Automatic Attendance Management System

Dr.S.V.N Sreenivasu M.Tech, Ph.D [1], Teja Mannepuli[2], Nandam Pavankumar[3], Nandam VenkatSivaSuneel[4]
[1] Professor, [2, 3 & 4] Student
[1]drsvnsrinivasu@gmail.com, [2] tejamannepuli2019@gmail.com,, [3] nandampavankumar025@gmail.com, [4] suneelsiva2001@gmail.com
Department of Computer Science and Engineering,
Narasaraopeta Engineering College, Narasaraopet, Andhra Pradesh, India

***Abstract*- Facial recognition technology plays a crucial role in various applications, from enhancing security at banks and organizations to streamlining attendance tracking in public gatherings and educational institutions. Traditional methods of attendance marking, such as signatures, names, and biometrics, can be time-consuming and error-prone. To address these challenges, a smart attendance system is proposed, leveraging Deep Learning, Convolutional Neural Networks (CNN), and the OpenCV library in Python for efficient face detection and recognition [1].**

The system utilizes advanced algorithms, including Eigen faces and fisher faces [4], to recognize faces accurately. While deep learning models excel with large datasets, they may not perform optimally with few samples. By comparing input faces with images in the dataset, the system automatically updates recognized names and timestamps into a CSV file, which is then sent to the respective organization's head. Additionally, the system allows users to upload a single photo or a group photo, and it returns matched photos as output using a CNN. This feature enhances the system's flexibility and usability, providing users with a convenient way to identify and track individuals in various scenarios. [2].

Keywords-- CNN algorithm, OpenCV, Attendance Tracking, Deep Learning, Dataset Preparation

1. INTRODUCTION

Facial recognition technology has revolutionized security and attendance tracking systems, offering a more efficient and accurate alternative to traditional[2] methods such as signatures and biometrics. This technology finds extensive application in banks, organizations, public gatherings, and educational institutions, where maintaining security and tracking attendance are paramount.

To enhance the capabilities of existing systems, a smart attendance system is proposed, leveraging Deep Learning, Convolutional Neural Networks (CNN), and the OpenCV[3] library in Python. This system utilizes advanced algorithms like Eigenfaces and Fisherfaces for precise face detection and recognition, ensuring accurate attendance marking.

While deep learning models excel with large datasets, they often struggle with few samples[4]. To overcome this limitation,

the system compares input faces with images in the dataset, automatically updating recognized names and timestamps into a CSV file. Additionally, the system allows users to upload a single photo or a group photo, returning matched photos as output using a CNN[5]. This feature enhances the system's flexibility and usability, providing users with a convenient and reliable method for identifying and tracking individuals.

Facial recognition technology has made significant advancements in recent years, thanks to advancements in deep learning and computer vision. These technologies have enabled the development of more sophisticated algorithms that can accurately identify and track faces in real-time, even in challenging conditions such as low light or occlusions.

One of the key advantages of facial recognition technology is its non-intrusiveness. Unlike traditional biometric methods such as fingerprint or iris scanning, facial recognition[6] can be performed from a distance, making it ideal for use in security systems where user convenience is important. Additionally, facial recognition can be used for more than just security purposes. In retail, for example, it can be used to track customer demographics and behavior, allowing businesses to tailor their marketing strategies more effectively.

Overall, facial recognition technology offers a versatile and efficient solution for a wide range of applications, from security and attendance tracking to marketing and customer analysis. By leveraging the latest advancements in deep learning and computer vision, the proposed smart attendance system aims to provide a reliable and user-friendly solution for accurately identifying and tracking individuals in various settings.

2. LITERATURE SURVEY

P.N. Belhumeur, J.P. Hespanha, & D.J. Kriegman - Known for their work on "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," which discusses the application of PCA and LDA in facial recognition technology to enhance identification accuracy.

Joy Buolamwini & Timnit Gebru - Authored "Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification," highlighting biases in commercial facial recognition technologies and advocating for more inclusive datasets.

Clare Garvie, Alvaro Bedoya, & Jonathan Frankle - Co-authored "The Perpetual Line-Up: Unregulated Police Face Recognition in America," a comprehensive report on the use of facial recognition by law enforcement in the United States, raising concerns about privacy and civil liberties.

Alex Krizhevsky, Ilya Sutskever, & Geoffrey E. Hinton - Their groundbreaking paper "ImageNet Classification with Deep

Convolutional Neural Networks" significantly advanced the field of deep learning, particularly in image recognition tasks.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, & Alexander C. Berg - Introduced the "SSD: Single Shot MultiBox Detector," a method for object detection that includes innovations in facial recognition through improved efficiency and accuracy.

