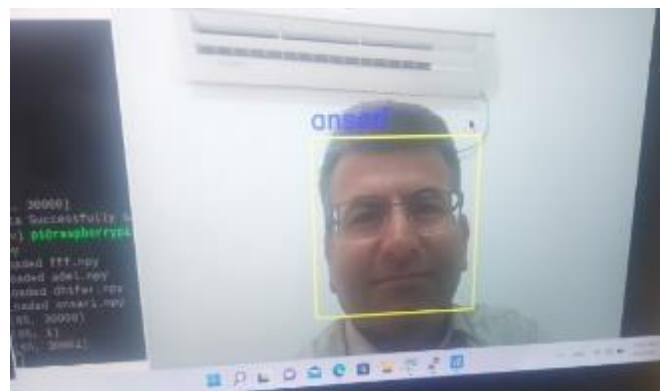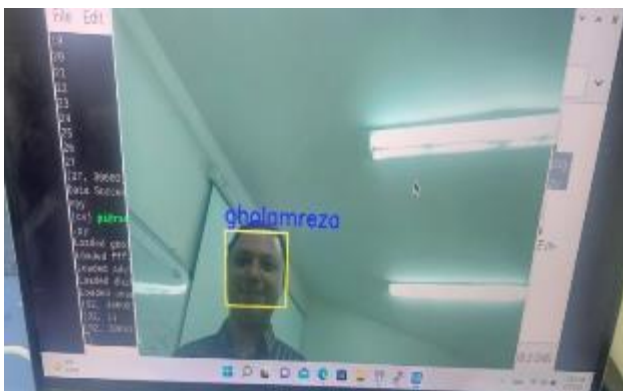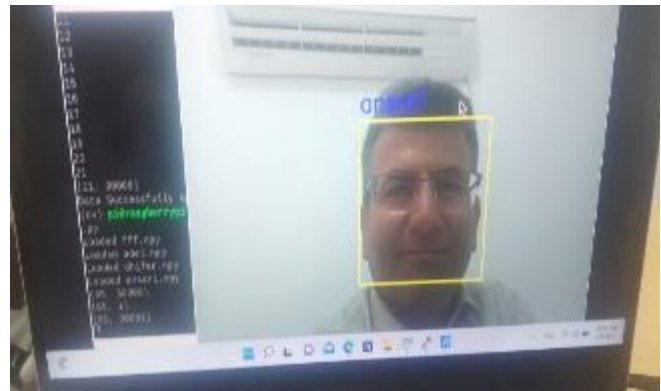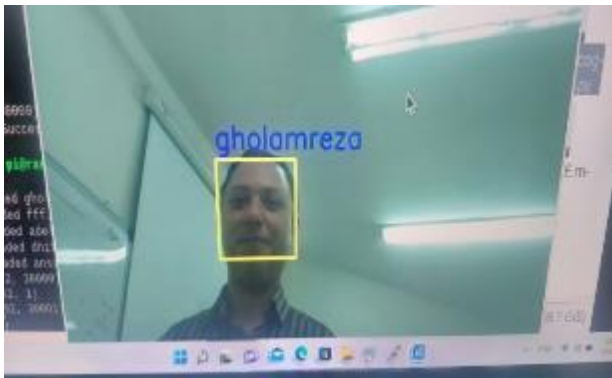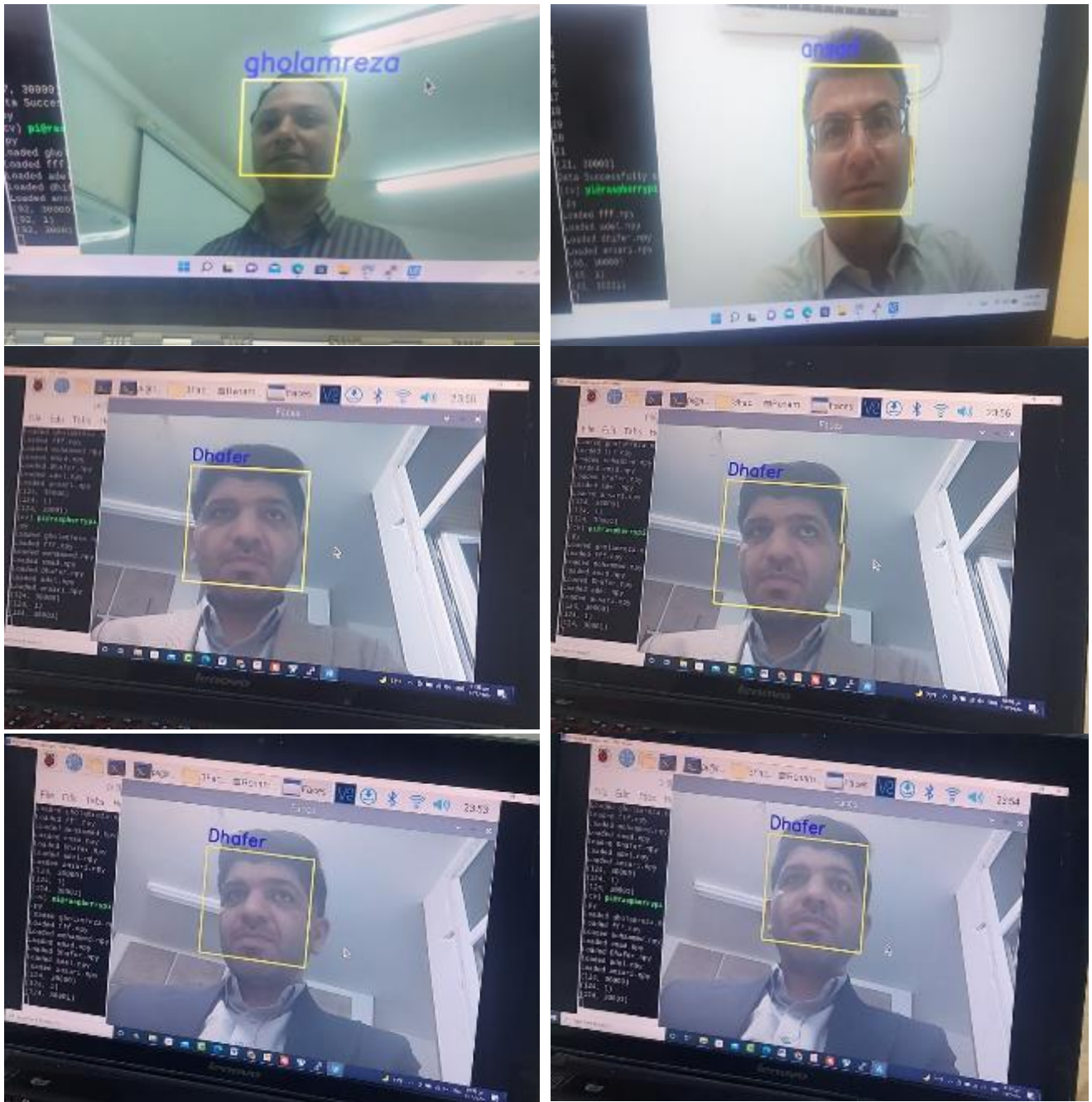
Fig. 5: Software and implementation views of facial detection.

