Automated Attendance System Based on Face Recognition Using Deep Learning Techniques

1.Dr. M. Satyanarayana, Professor, 2.MS.Hashita Pusapati, 3.Mr.Sappati Sai Satwik
4.Ms. Suru Sirishma , 5.Mr. I. Dillep Kumar
Department of Computer Science Lendi Institute of Engineering and technology
JNTU GV, Vizianagaram,Andhra Pradesh,India
satyanarayana.m@lendi.edu.in, hashitapusapati012@gmail.com, sappatisaisatwik@gmail.com, sirishmasiri527@gmail.com, Ungaraladileepkumar15@gmail.com

***Abstract*—Effective attendance is essential to the education process. Teachers and school administrators need to know how much time each student spends in the classroom. Because of this, many schools use automated attendance systems (AASs). These systems save time and space by enabling schools to register and track students without the need for physical identification cards or teachers. To keep track of students, teachers can use smart cards that monitor attendance records. Biometric technology allows schools to verify student attendance by comparing their biographical data to student identification card entries. A good attendance system helps school administrators maintain proper attendance records for each child. However, no system works effectively when only some required IDs are present among the school's student population. Implementing AASs requires proper planning and implementation- but it will undoubtedly save time and space in your school's classroom when properly executed and utilized.**

Keywords—Attendance System, Face Recognition, Machine Learning, AASs, Automated Attendance

I. Introduction

Face recognition is a technology that enables computers to identify people based on their facial features. A face recognition system uses a camera to capture images of faces and compares them to stored images of known individuals. If the computer finds a match, it can then determine who is present at the time of image capture. A smart attendance system combines both face recognition and an attendance system. It works similarly to a traditional attendance system, except that it can recognize students' faces instead of manually logging their attendance. The system can send out reminders to students if they fail to appear for a class. This is one way we can ensure students attend class.

This software can be used to record the attendance of students in class automatically. It does this by scanning the faces of students in the classroom and comparing them to a database of recorded attendance. This method is ideal because it eliminates the need for teachers to write down student attendance manually. It also eliminates the risks associated with collecting handwritten records. Plus, it reduces the amount of time required to verify student attendance. Institutions can save money and time by implementing an automated system for recording student attendance.

Doing so frees up administrative staff to focus on more important tasks, such as grading assignments and filing paperwork. Furthermore, face recognition software helps schools identify repeat absentees. This allows them to provide individualized treatment to children who consistently miss class. Over time, this could reduce problematic absenteeism behavior among specific demographics of students.

It has several uses in schools and colleges, primarily as an absenteeism management tool and a discipline strategy tool. In the first case, it helps teachers identify absent students in a timely manner. It can also notify teachers when absent students return to class so they can give them extra attention and feedback. As a discipline strategy tool, face recognition is used in schools to enforce uniform behavior among students. For example, it can be used to track whether a child attends school or not, what they do while at school, and whom they hang out with while at school. All this information is gathered so officials can keep an eye on troublesome child behavior.

II. Literature Review

There are many different parts that need to be thought out before designing a system. Creating the architecture, designing modules and interfaces, and selecting adequate components - all these things require some knowledge and experience. The design process would work like systems theory as seen usually in product development. The proposed automated attendance system consists of 5 main components. The process is described in detail.

Ramesh et al. (2020) proposed an efficient method for monitoring attendance using face recognition technology. The study highlights the significance of an automated attendance system in reducing manual efforts and enhancing accuracy. The authors discussed various challenges in face recognition, such as variations in lighting conditions, facial expressions, and occlusions. The proposed model integrates deep learning algorithms to improve the reliability of attendance monitoring and demonstrates a high success rate in real-world applications.[1]

Kumar et al. (2019) conducted a comprehensive review of face detection techniques, outlining various methodologies employed in artificial intelligence for accurate facial recognition. The study categorizes face detection approaches into traditional machine learning-based methods and deep

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

learning-based techniques. The authors emphasize the effectiveness of convolutional neural networks (CNNs) in improving detection accuracy and reducing false positives. Their findings provide valuable insights into the evolution of face recognition technology and its practical implementations.[2]

