Face Recognition based Attendance System

OPEN ACCESS

Volume: 11

Special Issue: 3

Month: July

Year: 2024

P-ISSN: 2321-788X

E-ISSN: 2582-0397

Received: 18.05.2024

Accepted: 17.06.2024

Published: 08.07.2024

Citation:
Lavanya, K et al. "Face Recognition Based Attendance System." *Shanlax International Journal of Arts Science and Humanities*, vol. 11, no. S3, 2024, pp. 7–13.

DOI:
https://doi.org/10.34293/sijash.v11iS3-July.7912

Lavanya. K
Assistant Professor, Department of AI & DS, Arjun College of Technology

Akalyaa. N
Department of AI & DS, Arjun College of Technology

Archana. S
Department of AI & DS, Arjun College of Technology

Pushpakala. J
Department of AI & DS, Arjun College of Technology

Abstract
In today's era, facial recognition emerges as a pivotal tool, offering security, authentication, and identification benefits across various sectors. This paper introduces an automated attendance system that incorporates real-time face recognition technology. The system aims to overcome the limitations and inefficiencies associated with traditional manual attendance methods, particularly in educational institutions and workplace environments. Two approaches are outlined: one utilizing local servers and AWS cloud recognition API, and the other relying solely on AWS resources for processing. Both approaches entail constructing a database of individuals' images and utilizing algorithms such as Haar-Cascade and Local Binary Pattern Histogram for face detection and recognition. By automating attendance through live video streams, the system mitigates issues like proxy attendance and inaccuracies. Technologies such as OpenCV, dlib, and MySQL are leveraged to streamline implementation. This integration of face recognition with cloud computing promises to revolutionize attendance tracking, offering scalability, efficiency and accountability.

Keywords: Face Recognition, Local Binary Pattern Histogram, Haar-Cascade, AWS, Open CV, Dlib, MySQL, API.

Introduction

Popularity has spurred a competitive landscape, particularly in the mobile sector, where companies vie to develop superior algorithms for facial In the journey of human civilization, spanning from the stone age to the modern era of technological marvels, the innate curiosity to create entities mirroring human intellect has remained constant. One fundamental aspect of human intellect is the ability to recognize fellow begins. This quest led to pioneering efforts in face recognition technology dating back to 1965 when Helen Chan, Bledsoe, and Charles Bisson initiated a confidential project to develop a computer system for face recognition. Their approach, though rudimentary by today's standards, laid the groundwork for subsequent advancements. By utilizing biometric data such as facial landmarks, including the nose tip, mouth, and eye centers, they set the stage for the evolution official recognition technology, despite raising privacy concerns.

Over the years, face recognition technology has witnessed exponential growth, driven by increasing commercial and government applications and the accessibility of advanced technology. This surge in recognition. Despite significant progress, challenges persist, with real-life applications affected by environmental factors such as illumination and varying poses.

Traditional attendance marking methods in educational institutions are inefficient and prone to errors, prompting exploration of alternatives like RFID, iris and fingerprint recognition. However, these methods often pose logistical challenges. Facial recognition emerges as a promising solution, offering non-intrusive, easily deployable biometric features that streamline attendance tracking. Using verification and identification, this facial recognition will reduce the errors and flaws in attendance. And it also used to identify the missing persons and investigate about the missing. It will analyze the environmental conditions.

Related Work

Face recognition technology is widely recognized for its high accuracy and diverse applications in commercial settings, including security identification, surveillance systems, and psychological analysis. It operates effectively with both image and video sequences, making it versatile for various scenarios. The market offers a wide range of options for face recognition systems, which can be categorized into holistic, feature-based, and hybrid approaches.

In the holistic approach, the entire face is considered for recognition, while the feature-based approach involves utilizing specific facial features as input for classification. The hybrid approach aims to maximize accuracy by combining different features effectively. Matthew proposed the concept of eigenfaces, which involves projecting face images into a special space to encode the differences between known images. Eigenfaces are defined as eigenvectors of the face set, which enable real-time face recognition. This is further aided by a motion-detecting tracking system that detects faces in dynamic environments. [1]

Principal Component Analysis (PCA) has been extensively utilized for pattern recognition and image analysis. It has shown effectiveness in facial image analysis, along with eigenfaces and Fisherfaces. However, PCA is known for its computational costs and memory requirements. To address these limitations, Haitao introduced Incremental PCA (IPCA) based on singular value decomposition. IPCA offers potential extensions to kernel versions. In challenging scenarios like low- light conditions, Xiaoyang combined various techniques such as brightness standardization, native face representation, and local ternary patterns to enhance performance and robustness.