Figure 5 demonstrates the use of Raspbian as the system and the use of the VNC (virtual network computing) server and viewer for remote connections between Raspberry Pi and Windows.

In practice, after running the algorithms (Haar cascade and LBP for face recognition and detection on a video camera), data will be collected that may be utilized to calculate metrics such as average time, detection accuracy, and false positives (see Table 1).

Table 1 - Haar Cascade and Local Binary Pattern Metric

Metric	Haar Cascade	Local Binary Pattern
Average Time (second)	0.35	0.28
Detection Accuracy (%)	92	89
False Positive	3	7

From the results, LBP appears to be fast, taking 0.28 seconds per image on average, compared to 0.35 seconds for Haar Cascade, and the accuracy of the Haar cascade is higher (92%) than LBP's (89%).

We compared our approach to the previous method [1] and found that our proposed algorithm, which includes checking to see if the image is of a real face or a fake one, is safer and faster than the previous method. This makes it a better solution for real-world applications.

6. Conclusion and future work

The face detection system that we have designed using a Raspberry Pi, Python programming, machine learning, and deep learning libraries needs further work regarding the face detection real-time embedded system, which is the front-end for the identification of persons. Switching to a face recognition system can increase the accuracy and security of identification, in addition to being easy to use as a standalone device or by accessing the system via the network. However, it requires more processing time to recognize the person. The recognition phase can be done by using another algorithm besides Haar Cascade/HOG, like Deep Learning/HOG approach.

Acknowledgements

References

[1]   Alhajim, D., Akbarizadeh, G. and Ansari-Asl, K., "FFDR: Design and implementation framework for face detection based on raspberry pi," in 2022 International Conference on Machine Vision and Image Processing (MVIP) (pp. 1-4). IEEE, DOI: 10.1109/MVIP53647.2022.9738788..

[2]   Tabassum, S., & Hossain, "Design and Development of Weather Monitoring and Controlling System for a Smart Agro (Farm)," ICA, vol. 9, pp. 65, 2018. DOI: https://dx.doi.org/10.4236/ica.2018.93005.

