VISITOR CHECKING SYSTEM

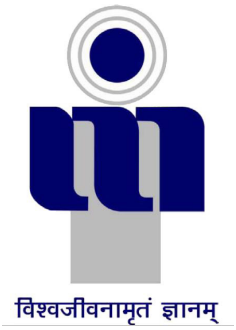
Bachelor of Technology

in

Computer Science and Engineering

by

Bongu Karthik(2021BCS-019)
Gumidelli Chandrahas(2021BCS-027)
Maram Vignesh(2021BCS-039)



ABV INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT
GWALIOR - 474015

JULY 2023

CANDIDATES DECLARATION

We hereby certify that the work, which is being presented in the report, entitled **VISITOR CHECKING SYSTEM** , in partial fulfillment of the requirement for summer project for **Bachelor of Technology in Computer Science and Engineering** and submitted to the institution is an authentic record of our own work carried out during the period *May 2023* to *July 2023* under the supervision of **Prof.Shashikala Tapaswi**. We also cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.
Date:

Name: Signature of the Candidate

Name: Signature of the Candidate

Name: Signature of the Candidate

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Date: Signature of the Supervisor(s)

ABSTRACT

The visitor checking system is a critical component of security and access control in various environments, like educational institutions. Conventional approaches often rely on manual identification methods, such as photo identification cards or manual verification by security personnel, leading to potential inaccuracies and security breaches. In this project, we propose a novel visitor checking system leveraging Siamese networks to enhance visitor identification and authentication.

Siamese networks are a class of deep learning architectures designed for measuring similarity between two inputs. In our system, we utilize Siamese networks to learn feature representations of visitors based on their facial images.The network we used has been trained on a large dataset, the system can then effectively compare incoming visitors' facial features with those in the database for verification and identification.

The proposed visitor checking system consists of two main stages: enrollment and verification. During the enrollment stage, visitors provide their identification details and have their facial images captured by the system. The Siamese network then processes these images and generates a compact and discriminative feature representation, creating an embedded vector for each visitor.

In the verification stage, when a new visitor arrives, their facial image is captured and processed through the trained Siamese network, producing a feature representation. The system then compares this representation with the stored templates to determine the similarity score. If the similarity score exceeds a predefined threshold, the visitor is successfully identified and granted access. Otherwise, the visitor may be flagged for further scrutiny or denied entry.

The proposed visitor checking system offers a scalable and efficient solution for secure access control, benefiting a wide range of environments where visitor identification is crucial. With its ability to continually learn and adapt to new data, the Siamese network-based system holds promising potential for addressing evolving security challenges and ensuring safer and more reliable visitor management practices.

ACKNOWLEDGEMENTS

We are grateful to Prof. Shashikala Tapaswi for allowing us to function independently and explore with ideas. We would like to take this opportunity to express our heartfelt gratitude to her not only for her academic guidance but also for her personal interest in our project and constant support as well as confidence-boosting and motivating sessions that proved extremely beneficial and were instrumental in instilling self-assurance and trust in us. The current work has been nurtured and blossomed mostly as a result of her valuable direction, astute judgment, recommendations, constructive criticism and an eye for perfection. Only because of her tremendous enthusiasm and helpful attitude has the current effort progressed to this point. Finally, I am grateful to the Institution and colleagues whose constant encouragement served to renew my spirit, refocus my attention and energy and helped me in carrying out this work.

Bongu Karthik
Gumidelli Chandrahas
Maram Vignesh

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

CNN	Convolutional Neural Network
API	Application Programming Interface
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets

CHAPTER 1

INTRODUCTION

In recent times, educational institutes have become increasingly conscious of the need to maintain a safe and secure environment for students, staff, and visitors alike. As they strive to strike a balance between maintaining an open and welcoming atmosphere while ensuring the safety of their community, modern technologies offer innovative solutions to address these challenges effectively.