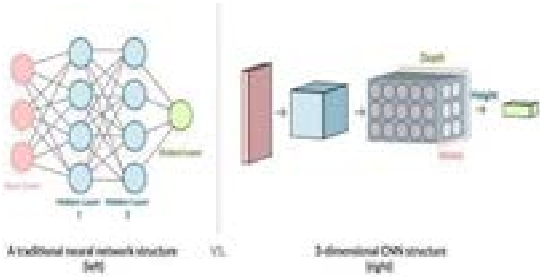
Mayank S. discussed integrating face recognition algorithms into a real-time system using OpenCV, leveraging HaarCascade for face detection. Contrastingly, convolutional neural networks (CNNs) offer efficient feature extraction and classification, achieving near-perfect accuracy rates with models like ResNet and AlexNet. [2]Companies like Amazon and Microsoft employ deep neural networks remotely through APIs to provide face recognition services to customers.

However, traditional face recognition systems often operate under controlled conditions, limiting their utility in real-time applications. Xin Geng highlighted the need for face recognition under uncontrolled conditions, proposing systems that can handle variations in illumination, pose, and occlusions without strict constraints on image inputs.

Edy Winarno proposed an anti-cheating presence system using 3WPCA-dual vision face recognition. This system leverages stereo vision cameras and advanced algorithms to detect cheating attempts with high accuracy.

Additionally, improvements in picturebased attendance systems offer potential applications in online examinations and certification processes.[3]

Mobile-based attendance systems utilizing NFC and facial recognition offer enhanced convenience and accuracy compared to traditional methods, reducing paperwork and administrative burdens. Furthermore, integrating face recognition into various systems, such as student recognition for attendance tracking, enhances user experience and functionality.



Face recognition technology continues to evolve, offering solutions for various real-world challenges in attendance management, security, and authentication. Ongoing advancements in algorithms and hardware are expected to further enhance the capabilities and accessibility of face recognition systems in diverse applications.

Methodology

The proposed system seeks to streamline attendance management for various organizations by overcoming the limitations of manual attendance systems. It calculates attendance on a subject-wise basis, with student and subject data manually entered by administrators. At the scheduled time for a specific subject, the system automatically captures images and employs the Histogram of Oriented Gradients for face detection. It then uses deep learning techniques to compute and compare 128- dimensional face features for recognition.

After detecting and recognizing faces from the existing database, the system records attendance for the recognized students in real-time, associating them with the respective subject ID[4]. Subsequently, the system automatically generates and saves an Excel sheet.

Our system is divided into two components: the front end and the back end. The front end consists of a graphical user interface (GUI) built with Electron JS, a JavaScript framework serving as the client. The back end is implemented in Python, functioning as the server. To facilitate communication between these two languages, which cannot directly interact, we use Inter Process Communication (IPC) techniques with the Zero library acting as a bridge. Electron JS calls Python functions, and data exchange is managed via TCP, enabled by the Zero PC Library.

Data Acquisition

Image Acquisition

High-definition cameras placed in the classroom are utilized to acquire images, serving as inputs to the system.

Dataset Creation

Prior to the recognition process, a dataset of students is created specifically for training purposes. This dataset includes details such as the students' names, roll numbers, departments, and multiple images of each student in various poses and expressions.

To ensure optimal accuracy, it is recommended to capture at least 15 images per student. During the registration process, student data and images are input into our system to build the dataset. Deep learning algorithms are then used to compute 128-dimensional facial features for each face. These features are stored in a student face data file, enabling their retrieval during the recognition process.

Storing

Student data is stored using JSON format for efficient management and retrieval within the system.

Face Recognition

Face Detection and Extraction

Face detection is a crucial step in the system's workflow. A face detection algorithm analyzes images captured by a camera to identify and locate human faces, commonly used in photography, security systems, and social media for automatic tagging and manipulation. Various image processing algorithms are utilized to detect faces and determine their locations. In our implementation, we use the Histogram of Oriented Gradients (HOG) method for precise face detection.