[3]   C. Hegde, Z. Jiang, PB. Suresha, J. Zelko, S. Seyedi, M.Smith, ... & G. D. Clifford, " Autotriage-an open source edge computing raspberry pi-based clinical screening system, " medRxiv, 2020. DOI: https://doi.org/10.1101/2020.04.09.20059840.

[4]   S. Pallavi, M. S. Sannidhan, Abhir BhandaryRetrieval of Facial Sketches Using Linguistic Descriptors: An Approach Based on Hierarchical Classification of Facial Attributes, "In Advances in Artificial Intelligence and Data Engineering, pp. 1131-1149, Springer, Singapore, 2021.
DOI: https://doi.org/10.1007/978-981-15-3514-7_84.

[5]   Z. Jiang, S. Yang, H. Huang, X. He, Y. Kong, A. Gao, C. Liu, K. Yan, and
S. Wang, " Programmable liquid crystal display based noise reduced dynamic synthetic coded aperture imaging camera (NoRDS-CAIC), " Optics express, vol. 28, pp. 5221-5238, 2020.   DOI: https://dx.doi.org/10.1364/oe.385547

 [6]   Mahalle, D. Ashlesha, "Artificial Intelligence Based Mask Detection With Thermal Scanning and Hand Sanitization Based Entry System, " Turkish Journal of Computer and Mathematics Education (TURCOMAT), vol. 12,
pp. 299-304, 2021.

[18] Mayur Surve, Priya Joshi, Sujata Jamadar, Minakshi Vharkate, Automatic attendance system using face recognition technique, Int. J. Recent Technol. Eng. (IJRTE).IEEE 9 (1) (May 2020). DOI:
https://doi.org/10.1016/j.gltp.2021.08.044

[Online].   Available: https://www.turcomat.org/index.php/turkbilmat/article/view/8288/6464
[7]   Pajankar, Ashwin, " Introduction to Raspberry Pi, " In Practical Linux with Raspberry Pi OS,. In press, Berkeley, CA, pp. 1-34, 2021.
DOI: https://doi.org/10.1007/978-1-4842-6510-9_1.

[8]   Khairuddin, M. H., S. Shahbudin, and M. Kassim, "A smart building security system with intelligent face detection and recognition," IOP Conference Series: Materials Science and Engineering, Vol. 1176. No. 1. IOP Publishing, 2021. DOI:
https://doi.org/10.1088/1757-899x/1176/1/012030.

[9]   T.N. Nguyen, S. Zeadally, A. Vuduthala, Cyber-physical cloud manufacturing systems with digital-twins, IEEE Internet Comput. (2021). DOI:
https://doi.org/10.1109/MIC.2021.3058921.

[10] J. Wang, , C. Jiang, H. Zhang, , Y. Ren, , K. C. Chen, , and L. Hanzo, " Thirty years of machine learning: The road to Pareto-optimal wireless networks, " IEEE Communications Surveys and Tutorials, vol. 22, pp. 1472-1514, 2020.   DOI:
https://dx.doi.org/10.1109/comst.2020.2965856

[11] H.Wechsler, "Reliable face recognition methods: system design, implementation and evaluation, " Springer Science and Business Media, vol. 7,2009. DOI: https://doi.org/10.1007/978-0-387-38464-1_8.

[12] Pavan, D, " REAL TIME OBJECT DETECTION USING DEEP LEARNING " , 2020.   [Online]. Available: DOI: http://ece.anits.edu.in/2019- 20%20BE%20Project%20REPORTS/CHPS%20_2019-2020%20project%20batch_2.pdf.

[13] Cerf, Moran, Jonathan Harel, Wolfgang Einhäuser, and Christof Koch, " Predicting human gaze using low-level saliency combined with face detection, " Advances in neural information processing systems, vol. 20, pp. 1-7, 2008.   [Online].   Available: DOI: https://proceedings.neurips.cc/paper/2007/file/708f3cf8100d5e71834b1db 77dfa15d6-Paper.pdf.

[14] C.Zhong, ,A. R. Reibman,H. A. Mina, , and A. J. Deering, " Designing a Computer-Vision Application: A Case Study for Hand-Hygiene Assessment in an Open-Room Environment, " Journal of Imaging, vol. 7, pp. 70, 2021.   DOI: https://dx.doi.org/10.3390/jimaging7090170.

[15] D. Wu, ,S. Chen, ,Y. Zhang, ,H. Zhang, ,Q. Wang,J. Li and Z. Shen, " Facial Recognition Intensity in Disease Diagnosis Using Automatic Facial Recognition, " Journal of Personalized Medicine, vol. 11, pp. 1172,2021.DOI: https://dx.doi.org/10.3390/jpm11111172.