One such technology is the implementation of a Visitor Checking System, a comprehensive approach that combines security and efficiency to manage the entry and exit of visitors on campus. Traditional methods, such as manual sign-ins and ID verification, can be cumbersome, time-consuming, and prone to errors. However, by harnessing the power of Siamese Neural Networks, educational institutes can significantly improve the visitor management process.

Siamese Neural Networks are a specialized class of artificial neural networks designed to compare and recognize similarities or differences between two input data points. This unique architecture makes them particularly well-suited for facial recognition tasks, as they can identify and authenticate individuals with remarkable accuracy and speed.

1.1 Context

In this project, we explore the application of Siamese Neural Networks in educational institutes to create a robust and efficient Visitor Checking System. By leveraging the power of deep learning algorithms, the system will be capable of quickly and accurately verifying visitors' identities, reducing the risk of unauthorized access, and enhancing the overall security posture of the institute.

Throughout this project, we will delve into the technical aspects of Siamese Neural Networks and their application in facial recognition. We will also explore the ethical considerations and privacy concerns associated with implementing such technology on an educational campus.

By presenting a well-rounded examination of the Visitor Checking System based on Siamese Neural Networks, this project aims to provide educational institutes with invaluable insights into how cutting-edge technology can be employed to create a safer and more efficient environment, fostering an optimal learning and working atmosphere for all stakeholders.

1.2 Objectives

The objective is to develop a Machine Learning model to register the visitors who want to enter the campus. This model helps the campus authorities to identify the visitors in case of any emergency situation occurs. The model is trained on different faces using a neural network architecture known as siamese networks.

- Visitor Registration: Create a system that efficiently and properly gathers visitor information, decreasing the need for manual data entry. The system should be able to process visitor information.
- Verification: To verify the identity of a visitor in case of any emergency.
- Reduction of manual processes: By automating the task of registering each visitor entering the campus, the number of mistakes that can occur the whole process is reduced. The time taken to register each visitor is also greatly reduced.

CHAPTER 2

LITERATURE REVIEW

2.1 Background of the Project

Maintaining a secure environment has risen to the top of the priority list for institutions in today's fast-paced world. A visitor checking system is one of the steps done to enhance security in institutions. A visitor checking system is a technological tool created to control and observe the flow of guests coming into and leaving a location.A visitor checking system's main goal is to increase security by precisely identifying people and making sure that only authorized visitors are allowed entry. Visitors must check in using the designated system when they arrive, which records their identity and the reason for their visit. The advantages are

- Through the replacement of outdated manual processes with effective digital workflows, the Visitor Checking System streamlines the entire visitor management process. It streamlines the check-in process for visitors, gathers crucial data,reducing the amount of paperwork and administrative effort.
- The security of your property is considerably improved by a visitor checking system. The technology lowers the risk of security breaches or incidents by precisely identifying and certifying visitors, preventing unauthorized access.

2.2 System Overview

The system architecture contains the following

1)Camera- To capture the faces of visitors while entering and exiting the campus.

2)Face Recognition Library- To identify the faces from the pictures of the visitors.

3)Siamese Network- To identify the visitors.

4)Database- To store the pictures and personal information of the visitors.

2.3 Related Works

2.3.1 Face Detection

Face detection is a computer vision task that involves locating and identifying human faces within images or video frames. The primary goal of face detection is to determine whether there are any faces present in the given input data and, if so, to identify their bounding boxes (rectangular regions that encompass the faces).

Several machine learning models have been developed to perform face detection. Here are a few well-known ones:

Haar Cascades: Haar Cascades is a machine learning-based face detection technique proposed by Viola and Jones in 2001. It uses a series of simple image features called Haar-like features along with a machine learning algorithm (often AdaBoost) to classify regions of an image as either containing a face or not. Despite its age, Haar Cascades remains widely used due to its efficiency and effectiveness for real-time face detection applications.