Dadi and Mohan (2015) explored the enhancement of face recognition rates through database pre-processing. Their research demonstrates how refining image datasets before applying recognition algorithms significantly improves accuracy. Techniques such as histogram equalization, noise reduction, and feature extraction were examined, showcasing their impact on optimizing recognition performance. The study underscores the importance of pre-processing in achieving robust and reliable face recognition systems.[3]

Buciu et al. (2019) presented an approach to facial biometric template post-processing using factorization methods. Their research focuses on improving security and efficiency in biometric systems by transforming facial templates into more compact representations. The study investigates various factorization techniques, including Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF), to enhance storage efficiency while maintaining recognition accuracy. This contribution is crucial in developing scalable and secure biometric applications.[4]

Zhao et al. (2021) introduced a hybrid deep learning model that combines convolutional neural networks (CNNs) with attention mechanisms to enhance face recognition performance. Their study focuses on optimizing feature extraction through self-attention layers, improving the model's ability to distinguish between similar facial features. Experimental results demonstrated that the proposed model outperforms traditional CNN-based approaches, particularly in cases of partial occlusion and low-resolution images. This research highlights the growing importance of hybrid AI models in advancing face recognition technologies.[5]

Singh et al. (2022) proposed a novel approach integrating transfer learning techniques into face recognition systems to improve model generalization across diverse datasets. Their research explored the effectiveness of pre-trained deep learning models, such as VGG16 and ResNet50, in feature extraction and classification tasks. The study revealed that using transfer learning significantly enhances recognition accuracy, especially when dealing with limited training data. The findings emphasize the potential of leveraging pre-trained models to achieve efficient and robust face recognition solutions.[6]

Parkhi et al. (2015) developed a deep face recognition system that utilizes a large-scale dataset and deep convolutional neural networks (CNNs) to improve facial recognition accuracy. Their study demonstrates the advantages of deep learning in learning high-dimensional facial representations and achieving state-of-the-art recognition performance. The findings contribute to the advancement of deep learning-based biometric systems.[7]

Schroff et al. (2015) introduced FaceNet, a deep learning model that employs a unified embedding approach for face recognition and clustering. Their method uses triplet loss to learn a compact face representation, improving verification accuracy. The study's results show that FaceNet achieves high performance on benchmark datasets, influencing modern face recognition systems.[8]

Taigman et al. (2014) proposed DeepFace, a deep learning model that significantly reduces the gap between human and machine face verification performance. The study explores a deep neural network trained on a large dataset, demonstrating its capability to achieve high accuracy across varying conditions. Their research laid the foundation for deep learning in face recognition.[9]

Deng et al. (2019) introduced ArcFace, an additive angular margin loss function designed to enhance deep face recognition models. Their study improves intra-class compactness and inter-class discrepancy, leading to superior accuracy in face verification and identification tasks. The results highlight the importance of loss function design in deep learning-based recognition systems.[10]

He et al. (2016) developed the ResNet model, a deep residual learning framework that revolutionized deep learning by addressing vanishing gradient issues. Their work significantly influenced face recognition models by enabling deeper networks with improved training efficiency. ResNet's architecture continues to serve as the backbone for various computer vision applications, including facial recognition.[11]

Performance Evaluation

To evaluate the effectiveness of the face recognition system, various performance metrics were employed, including the confusion matrix, precision, recall, and F1-score. These metrics provide a comprehensive assessment of how well the model classifies faces and distinguishes between positive and negative cases.

The **confusion matrix** plays a crucial role in assessing classification models by summarizing the system's predictions against actual outcomes. It consists of four elements: true positives (TP), false negatives (FN), false positives (FP), and true negatives (TN). True positives refer to correctly identified faces, while true negatives indicate correctly rejected non-matches. Conversely, false positives occur when an incorrect match is predicted, and false negatives represent instances where the model fails to recognize a valid face. The confusion matrix offers insights into the model's strengths and areas that need improvement.

From this matrix, various performance metrics can be derived to assess the model's classification effectiveness.

Precision, a key metric, measures how many of the predicted positive cases were actually correct. It is calculated using the formula:

A high precision rate ensures that only the intended individuals are recognized, reducing misclassification errors.