[16] D. Boczar,F. R. Avila,R. E. Carter,P. A. Moore,D. Giardi,G.Guliyeva and A. J.Forte, " Using facial recognition tools for health assessment, " Plastic Surgical Nursing, vol. 41, pp. 112-116, 2021.   DOI: https://doi.org/10.1097/PSN.0000000000000410.

[17] Shetty, Anirudha B., and Jeevan Rebeiro. "Facial recognition using Haar cascade and LBP classifiers." Global Transitions Proceedings 2.2 (2021): 330-335.   DOI: https://doi.org/10.1016/j.gltp.2021.08.044

[19] N. Palanivel, S. Aswinkumar, J. Balaji, Automated attendance systems using face recognition by K-means algorithms, International Conference on System Computation Automation and Networking, IEEE, 2019. DIO: https://doi.org/10.1109/ICSCAN.2019.8878790

[20] D Jenif, W S Souza, S Jothi, A Chandrasekar, Automated attendance marking and management system by facial recognition using histogram, 5th International Conference on Advanced Computing & Communication Systems (ICACCS), IEEE, 2019. 334 [5][1]Fig.8Fig.9 A.B. Shetty, Bhoomika, Deeksha et al. Global Transitions Proceedings 2 (2021) 330–335 DOI: https://doi.org/10.1109/ICACCS.2019.8728399

[21] AZM Ehtesham Chowdhury, Omar Khaium Chowdhury, Md.Assaduzzaman Samrat, Md.Zillur Rahman, Tanvir Ahmed, HFDCM: a low-cost machine learning based class attendance monitoring system, 2nd International Conference on Intelligent Comput ing, Instrumentation and   Control Technologies (ICICICT), IEEE, 2019. DOI: https://doi.org/10.1109/ICICICT46008.2019.8993212.

[22] R Nandhini, N Duraimurugan, S.P. Chokkalingam, Face recognition based attendance system, Int. J. Eng. Adv. Technol. (IJEAT) 8 (3S) (February 2019) ISSN: 2249–8958.   DOI: http://dx.doi.org/10.17577/IJERTV9IS060615

[23] Omar Abdul Rhman Salim, Rashidah Funke Olanrewaju, Wasiu Adebayo Balogun, Class attendance management system using face recognition, 7th International Conference on Computer and Communication Engineering (ICCCE), IEEE, 2018. DOI: 10.1109/ICCCE.2018.8539274

[24] X-Y Li, Xiang-Yu, Face recognition supported HOG and quick PCA rule, Euro China Conference on Intelligent Data Analysis and Applications, 2017. DOI: https://doi.org/10.1016/j.gltp.2021.08.044

[25] C.B. Yuvaraj, M. Srikanth, V. Santhosh Kumar, Y.V. Srinivasa Murthy, and Shashidhar G. Koolagudi, "An approach to maintain attendance using image processing techniques", International Conference on Contemporary Computing (IC3), 10-12 August 2017,Noida, India. DIO: https://ieeexplore.ieee.org/document/8284353

[26] Sujay Patole, Yatin Vispute, Automatic attendance system based on face recognition, Int. J. Innov. Res. Sci., Eng. Technol. 6 (August 2017). DOI: https://www.semanticscholar.org/paper/Automatic-Attendance- System-Based-On-Face-Patole-Vispute/159faa1d7b1fa687730fc84f55cf2a58001b320e

[27] Viola, P. and Jones, M.J., 2004. "Robust real-time face detection,". International journal of computer vision, 57(2), pp.137-154. DOI: https://doi.org/10.1023/B:VISI.0000013087.49260.fb

[28] F.M. Javed Mehedi Shamrat, Anup Majumder, Probal Roy Antu, Saykot Kumar Barmon, ItishaNowrin, Rumesh Ranjan "Human Face Recognition Applying Haar Cascade Classifier" International Conference on Pervasive Computing and Social Networking, Salem, Tamil Nadu, India, 19-20, March 2021 DOI: DOI: https://doi.org/10.1007/978-981-16-5640-8_12

[29] Sirajdin Olagoke Adeshina, Haidi Ibrahim, Soo siang teoh and seng chun hoo "custom face classification model for classroom using haar-like and lbp features with their performance comparisons", MDPI 6, Jan 2021. DOI: DOI:https://doi.org/10.3390/electronics10020102

[30] Sharma, Riti, and Andreas Savakis. "Lean histogram of oriented gradients features for effective eye detection." Journal of Electronic Imaging 24.6 (2015): 063007-063007. DOI: https://doi.org/10.1117/1.JEI.24.6.063007