2.3.2 Face Verification

In recent years, several face recognition techniques have been developed, each with its unique architecture and approach. One notable algorithm is DeepFace, created by Facebook in 2014. It employs a CNN-based architecture with six convolutional layers and a loss function for face verification. Google introduced FaceNet in 2015, which uses a 22-layer CNN architecture combining Zeiler and Fergus, and GoogLeNet. FaceNet applies the triplet loss function for face recognition.

Another significant development came from the Visual Geometry Group (VGG) at Oxford University in 2015, called VGGFace. This architecture is based on VGG

and consists of 18 convolutional layers along with a triplet loss function for enhanced performance.

Carnegie Mellon University researchers proposed another approach using GoogLeNet architecture, followed by the triplet loss function. This model is known as OpenFace, and it optimizes the parameters to enable training with smaller datasets.

These algorithms were evaluated using the Labeled Faces in the Wild (LFW) dataset. DeepFace achieved an accuracy of 97.35% ,aceNet achieved 99.63% accuracy, VGGFace achieved 98.95% accuracy, and OpenFace achieved an accuracy of 92.92%.

CHAPTER 3

METHODOLOGY

Face detection is a computer vision task that involves locating and identifying human faces within images or video frames. The primary goal of face detection is to determine whether there are any faces present in the given input data and, if so, to identify their bounding boxes (rectangular regions that encompass the faces).Several machine learning models have been developed to perform face detection.

3.1 Siamese Networks

Siamese Neural Networks are a specialized class of artificial neural networks designed to compare and recognize similarities or differences between two input data points. The term "Siamese" comes from their architectural resemblance to Siamese twins, as they share the same parameters and architecture while processing two distinct inputs simultaneously. These networks have found significant applications in tasks such as facial recognition, signature verification, one-shot learning, and similarity-based clustering.

The primary goal of Siamese Neural Networks is to learn a meaningful representation of the input data such that similar data points are mapped closer together in the learned feature space, while dissimilar data points are pushed further apart. This makes them highly effective in tasks where measuring similarity or dissimilarity between data samples is essential.

The key components of a Siamese Neural Network include:

1.Shared Architecture: The Siamese network consists of two or more identical subnetworks, sharing the same set of parameters. These sub-networks process their respective input data independently but with the same learned weights and biases.

2.Feature Extraction: Each sub-network in the Siamese architecture extracts relevant features from its input data. In the context of facial recognition, for example, these sub-networks analyze facial images and transform them into a lower-dimensional representation capturing unique facial features.

3. Distance Metric: The learned feature representations from the shared sub-networks are then compared using a distance metric, typically Euclidean distance or cosine similarity. The distance metric quantifies the similarity between two input samples, with smaller distances indicating higher similarity.

4. Loss Function: The Siamese Neural Network is trained using a specific loss function that encourages similar samples to have small distances and dissimilar samples to have large distances. The contrastive loss function and triplet loss function are commonly used in this context.

The training process of Siamese Neural Networks usually involves providing pairs of samples, each labeled as either similar or dissimilar. During training, the network learns to adjust its parameters to minimize the distance between similar samples and maximize the distance between dissimilar ones, effectively learning to distinguish between them.

Siamese Neural Networks have shown great promise in various applications, particularly in scenarios where labeled training data is limited, as they enable effective one-shot learning. Furthermore, their ability to learn meaningful feature representations has made them a popular choice for similarity-based tasks, contributing to enhanced performance and accuracy in numerous domains.