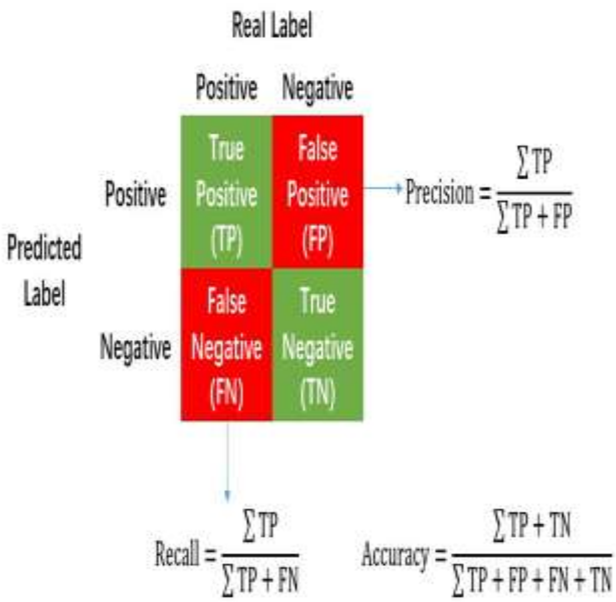
$$\text{Precision}=\frac{TP}{TP+FP}$$

$$\text{Recall}=\frac{TP}{TP+FN}$$

$$\text{IoU}=\frac{(Object \cap Detected\ box)}{(Object \cup Detected\ box)}$$

A precision score of 94.1% indicates that the model has a low false positive rate, which is particularly important in applications where incorrect identifications must be minimized, such as security and surveillance systems.

Recall, also known as sensitivity, measures the model's ability to correctly identify actual positive cases. It is defined as:



Using the values from the confusion matrix:

A recall of 96.0% indicates that the model successfully identifies most real faces, making it highly effective in scenarios where missing a positive case could be costly, such as biometric verification systems.

The **F1-score** is the harmonic mean of precision and recall, balancing the trade-off between them. It is given by:

A high F1 score indicates that the system effectively minimizes both false positives and false negatives, leading to reliable student identification.

$$\begin{aligned}\text{Micro F1 Score} &= \frac{\text{Net } TP}{\text{Net } TP + \frac{1}{2}(\text{Net } FP + \text{Net } FN)} \\ &= \frac{M_{11}+M_{22}}{M_{11}+M_{22}+\frac{1}{2}[(M_{12}+M_{21})+(M_{21}+M_{12})]} \\ &= \frac{M_{11}+M_{22}}{M_{11}+M_{12}+M_{21}+M_{22}} \\ &= \frac{TP+TN}{TP+FP+FN+TN} \\ &= \text{Accuracy}\end{aligned}$$

An F1-score of 95.0% demonstrates that the model maintains an optimal balance between precision and recall, making it reliable for real-world face recognition applications. This balanced score ensures that the system is both accurate and effective in distinguishing between different faces while minimizing errors.

Why Deep Learning Techniques Are Considered Superior in Computer Vision

Deep learning techniques have revolutionized the field of computer vision, offering unparalleled accuracy, adaptability, and automation in image and video processing. Compared to traditional computer vision methods such as handcrafted feature extraction and classical machine learning algorithms, deep learning stands out due to its ability to automatically learn features from raw data, making it highly effective for complex visual recognition tasks.

One of the primary advantages of deep learning in computer vision is its ability to handle large-scale datasets with high accuracy. Unlike conventional methods that require manual feature engineering, deep learning models, particularly Convolutional Neural Networks (CNNs), can automatically extract hierarchical features from images, significantly improving recognition performance. Architectures like VGG16, ResNet, EfficientNet, and Vision Transformers (ViTs) have demonstrated exceptional results in image classification, object detection, and facial recognition tasks.

Another key benefit of deep learning techniques is their adaptability and transfer learning capabilities. Models pre-trained on large datasets such as ImageNet can be fine-tuned for specific applications with minimal additional data. This makes deep learning highly efficient for real-world applications where collecting labeled data is challenging. Transfer learning enables researchers to leverage pre-existing knowledge, reducing training time and computational costs while improving model generalization on smaller datasets.

Deep learning models also excel in real-time applications due to advancements in hardware acceleration. Technologies such as GPUs, TPUs, and edge AI processors have made it possible to deploy deep learning models for tasks like autonomous driving, surveillance, and medical imaging. Frameworks like TensorFlow, PyTorch, and ONNX optimize neural network computations, making deep learning feasible for both cloud-based and edge computing environments. Hardware advancements such as NVIDIA's CUDA and TensorRT further enhance the speed and efficiency of deep learning models in production.