Over the past few years, the modern researchers have been developing solutions for various kinds of problems in which face detection and recognition has been the major task. According the various surveys, there a few models which have been developed. Aiman proposed a model which is used to for face detection. It uses deep learning neural network which with a small data set of images. The images in the data set are applied with Gaussian and poison noise which results in doubling the data set. CNN contains the RELU layer which converts the images into better form and hence a good success rate is achieved.

Your proposal aligns with the trend of using facial recognition for attendance tracking, which has been widely studied and implemented in recent years. The use of deep learning and CNNs for face detection and recognition is well-established in the literature due to their ability to learn complex patterns in images.

The automatic update of recognized names and timestamps into a CSV file aligns with the goal of streamlining attendance tracking processes. This feature has been implemented in various attendance management systems using facial recognition technology.

The system is implemented using the OpenCV library[3] in Python for face detection and recognition. OpenCV provides a robust and efficient framework for image processing tasks, making it ideal for real-time applications such as attendance tracking. The system allows users to upload single or group photos, which are then matched against images in the dataset using the trained CNN model[5].

Allowing users to upload a single photo or a group photo for matching faces is a useful feature that enhances the system's usability. This feature is often implemented using CNNs for face detection and recognition in real-world applications.

My proposed system focuses on using a CNN to extract features from input images (either single or group photos) and then matching these features with images in the dataset to find the best matches. This approach is commonly used in facial recognition systems to compare and identify individuals across different images.

Computer Vision:

Computer vision is a field that focuses on enabling computers to interpret and understand the visual world, including images and videos. It involves the development of algorithms and techniques that allow machines to extract information from visual data, manipulate it, and make decisions based on it. Computer vision is a crucial component of artificial intelligence (AI) and is used in various applications such as self-driving cars, robotics, and photo correction apps.

OpenCV, or Open Source Computer Vision Library, is a powerful open-source library that provides tools and algorithms for computer vision, machine learning, and image processing. It is widely used in both academic research and industrial applications due to its extensive functionality and ease of use. OpenCV enables developers to process images and videos to identify objects, faces, or handwriting. It can also be integrated with other libraries, such as NumPy, to perform complex mathematical operations on image data.

In computer vision, the concept of vector space is often used to represent images and their features. By performing mathematical operations on these features, such as matching, clustering, or classification, computer vision systems can analyze and understand visual data more effectively.

3. CNN ARCHITECTURE

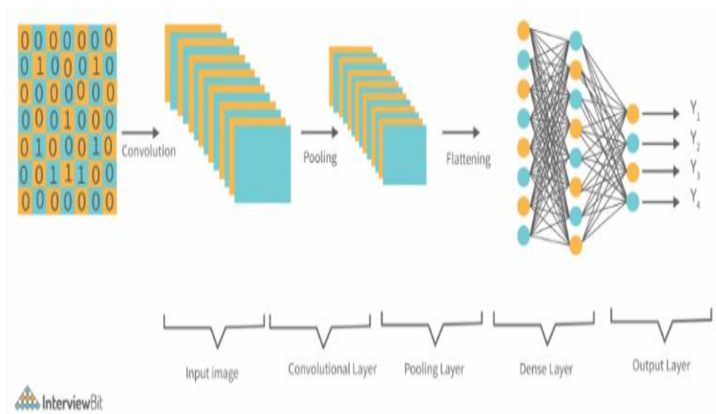


Fig 1: CNN Architecture

Fig1 shows the typical Convolutional Neural Network (CNN) architecture consists of several key layers. The input layer receives the input image or feature map, which is then passed through a series of convolutional layers. These convolutional layers apply filters to extract features from the input, such as edges, textures, or shapes. Each convolutional layer is typically followed by an activation layer, which introduces non-linearity into the network using activation functions like ReLU (Rectified Linear Unit).

After the convolutional layers, the feature map is often passed through a pooling layer. The pooling layer reduces the spatial dimensions of the feature map, reducing computational complexity while retaining important features. Finally, the output from the convolutional and pooling layers is flattened and passed through one or more fully connected layers. These fully connected layers connect every neuron in one layer to every neuron in the next layer, performing classification based on the extracted features.