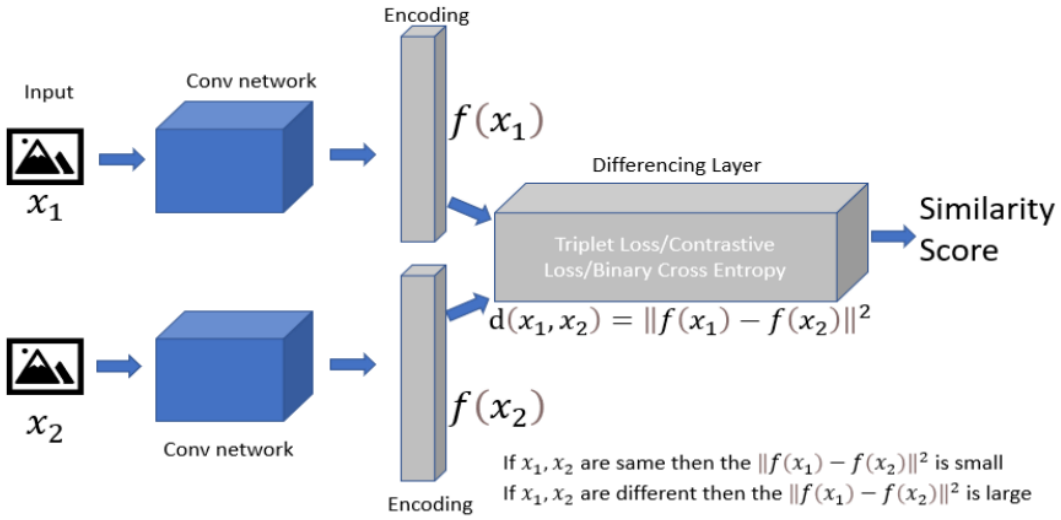


Figure 3.1: Calculation of Similarity Score in Siamese Neural Network

3.2 Haar Cascades

Haar cascades, also known as Haar classifiers or Haar features, are an object detection algorithm used for identifying objects or specific patterns within digital images. Haar cascades are particularly well-suited for detecting faces in images or video streams. They work by applying a series of simple rectangular Haar (fog)-like features to different sub-regions of an image. These features are calculated by subtracting the sum of pixel values in white rectangle regions from the sum in black rectangle regions.

A Haar cascade is a collection of trained Haar classifiers, organized in a specific structure. Each classifier is trained to identify a specific visual pattern or feature, such as an edge, a line, or a specific texture. The cascading aspect of the algorithm refers to the arrangement of classifiers in a cascade, where each subsequent classifier becomes more complex and specific than the previous one.

During detection, the Haar cascade scans an image or video frame using a sliding window technique. At each position, the cascade applies each classifier in a cascade sequentially, and if a classifier fails to detect the pattern, the detection process is stopped for that particular window. This approach allows for efficient computation, as it quickly rejects regions that are unlikely to contain the object of interest.

We used HaarCascade to identify faces in the images taken by the camera and crop the image to only have the face of the person. This makes it easy for the faceNet to easily convert face images into face embeddings without any errors which could be caused due to the background, thus increasing the accuracy of our project.

3.3 FaceNet

A facial recognition system called FaceNet was created by Google researchers using deep learning. It is intended to produce high-dimensional embeddings, sometimes referred to as face vectors or face embeddings, which represent the particular traits and features of a face. Then, these embeddings can be applied to tasks involving faces, such as face verification, grouping, and identification.

Convolutional neural networks (CNNs) are an architecture that FaceNet uses to extract facial information from input photos. In order to learn a mapping from the input image space to a high-dimensional feature space where faces of the same person are close together and faces of different people are far away, the network is trained on a large dataset of face photos.

FaceNet's main goal is to discover an embedding space in which the Euclidean distance between face embeddings can accurately gauge how similar two faces are. FaceNet strives to produce a face representation that is independent of elements like position, lighting, and expression while preserving the fundamental facial features by

encoding faces into a high-dimensional space.