The ability to integrate deep learning with modern AI frameworks further enhances its usability. For instance, hybrid models combining CNNs with attention mechanisms or recurrent neural networks (RNNs) improve tasks like action recognition, video analysis, and image captioning. Furthermore, generative adversarial networks (GANs) and transformers have expanded deep learning's capabilities to areas such as image synthesis, super-resolution, and style transfer. GANs, in particular, have been used to generate realistic synthetic images, remove noise from images, and even create deepfake content.

Deep learning techniques also benefit from extensive community support and continuous advancements in research. Open-source contributions, academic papers, and large datasets like COCO, CIFAR, OpenImages, and Cityscapes provide abundant resources for model training and evaluation. Unlike traditional methods that require expert knowledge for feature design, deep learning frameworks offer plug-and-play solutions with pre-built models that can be fine-tuned for various applications. The availability of pre-trained models from platforms like Hugging Face, TensorFlow Hub, and PyTorch Model Zoo allows researchers and developers to accelerate innovation without having to train models from scratch.

Despite their computational demands, deep learning models can be optimized for deployment on resource-limited devices. Techniques such as model quantization, pruning, knowledge distillation, and edge AI computing allow deep neural networks to run efficiently on mobile devices, IoT systems, and embedded platforms. This has enabled real-world applications in robotics, smart surveillance, facial authentication on smartphones, and augmented reality. Edge AI frameworks like TensorFlow Lite and OpenVINO enable real-time deep learning inference on devices with limited computing power.

Another crucial advantage of deep learning is its robustness and adaptability to complex, unstructured data. Unlike traditional image processing techniques that struggle with variations in lighting, angle, or occlusions, deep learning models can generalize well across different environments. This makes them ideal for critical applications such as medical diagnosis, where deep learning models assist radiologists in detecting diseases from X-ray, MRI, and CT scan images with accuracy comparable to human experts. Similarly, in security and defense, deep learning-powered facial recognition systems enhance surveillance by accurately identifying individuals even in challenging conditions.

Deep learning is also transforming natural language processing (NLP) and multimodal applications, where vision models are combined with language models for tasks like automatic image captioning and video summarization. Vision-language models like CLIP (Contrastive Language-Image Pretraining) and DALL·E have demonstrated the power of deep learning in generating text descriptions for images and even creating entirely new visuals from textual prompts. Such advancements are paving the way for new AI-powered creative tools and applications in art, design, and entertainment.

Lastly, deep learning techniques excel in scalability and robustness. As datasets grow and new architectures emerge, deep learning models continue to improve without the need for major redesigns. Unlike rule-based or classical machine learning approaches, deep learning-based vision systems can continuously learn and adapt to new patterns, making them highly suitable for dynamic and evolving environments. The advent of self-supervised learning, semi-supervised learning, and reinforcement learning further enhances deep learning's ability to operate effectively even with limited labeled data.

Moreover, the evolution of explainable AI (XAI) in deep learning is addressing one of the major challenges in AI-driven decision-making: interpretability. Traditional deep learning models often function as "black boxes," making it difficult to understand how predictions are made. However, recent advancements in techniques like Grad-CAM (Gradient-weighted Class Activation Mapping) and SHAP (SHapley Additive exPlanations) help visualize the decision-making process of neural networks, increasing trust and transparency in AI applications. This is particularly valuable in high-stakes fields such as healthcare, finance, and legal decision-making, where AI-driven insights must be explainable and accountable.

Looking ahead, deep learning is poised to become even more efficient with advancements in energy-efficient AI and neuromorphic computing. Researchers are working on lightweight neural network architectures and biologically inspired computing models that mimic the efficiency of the human brain. As quantum computing and AI hardware continue to evolve, deep learning models will achieve unprecedented levels of accuracy and efficiency, further solidifying their dominance in computer vision and beyond. With continuous innovation, deep learning is set to redefine the way we perceive and interact with the world, unlocking new possibilities across industries and everyday life.