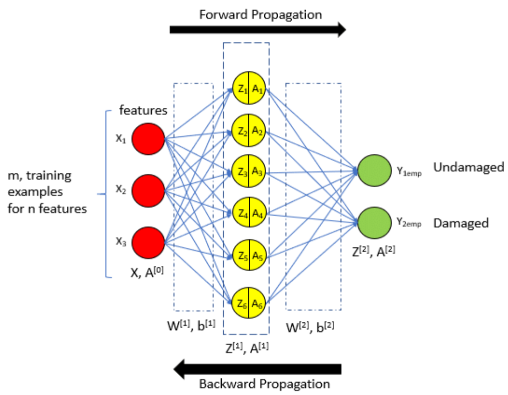


Fig-2: Input 2Layer

a. **Input 2Layer:** Fig-2 shows layer represents the input image or a batch of images. Each image is typically represented as a 3D array (height x width x channels), where the channels can be RGB (3 channels) or grayscale (1 channel).

b. **Convolutional Layer:** The convolutional layer applies a set of filters to the input image(s). Each filter is a small matrix that slides over the input image(s) to produce a feature map. These filters help detect various features in the images, such as edges, textures, or patterns.

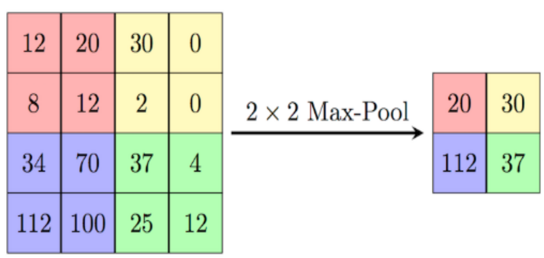


Fig-3: Pooling Layer

c. **Pooling Layer:** Fig-3 shows pooling layer reduces the spatial dimensions of the feature maps generated by the convolutional layer. It helps in reducing the computational complexity of the network and makes the model more robust to variations in the input images. Common pooling operations include max pooling and average pooling.

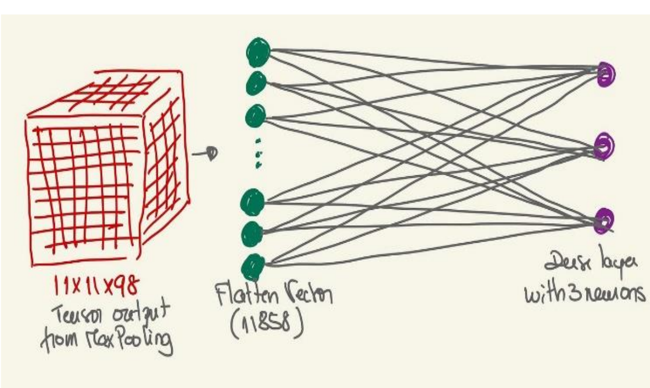


Fig-4: Dense Layer

d. **Dense (Fully Connected) Layer:** Fig-4 shows dense layer is a traditional neural network layer where each neuron is connected to every neuron in the previous layer. This layer is typically used at the end of the CNN architecture to classify the features extracted from the previous layers into different classes

e. **Output Layer:** The output layer produces the final output of the CNN, which could be class probabilities (in the case of classification tasks) or continuous values (in the case of regression tasks). The number of neurons in the output layer depends on the number of classes in the classification task or the number of output values in the regression task.

CNN architecture consists of layers designed to extract features from input images and classify them into different categories. The convolutional and pooling layers extract and reduce features, while the dense layer performs the classification based on these features, with the output layer providing the final prediction. [17].

WORK FLOW:

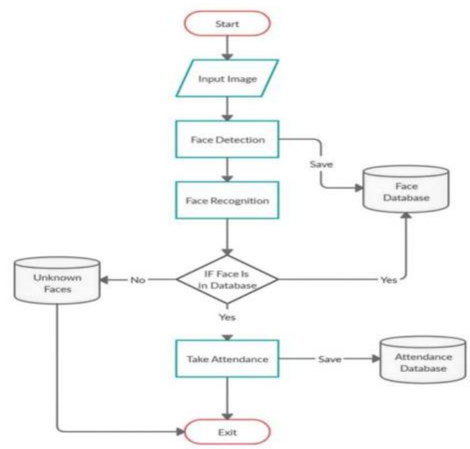


Fig 5: Work of Face Recognition

The fig 5 shows the implement face recognition using a Convolutional Neural Network (CNN) for the smart attendance system, the process can be broken down into several key steps. First, gather a dataset of face images with corresponding labels, such as names or IDs, for training the model. Preprocess the images by resizing them to a uniform size, converting them to grayscale, and normalizing pixel values.