In the context of FaceNet, triplet loss is used to train the network to map face images to a compact embedding space where the embeddings of similar faces are closer together, and the embeddings of dissimilar faces are farther apart.
The triplet loss is defined as follows:

Let
A be the anchor image (a face image from the dataset).
P be a positive image (another image of the same person as the anchor).
N be a negative image (an image of a different person from the anchor).
The triplet loss function is then given as
$L_{\text{triplet}} = max((f(A) - f(P))^{(2)} - (f(A) - f(N))^{(2)} + margin, 0)$
Where

||.|| represents the Euclidean distance between two embeddings.

f(A), f(P), and f(N) are the embeddings (vectors) of the anchor, positive, and negative images, respectively.

The "margin" is a hyperparameter that sets a minimum desired separation between the positive and negative pairs. It prevents the loss from being too small when the embeddings are already well separated.

During training, the siamese network processes three images (anchor, positive, and negative) simultaneously and computes their embeddings. The triplet loss is then computed based on these embeddings and used to update the network's parameters via backpropagation. The training process aims to minimize the triplet loss, which encourages the network to learn discriminative embeddings for face recognition.

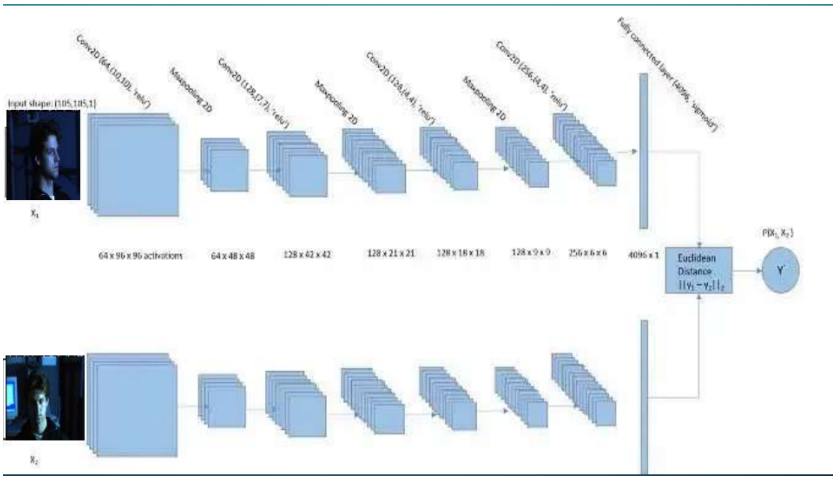


Figure 3.2: Architecture of Siamese Neural Network

3.4 DataSets

3.4.1 Labeled Faces in Wild

Labeled Faces in the Wild, a database of face photographs designed for studying the problem of unconstrained face recognition. The data set contains more than 13,000 images of faces collected from the web. Each face has been labeled with the name of the person pictured.

Figure 3.3: Labeled Faces in Wild Dataset

3.4.2 Youtube Face Database

YouTube Faces Database, a database of face videos designed for studying the problem of unconstrained face recognition in videos.The data set contains 3,425 videosof 1,595 different people.

Figure 3.4: Youtube Face Database

3.5 Functionalities

- facetoembeddings()

  This function is used to convert an image of a visitor to an embedding vector using FaceNet and by removing the cropping out the background from the face using HaarCascade .This function also returns the coordinates of the bounding box of the face which can be used for other purposes like to show the position of the face.

- registerface()

  This function is used to add a visitor into our database which is a dictionary. This function captures the image of the visitor when he enters the institution, crops his face and converts the image into embeddings which is stored in a file present in the local system. We also prompt the user to enter the name of the visitor which is also stored. After we update the dictionary we store the dictionary as a pickle file in the local system.

- deleteface()

  This function allows the user to delete the data of the visitor when he decides to leave the campus premises. This is done by capturing the face of the visitor, identifying the visitor from the database and deleting his records. The vector embedding of the visitor is deleted from the dictionary and this updated dictionary is stored as a pickle file in the local storage system.

- viewvisitors()

  This function is used to check the names of all the visitors which are present in the campus at that particular time. This function could be used to check the number of visitors visiting the campus or to manually check whether the name of the visitor has been registered or not.