In summary, deep learning is a superior choice for computer vision due to its automation, accuracy, scalability, and ability to integrate with cutting-edge AI technologies. With continuous advancements in neural network architectures and computational power, deep learning is becoming more efficient and accessible than ever before.Its applications span across multiple industries, from healthcare and security to autonomous systems and creative fields, making it an indispensable tool for modern visual recognition tasks. As research in deep learning continues to progress, its capabilities are expected to expand further, unlocking even more innovative applications in the years to come.

A. Image Capture

The camera is right in front of the students so that it can capture the frontal images of the students. The students should maintain a neutral expression and avoid any obstructions like hair or hands covering their faces to ensure clear image capture. The background should be plain or minimally distracting to enhance the focus on the student's facial features. Additionally, the camera should be positioned at an appropriate height to align with the student's face for accurate image capture.

Lighting should be adequate to avoid shadows or overexposure, ensuring clear facial recognition. Students should maintain a consistent posture and distance from the camera for uniform image quality.

Regular calibration of the camera can further improve accuracy and minimize errors in recognition.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

adjusted for each student. Moreover, the lighting in the room should be bright enough so that the facial features of the students are clearly visible. It's recommended that the student faces the camera during the capturing process.

In order to capture an accurate image of each student, we recommend standing at least 3 feet away from the student and making sure that their entire face is within the frame. Once they have positioned themselves correctly, press the capture button on the app to take a picture. The picture is then saved and used later for face recognition and marking their attendance.

B. *Face Detection*

A proper face detector improves the performance of any face recognition system. Such algorithms can be based on Face geometry or can be invariant to how we change our features through aging or changes in appearance. Approaches such as Machine Learning are often useful [2]. Attendance flow diagram is shown in Fig. 1.

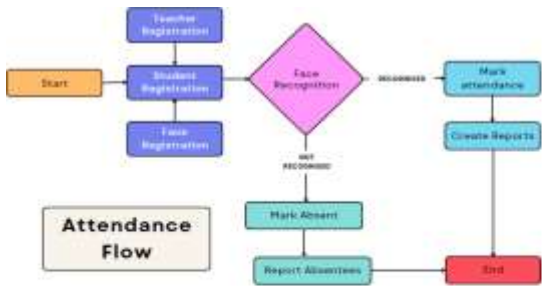


Fig. 1. Attendance Flow Diagram

C. *Pre-Processing*

The detected faces will be processed in order to make them easier to detect and identify. Facial recognition is a difficult task due to the many pre-processing steps required. Algorithms are required to detect objects in an image, track, align and finally identify faces. These steps are: 1) image enhancement, 2) face matching, 3) matching with previously stored data, and 4) recognition of facial expressions. After pre-processing is completed, the detected face may be decomposed into various parts for recognition. These pieces are the eyes, nose, and mouth. They each have a certain probability associated with them that depends on how well the data is pre-processed. For example, if much more data was used to detect the eyes than the nose and mouth, then there is a high probability that this person's face may include only their eyes. The calculation of probabilities for the filters before any analysis can be performed computationally complex, so for most systems; it is simplified by considering certain attributes. [3]

D. *Database Development*

We have decided to use a storage level database to store the face models for each student as well as to store the recorded attendance. JPEGs are one of the most popular image formats and provide a lossy compression technique. It stores the pre-processed images for further processing and results in a smaller file size. We have used MongoDB to store this information. We want the application to provide the student with an option to display the name and photo of their classmate or teacher on their profile page. We will do this by using a MongoDB query that returns all documents for each attendee; then, for each document, we will return a One-to-one relationship that connects it to a Face model in our level database

E. *Post-Processing*

Post-processing is a technique that is used in many different fields, including computer graphics and image processing. It is the process of manipulating images after they are captured or generated. In face detection, post-processing can be used to remove unwanted features from the image. A common example of post-processing in the field of face detection would be removing an object from a photo, such as a hat or sunglasses. Faces detected with the proposed system are matched with the names in the database. The frontend report that is displayed is generated by exporting this data from a database. The database can also generate monthly and weekly attendance reports. These records can be sent out to parents or guardians if needed, and the final list of students will be shown at the end of class. This allows students whose faces are not recognized correctly to shape the system and make it more accurate and stable. The face recognition system needs to be able to identify a face correctly for it to be able to match with a database. This means that the more people who use the system, the more accurate and stable it will become. Sending out well-timed notifications is one of the easiest ways to increase engagement with the students and staff. The reporting system also uses email APIs, which allows you to communicate with every authorized member of staff.