Next, design a CNN architecture suitable for face recognition, including convolutional, activation, pooling, and fully connected layers. Train the CNN model using the preprocessed dataset to learn features for face recognition.

For implementation, use OpenCV for face detection in input images or frames from a camera. Pass the detected faces through the trained CNN model to recognize faces based on learned features. Match recognized faces with the database and update attendance records accordingly.

Develop a user-friendly interface for interaction, including options for registration, attendance taking, and help. Test the system with different scenarios to ensure accuracy and reliability before deploying it for real-world usage.

IMPLEMENTATION OUTPUT'S:

OpenCV is a powerful tool for image processing and computer vision tasks, making it well-suited for developing an attendance management system based on facial recognition.

Using Flask for my interface is a good as it will allow you to create a web-based interface for your attendance management system. Flask is a lightweight and flexible framework, making it easy to integrate with my OpenCV-based backend.

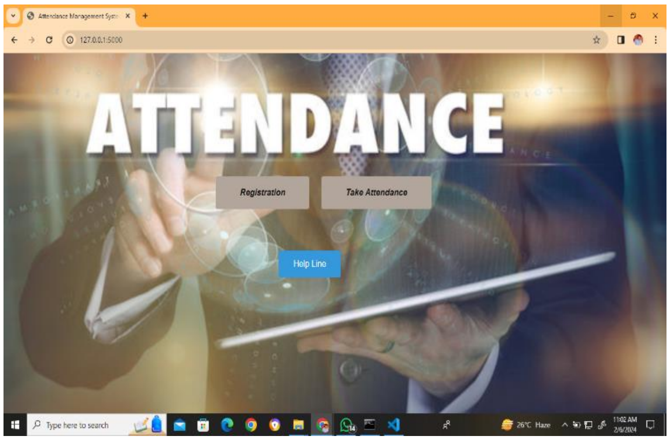


Fig 6: web interface

The fig 6 The web interface of the smart attendance system contains three main features: Registration, Attendance Taking, and Help.

Registration: This feature allows new users to register themselves in the system by providing their basic information such as name, ID, and possibly a photograph. This information is then stored in the system's database for future reference.

Attendance Taking: The core functionality of the system, this feature enables users to take attendance by capturing faces using a camera or webcam. The system then processes these faces using advanced algorithms to identify individuals and mark their attendance.

Help: This feature provides users with assistance and guidance on how to use the system effectively. It may include tutorials, FAQs, or contact information for technical support.

The web interface's user-friendly design and intuitive layout make it easy for users to navigate between these features and efficiently manage attendance records.

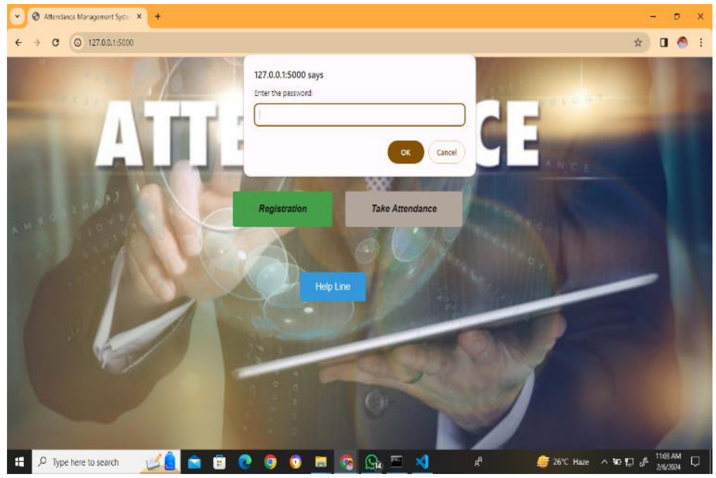


Fig 7: Authentication

The fig 7 shows the user clicks on the "Registration" button in the web interface, a popup message appears, prompting them to enter a password. This feature ensures that only authorized users, such as the admin or main person, can register new users. If the admin enters the correct password, the system navigates to the registration page, where new users can be added to the system. However, if the admin enters an incorrect password, the system displays a message indicating that the password is incorrect and prompts the admin to enter the correct password. This additional layer of security helps protect the system from unauthorized access and ensures that only authorized personnel can manage user registrations.