- verifyface()

  This function is used to verify the face of the visitor when needed. It is done by capturing the image of the visitor, converting into an embedding and checking the similarity of this embedding with the embeddings present in the database. If the visitor has been registered, his name would be displayed, else it would be displayed as unknown.

3.6 Working

The two main functions of the visaitor checking system is registratioin and identification of visitors. The visitor checking system utilizes Haar cascades and FaceNet for face identification and registration. When a visitor arrives, the system captures their image using a camera and applies the Haar cascade classifier to detect their face. The detected face is then cropped. Next, FaceNet generates a unique fixed-length vector (embedding) for the face image. This embedding, along with the visitor's information, such as name is saved in a database, completing the registration process. During identification, when a visitor approaches the system again, their image is captured and processed similarly. The FaceNet model generates an embedding for the detected face, which is then compared with the stored embeddings of registered visitors. If the similarity between the embeddings exceeds a certain threshold, the system identifies the visitor with a known identity. Based on this identification, the system can perform specific actions, such as granting access to certain areas or providing personalized information to the visitor.

CHAPTER 4

RESULTS

4.1 Summary of Findings

METRIC	DeepFace	DeepfaceID2+	FaceNet
Precision	94.35%	96.57%	97.63%
Recall	91.63%	96.34%	98.18%
Accuracy	91.14%	93.24%	95.12%
Time Taken to Process	0.17sec	0.25sec	0.33sec

Figure 4.1: Comparision Table

4.2 Comparision

4.2.1 Architecture

- DeepFace uses a deep convolutional neural network (CNN) to extract features from faces.
- DeepFaceID2+ and FaceNet uses a deep convolutional neural network (CNN) and siamese networks to extract features from faces.

4.2.2 Training Data

- DeepFace was trained on a dataset of 4.4 million images of faces.
- DeepFaceID2+ and FaceNet were trained on a dataset of 200 million images of faces.

4.2.3 Loss Function

- DeepFace uses a softmax loss function, which is a type of classification loss. The softmax loss function is used to classify images into different classes, such as "positive" or "negative".
- DeepFaceID2+ uses a contrastive loss function, which is a type of metric learning loss. The contrastive loss function is used to learn a distance metric between faces.
- FaceNet uses a triplet loss function, which is a type of metric learning loss. The triplet loss function is similar to the contrastive loss function, but it also takes into account the distance between an anchor face and a negative face. This allows the model to learn a more discriminative distance metric.

4.3 Website

Backend of this model is created using FLASK framework in which the API end points like GET and POST are built to get the data and fetch the data and then display the output in the frontend. The frontend application is made using HTML, CSS, bootstrap where user can choose whether to verify, add, delete a visitor. The face captured in the website goes through our model to give the results required by the user.We have deployed the ML model through Flask. It helps us in running the model on localhost. The server can be further deployed to heroku so it can be accessed anytime instead of limiting it to local system.
Verification Page and the List of visitors can also be viewed.