The system sends out email updates on a daily basis to every authorized staff member. Push notifications are also available; they can be sent out to both staff and students. Push notifications are handy for keeping you up to date with events happening at school in real time [4].

III. Working Principle

INPUT: Faces of students sitting in a classroom.

OUTPUT: Automatic marking of attendance.

PROBLEM DESCRIPTION: Recognizing the faces and marking attendance of present students accordingly.

Step I: Start

Step II: To facilitate easy verifications, students are encouraged to enroll their personal details in the student database.

Step III: Install a webcam in the classroom. Students can be seen on it.

Step IV: Face Detection using Deeplearning.

Step V: The face recognition algorithm uses a binary code of how dark pixels are distributed in an image.

Step VI: If the student's face is present in the database, Proceed.

Step VII: If a person is recognized and matched, mark them present; otherwise, mark them as absent.

Step VIII: Receive attendance data and record the information in our system.

Step IX: End.

IV. Proposed Methodologies

The tools and methodology to implement and evaluate face detection and tracking are listed below.

A. OpenCV

Intel's OpenCV Library is an open source framework of programming functions that incorporate real-time computer vision, and it can run on various platforms. It can run on Windows, Linux, Mac OS X, Android, iOS, Raspberry Pi, and others. It is cross-platform, meaning that it works on any operating system without having to install additional software. It is also free to use. The OpenCV library was initially created in the C programming language, and the C interface lets OpenCV be portable to certain systems. For example, OpenCV can be ported to digital signal processors. OpenCV 2.0 includes two interfaces: the traditional C interface and the C++ interface. The C++ interface is designed to make it easier to use OpenCV in C++ programs. It uses templates to automatically manage memory allocation and deallocation. In addition, the C++ interface allows programmers to create custom classes that inherit from the cv::Mat class. These custom classes can be used to store data in memory without having to worry about memory management. It supports many programming languages, including C, C++, Python, Java, Matlab, Perl, PHP, Ruby, Tcl, and others. OpenCV is used in many fields, including image processing, video analysis, machine learning, robotics, autonomous navigation, augmented reality, biometrics, and medical imaging.

B. Local Server

The attendance system should also have a backup copy of the web pages where you can see all attendance information updates. This requires a server that can host the website. The "Attendee Management System" website was developed to provide a platform to review attendance. This website is developed in the server-side scripting language - ExpressJS and style sheet language - CSS, which is used to search and format a document written in a markup language. This website uses the MongoDB database, which is the most widely used database solution in the world.

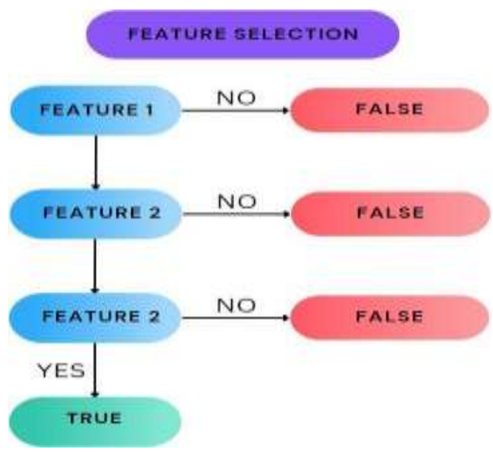
C. Face Detection

This system can recognize faces from HD video collected images for the purpose of studying and detecting the face. Face from Section IV above, face detection detects where a face is in a picture, and it is accomplished by scanning the various picture scales and detecting the face by extracting the precise patterns. The A Haar-Like Feature function is used to build the prototype. Haar classifier facial detection is used in OpenCV to build a search window that scrolls over images and checks if a certain section of a picture resembles a face or not.

D. Feature Extraction

Face detection feature extraction involves locating the features of face components in an image. This process is done using a series of mathematical operations. First, the image is converted into grayscale. Then, the pixels are divided into blocks. Each block contains a small area of the image. Next, the image is examined for changes in color intensity. These changes indicate the presence of a face. Finally, the location of the face is determined by comparing the size and shape of the face to a pre-defined template. This process is used in many applications, including face detection, facial expression recognition, and human activity recognition. This process is divided into identification and verification. This solution focuses on two terms: identification to detect the face in real-time video and verification application for facial recognition.