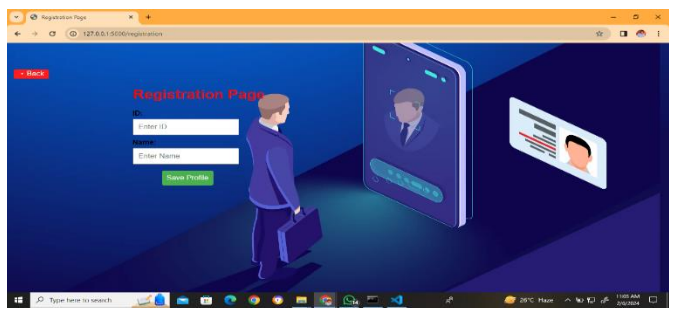


Fig 8: User Registration

The fig 8 In the registration process of the smart attendance system, users provide their names and IDs. Upon clicking the "Save Profile" button, the system opens the camera to capture several face photos in different poses and expressions. These photos are then processed using the implemented algorithms to extract facial features and create a unique profile for the user. The photos are saved in a designated folder for future reference. Finally, a message is displayed on the interface confirming that the profile has been successfully saved, ensuring that the registration process is complete and the user's information is securely stored in the system.

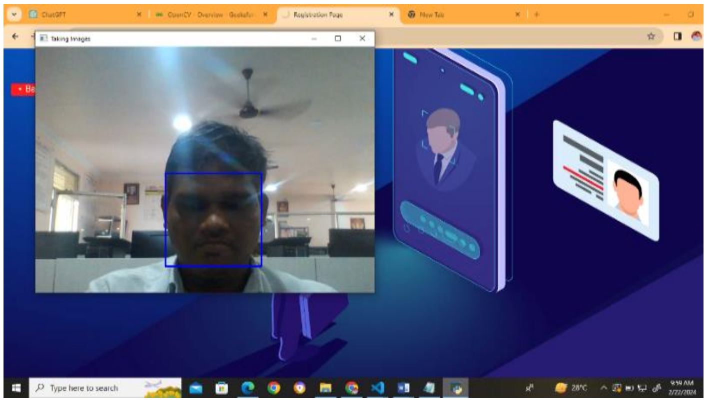


Fig 9: Image’s capturing

Fig 9 shows the after saving the image, the system will perform face recognition to take attendance. It will open and identify the faces, displaying the person based on our data otherwise, it will show that the face was not recognized. After recognizing faces, the camera will turn off automatically, and the recognized faces' names and IDs will be stored in one Excel sheet, including the time and date. When we click the download button, the Excel sheet will be downloaded to PC.

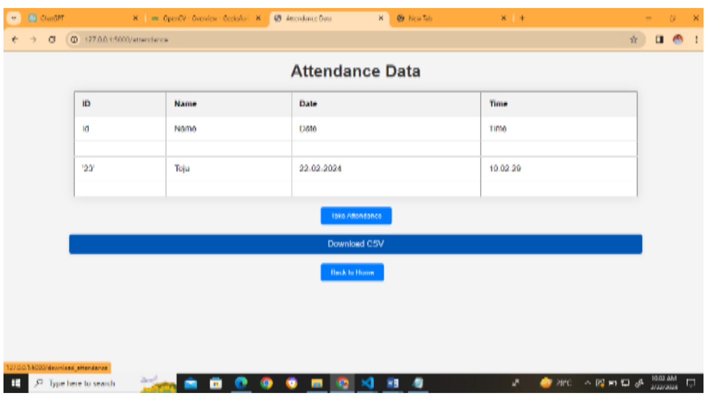


Fig 10: capturing attendance and save attendance

Fig 10 shows the process of capturing attendance and saving it in an Excel sheet. The smart attendance system captures the faces of individuals using a camera or webcam. These faces are then processed using the implemented algorithms, such as Eigenfaces and Fisherfaces, for precise detection and recognition. Once the faces are recognized, the system automatically updates the attendance records in an Excel sheet, along with timestamps for each entry.

Saving attendance in an Excel sheet provides a convenient and accessible way to manage and track attendance records. It allows for easy organization and analysis of attendance data, making it simpler for administrators to monitor attendance trends and identify any irregularities. Moreover, the Excel sheet can be easily exported or integrated with other software for further processing or reporting purposes.

The integration of facial recognition technology with Excel for attendance management enhances the efficiency and accuracy of attendance tracking systems, making them more reliable and user-friendly.