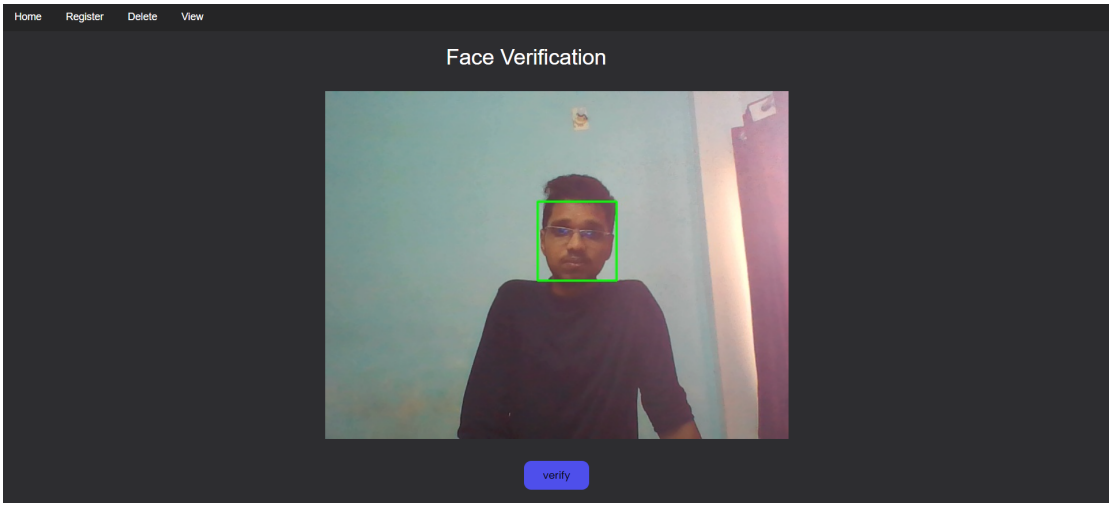


Figure 4.2: Before Verification

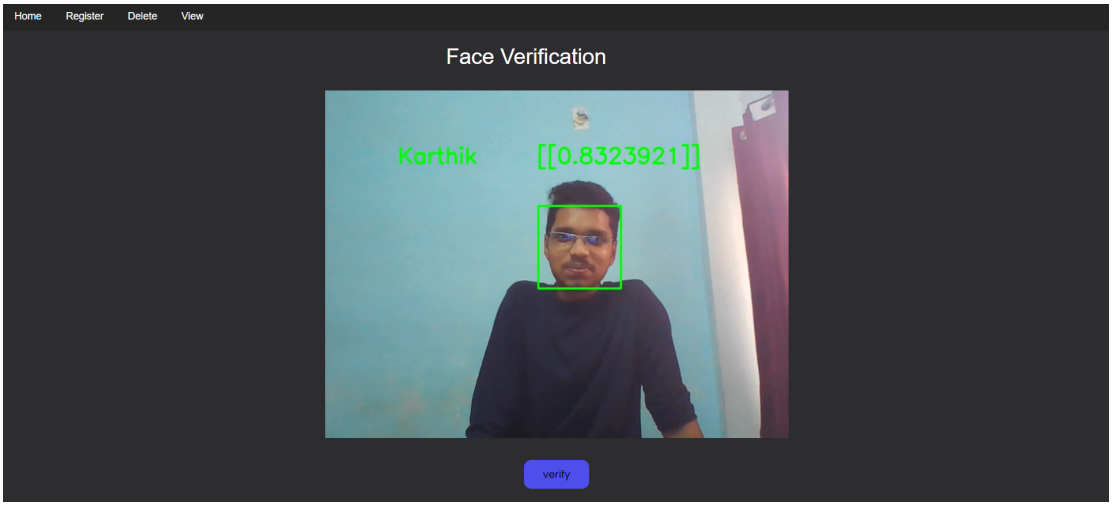
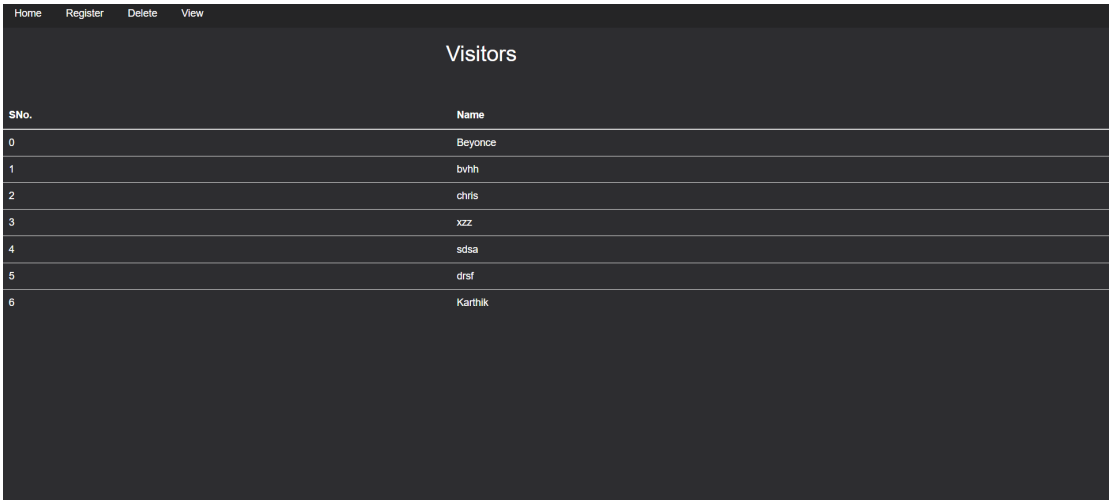