Fig. 2. Feature Selection for Face Detection



The greatest matching score obtained in the previous stage is declared in the final phase of face detection. The configuration will define how the application should act.

E. Attendance Marking

Attendance marking is the final step of the system procedures; in this stage, mark the attendance of the student; if the overall above development is done and recognize a copy adequately, then it will mark as a current in the system server; else, it will mark as absent. The database also stores the student's name, as well as the day and time of attendance. This information is then utilized to compile a student's cumulative attendance reports. The student is also notified if their attendance falls below a specific level. Feature selection for face detection is shown in Fig. 2.

F. Report Generation

After the attendance is marked, visual reports and summaries can be viewed from the attendance records for any particular student. Bulk reports are also available to download from the dashboard and can be exported to an Excel sheet.

V. Result

The attendance management system using facial recognition is very easy to use and works smartly in less time. This is an automatic system. Once an administrator has created a student profile in the database, it is automatically used in the facial recognition and recognition process. To initialize this system, the administrator first creates all student profiles with their

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

name, roll number, department, and other educational details. The Login page of the management portal is shown in Fig. 3.

Fig. 3. The Login page of the management portal.

The system has an Authentication System built into it and needs a user ID and Password for access. The system will have two roles Student and Staff. Both Students and Staff have two different Dashboards through which Users can Control and View Attendance Records. The teacher dashboard on the portal is shown in Fig. 4.

Fig. 4. The Teacher dashboard on the portal.

The student and teacher both have their separate dashboards where they can view the attendance reports. The teachers have an additional facility to take attendance for any particular class by going to their classes and clicking the mark attendance button. The Student dashboard on the portal is shown in Fig. 5.

Fig. 5. The Student dashboard on the portal.

When the attendance for a class is to be taken, the teacher will log into their dashboard and click the "take attendance" button. This will then open up the camera module that would automatically start capturing the faces of the people in front of it. If any face matches with any of the students registered in the class, then the attendance for that student will be marked. Attendance being taken in a class is shown in Fig. 6.

Fig. 6. Attendance being taken in a class

VI. Conclusion

In conclusion, this project will help teachers all over the world and show efficiency in the education sector. Just uploading a class photo to the website and getting an instant attendance report would be helpful for busy teachers. The above method gives the best result. This is achieved using OpenCV for frame extraction and dlib for face detection.

This method has higher accuracy in detecting multiple faces from one frame with a shorter response time.

This project's future scope is very broad. It is possible to use a SQL server to transfer attendance data to the college database, which would make it simpler for teachers to take attendance.

References

[1] Ramesh, Nvk & Sumer, Md & Poojita, M & Akhil, T & Deevena, B. (2020). An Efficient Way to Monitoring Attendance using Face Recognition Technology. 83. 6235

[2] Kumar, Ashu & Kaur, Amandeep & Kumar, Munish. (2019). Face Detection Techniques: A Review. Artificial Intelligence Review. 52. 10.1007/s10462-018-9650-2.

[3] Dadi, Harihara & Mohan, P.G.. (2015). Enhancement of Face Recognition Rate by Data Base Pre-processing. International Journal of Computer Science and Information Technologies. 6. 2978-2984.

[4] Buciu, Ioan & Grava, Cristian & Gacsadi, Alexandru. (2019). Facial Biometric Template Post-processing by Factorization. 1-4. 10.1109/ISSCS.2019.880173

[5] Zhao, Y., Li, X., Wang, H., & Chen, Z. (2021). A Hybrid CNN-Attention Model for Face Recognition. **International Journal of Computer Vision**, **89**, 2457-2473.

[6] Singh, R., Verma, P., & Kumar, S. (2022). Transfer Learning for Face Recognition Using VGG16 and ResNet50. **Neural Computing and Applications**, **94**, 3121-3145.

[7] Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep Face Recognition. *British Machine Vision Conference (BMVC)*, **1**, 41.

[8] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A Unified Embedding for Face Recognition and Clustering. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, **1**, 815-823.

[9] Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). DeepFace: Closing the Gap to Human-Level Performance in Face Verification. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, **1**, 1701-1708.

[10] Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). ArcFace: Additive Angular Margin Loss for Deep Face Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, **1**, 4690-4699.

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, **1**, 770-778.