WORK FLOW OF IMAGE MATCHING:

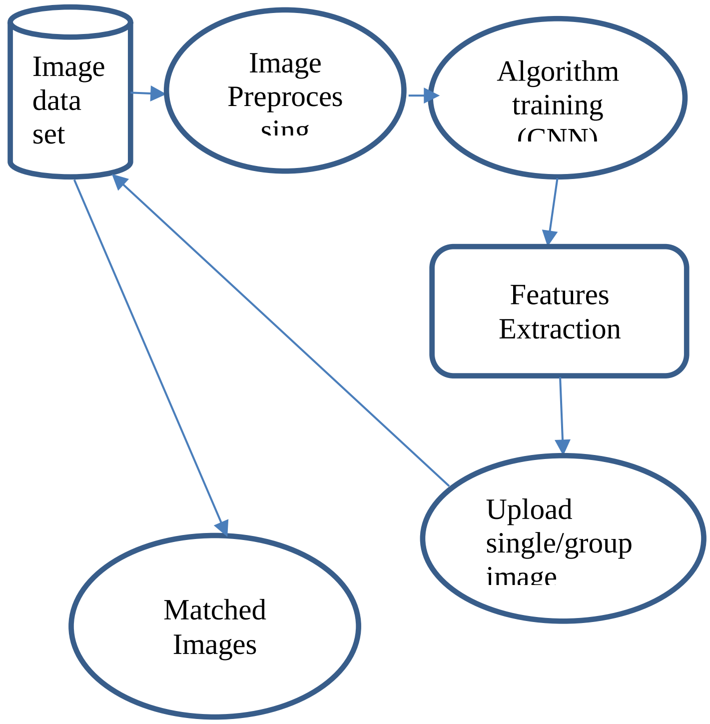


Fig 11: Matching image work flow

IMAGE PREPROCESSING:

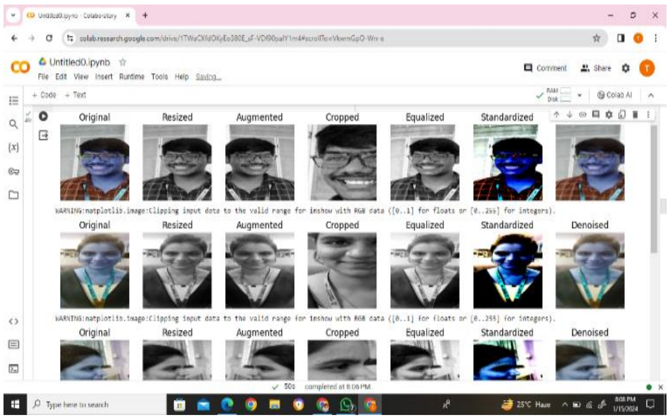
Fig 8 shows the using deep learning preprocessing methods .we perform the image preprocessing on image dataset. Like

A. **Resizing:** This refers to changing the dimensions of an image. It can be done to fit a certain size requirement or to prepare images for processing in a machine learning model, which often requires images to be of a consistent size.

B. **Normalization:** Normalization is the process of adjusting the values of an image to a standard range. In the case of images, this usually means scaling the pixel values to be between 0 and 1, or -1 and 1. Normalization helps in reducing the variation in pixel values and can improve the performance of machine learning models [17].

C. **Rescale:** Rescaling is similar to normalization but involves scaling the pixel values by a certain factor. For example, rescaling by 1/255 would convert pixel values from the range 0-255 to the range 0-1.

D. **Augmentation:** Data augmentation is a technique used to artificially expand the size of a dataset by creating modified versions of images in the dataset. This can include flipping, rotating, cropping, or changing the brightness of images. Data augmentation is commonly used in deep learning to improve the robustness of models and prevent overfitting.



4. TRAIN CNN MODEL

During the training of a Convolutional Neural Network (CNN) algorithm, the model's accuracy typically improves over time as it learns from the dataset. This improvement is often visualized in a figure where the x-axis represents the number of epochs or iterations over the training dataset, and the y-axis represents the model's accuracy.

In your case, the figure below shows the accuracy trend of the CNN algorithm during training, with the x-axis indicating the number of epochs and the y-axis indicating the accuracy percentage. The figure illustrates how the model's accuracy gradually increases with each epoch, indicating that the model is learning and improving its performance over time.
The final accuracy achieved, such as 71.34%, reflects the model's ability to correctly classify images in the test set after training.