Figure 4.3: After Verification



SNo.	Name
0	Beyonce
1	bvhh
2	chris
3	xzz
4	sdsa
5	drsf
6	Karthik

Figure 4.4: Visitors List

CHAPTER 5

CONCLUSION

5.1 Future Scope

- Integration with Access Control Systems: To automate the entry procedure, integrate the facial recognition system with access control systems. On the basis of known visits, this can entail unlocking doors or providing access to particular locations.
- Database and Scalability: Especially when working with huge volumes of data, optimise the face recognition database for quicker and more effective searches. To ensure scalability and performance, take into account using advanced database administration approaches.
- Multi-modal Biometrics: To improve the overall security and precision of visitor identification, think about combining different biometric modalities, such as facial recognition with fingerprint or iris recognition.

5.2 Limitations

- Limited Pose and Viewpoint Tolerance: Haar Cascade and FaceNet might not handle extreme pose variations or non-frontal views well. If the visitor's face is significantly tilted or viewed from an unusual angle, the system's accuracy may decrease.
- Dependency on Lighting Conditions: The performance of Haar Cascade and FaceNet can be sensitive to lighting conditions. Illumination changes, shadows, or variations in ambient lighting can affect the accuracy of the face detection and recognition processes.

- Single Modality: The visitor checking system using Haar Cascade and FaceNet relies solely on facial features for identification. In cases where the face is obscured, damaged, or not visible due to accessories (e.g., masks) or medical conditions, the system may not be able to identify visitors accurately.

5.3 Novelty

Our team has worked tirelessly to enhance the capability of our visitor checking system, by incorporating multiple features. While there are other models available in the market, our model stands out due to the fact that it incorporated multiple features not present in some models. We use the concept of Siamese neural networks which requires only a single picture of the visitor unlike other systems which might require several of them. The database stores the embedding vector of the picture instead of the picture itself thus requiring less space to function. Since the original facial images are not stored directly, there are fewer privacy concerns related to facial image storage and misuse. Our visitor checking system uses FaceNet model to convert faces into embeddings which has been trained on a lot of images leading to high accuracy of the model which might not be possible by other models.

REFERENCES

[1] *Flask Documentation*: n.d., `https://flask.palletsprojects.com/en/2.3.x/`.

[2] Koch, G., Zemel, R. and Salakhutdinov, R.: 2015, Siamese neural networks for one-shot image recognition, *Proceedings of the 32nd International Conference on Machine Learning (ICML)*.
URL: *https://www.cs.cmu.edu/ rsalakhu/papers/oneshot1.pdf*

[3] Schroff, F., Kalenichenko, D. and Philbin, J.: 2015, Facenet: A unified embedding for face recognition and clustering, *arXiv preprint arXiv:1503.03832* .
URL: *https://arxiv.org/abs/1503.03832*

[4] Viola, P. and Jones, M.: 2001, Rapid object detection using a boosted cascade of simple features, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* .
URL: *https://www.cs.cmu.edu/ efros/courses/LBMV07/Papers/viola-cvpr-01.pdf*