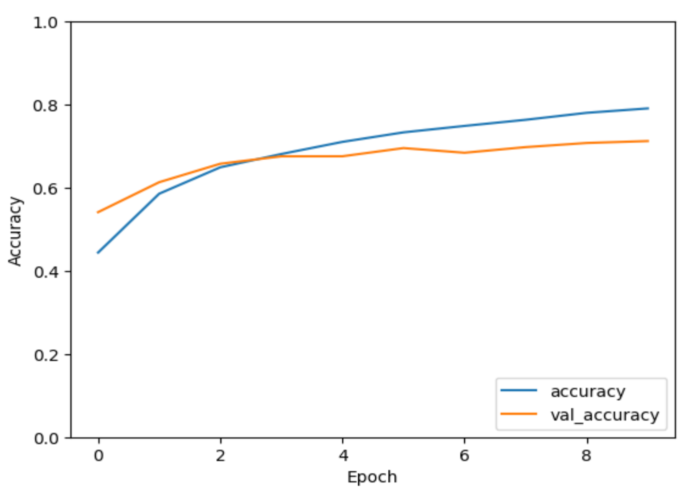


Fig 12: CNN training Accuracy graph

Understanding the accuracy trend is crucial for evaluating the performance of the CNN algorithm[17] and identifying potential issues such as overfitting or under fitting. Fig 8 shows analyzing the accuracy trend, you can make informed decisions about the model architecture, hyper parameters, and training process to optimize the model's performance.

To save the trained model with the name "best_match.h5," you can use the model's serialization functionality. This allows you to save the model's architecture, weights, and training configuration to a file, which can then be loaded later for inference or further training. It's important to use separate training and test datasets to evaluate the model's performance accurately. The training dataset is used to train the model, while the test dataset is used to evaluate its performance on unseen data.

RESULT:

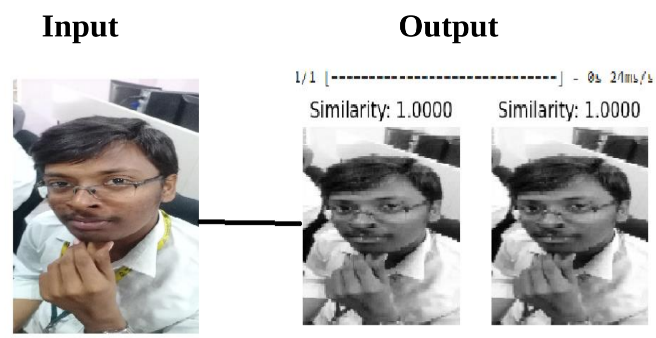


Fig 13: Single photo matching

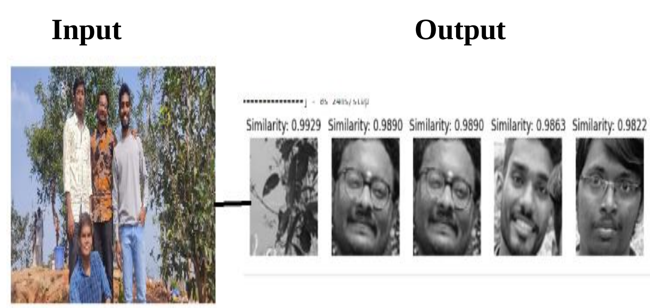


Fig 14: Group photo Matching

Fig 13, 14 shows the result of the single photo or group photo matching using the CNN algorithm is displayed in the smart attendance system's interface. When a user uploads a single photo or a group photo, the system processes the images using the Convolutional Neural Network (CNN) algorithm to identify and match faces with those in the database.

For a single photo, the system compares the uploaded image with the database of registered faces and returns the matched face along with the user's information, such as name and ID. The result is displayed on the interface, indicating whether the matching was successful or not.

In the case of a group photo, the system detects and recognizes multiple faces in the image. It then matches each face with the database and displays the results individually for each recognized face.

This feature allows for efficient and accurate identification of individuals in group photos, making it easier to track attendance in events or gatherings where multiple people are present.

CONCLUSION AND FUTURE SCOPE:

The proposed model has the capability of detecting and recognizing different faces and images from the camera. The face recognized matching purpose we using another module OpenCV. It's a powerful library for computer vision tasks, and it should work well for your facial recognition system.

The data set which contains the images are pre-trained and tested using deep learning so that the input images would be well detected. This method is secure enough, reliable and available for use.

Further CNN adds robustness to the model and using this approach of training data, 71.34% recognition rate has been achieved. Deep learning has advantage over machine learning for other face recognition techniques.

The resultant of this entire process is nothing but creating an attendance marking system in which the unique id, name and some more details of the recognized faces could be entered automatically into a CSV file.

In further updating this attendance system can be taken towards web development by creating a website and marking the attendance of the people automatically into the website of the organization so that there would be no need to update or mark attendance manually.

Additionally, the model's ability to find the best match photos from the dataset when uploading single or group photos enhances its utility and effectiveness.

Presently this project is developed using Flask python and is successfully running on web. In the future upation, a mobile application will be developed in which each and every student are given access with unique login details so that they can track their status of attendance from anywhere round the globe.

5. REFERENCES

[1] H. Yang and X. Han, "Face Recognition Attendance System Based on Real-Time Video Processing", *IEEE Access*, vol. 8, pp. 159143-159150, 2020.

[2] D. M. Prasanna and C. G. Reddy, "Development of Real Time Face Recognition System Using OpenCV", *International Research Journal of Engineering and Technology*.

[3] Siti Ummi Masruroh et al., "NFC Based Mobile Attendence System with Facial Authorization on Raspberry Pi and Cloud Server", *2018 6th International Conference on Cyber and IT Service Management (CITSM)*, pp. 1-6, 2018.

[4] J. Alghamdi, R. Alharthi, R. Alghamdi, W. Alsubaie, R. Alsubaie, D. Alqahtani, et al., "A survey on face recognition algorithms", *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS)*, pp. 1-5, 2020.

[5] S. M. Anzar, N. P. Subheesh, A. Panthakkan, S. Malayil and H. A. Ahmad, "Random interval attendance management system (RIAMS): A novel multimodal approach for post-COVID virtual learning", *IEEE Access*, vol. 9, pp. 91001-91016, 2021.

[6] J. W. D'Souza, S. Jothi and A. Chandrasekar, "Automated attendance marking and management system by facial recognition using histogram", *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, pp. 66-69, 2019.

[7] T. Sharanya, U. Kasturi, P. Sucharith, T. Mahesh and V. Dhivya, "Online attendance using facial recognition", *Int. J. Eng. Res*, vol. 9, no. 06, pp. 202-207, 2020.

[8] N. Yasmeen, K. K. R. C and S. Doss, "Intelligent systems powered hourly attendance capturing system", *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 1536-1541, 2023.

[9] R. S. K. A. A. R and C. K. Kannan, "Automatic attendance monitoring system using LBPH and HAAR algorithm", *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, pp. 1604-1608, 2023.

[10] K. Vignesh, A. Abirami, P. Manideep, K. Vasu, K. Rakesh and S. V. Vardhan, "Smart attendance system using deep learning", *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 1081-1087, 2023.

[11] NING ZHU, ZEKUAN YU and CAIXIA KOU, "A New Deep Neural Architecture Search Pipeline for Face Recognition", *IEEE Access*, 2020.

[12] ASMA BAOBAID, MAHMOUD MERIBOUT, VARUN KUMAR TIWARI and JUAN PABLO PENA, "Hardware Accelerators for Real-Time Face Recognition: A Survey", *IEEE Access*, 2022.

[13] Jian ZHAO, Chao ZHANG, Shunli ZHANG, Tingting LU, Weiwen SU and Jian JIA, "Pre-detection and binary-wordbook meager representation grounded face recognition algorithm in non-sufficient training samples", *Journal of Systems Engineering and Electronics*, 2018.

[14] ZAKIR KHAN, ARIF IQBAL UMAR, SYED HAMAD SHIRAZI, MUHAMMAD SHAHZAD, MUHAMMAD ASSAM, MUHAMMAD TAREK I. M. EL-WAKAD, et al., "Face Recognition via Multi-LeveI3D-GAN Colorization", *IEEE Access*, 2022.

[15] M. SAMI ZITOUNI, PETER LEE, UICHIN LEE, LEONTIOS J. HADJILEONTIADIS and AHSAN KHANDOKER, "Privacy Aware Affective State Recognition from Visual Data", *IEEE Access*, 2022.

[16] Ashish Yadav, Aman Sharma and Sudeept Singh Yadav, "Attendance Management System Based on Face Recognition Using Haar-Cascade", *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*.

[17] Syam Kakarla, Priyaranjan Gangula, M. Sai Rahul, C. Sai Charan Singh and T. Hitendra Sarma, "IEEE-HYDCON Smart Attendance Management System Based on Face Recognition Using CNN", *2020 IEEE-HYDCON*, 2020.