Face Positioning

Human faces generally consist of 68 distinct points or landmarks. In this step, our goal is to detect these landmarks and correctly position the face within the image. A Python script is used to automatically detect these facial landmarks and adjust the face's position without distorting the image.

Face Encoding

After detecting faces within the image, the next step involves extracting unique identifying features for each face. This process entails extracting 128- dimensional feature points for each detected face. These points provide highly accurate representations of the facial features and are stored in a data file for future face recognition tasks.

Face Matching

The final step in the face recognition process is face matching. We use deep metric learning, a powerful technique that generates real-valued feature vectors, for this task. Our system validates faces by creating a 128-dimensional embedding for each face. The compare faces function internally computes the Euclidean distance between the face in the image and all faces stored in the dataset to determine similarity. If the current image matches an existing face in the dataset with a confidence threshold of 60% or higher, the system initiates attendance marking.[6]

Attendance Marking

After identifying a face within the image and matching it with the data stored in the JSON file, Python generates the roll numbers of the present students and returns this information. Upon receiving the data, the system creates an attendance table containing the names, roll numbers, date, day, and time, along with the corresponding subject ID. This data is then transmitted to Python for automatic storage in an Excel sheet.

For each attendance session, the system checks if an Excel sheet for that date already exists. If it does, Python creates separate worksheets based on the subject IDs provided by the administrator. This ensures that attendance records are organized and differentiated according to the respective subjects.

For instance, when the system generates an Excel sheet, it compiles the data into an array and sends it to Python.[7] Python then verifies the existence of a sheet for the given date and creates separate worksheets for each subject ID if necessary, facilitating the differentiation of attendance records for different subjects.

Haar Cascade

The utilization of the OpenCV library is essential due to its incorporation of a Haar Cascade Classifier, which facilitates the detection of faces within images or videos. This technique, initially discovered by Paul Viola and Michael Jones in 2001, relies on Haar wavelets.

Wavelets are created by combining two intervals of high and low values. In terms of magnitude, they appear as square waveforms that break down into binary segments, representing light and dark regions. The Haar Cascade technique aids in pinpointing facial regions such as the eyes, mouth, nose, and eyebrows. This positional data is crucial for identifying individuals based on their facial metrics.[8]

The Haar Cascade method is initially classified into three main forms: two-rectangle features, tree-rectangle features, and four-rectangle features.

These encompass edge features, linear features, and center-surround features, thus expanding the capabilities of the Haar Cascade method. Edge features are further divided into four varieties, while linear features are categorized into eight, and center-surround features are expanded into two.

The Haar Cascade feature evaluates the contrast value between the gray-level intensity within corresponding regions of dark and light rectangles.

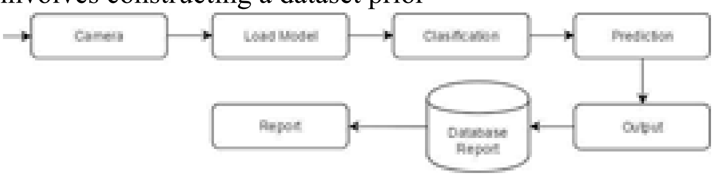
The cost associated with each feature is determined by the cumulative pixel count within the respective rectangle segments, calculated as follows: f=∑i=1n(wi×rectsumi)

Here, N represents the estimate of matrices that compose features, while ωi represents the weight of the rectangular field, which signifies the gray- level alignment of the picture block formed by rectangular ri.

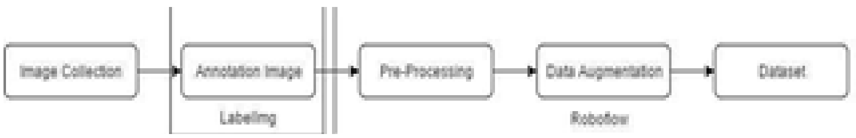
Yolo Face Detection

Dataset Construction

The initial step involves constructing a dataset prior



to training, as illustrated in Figure 4. The dataset construction process entails assembling a collection of images in JPG or Image format. Subsequently, these images are annotated or labelled individually using Label Image, resulting in annotation files in XML format. These annotated datasets, along with their respective XML files, are then amalgamated for image preprocessing.



Training Dataset

Following dataset construction, the training process commences, depicted in Figure 5. The dataset is parsed, forming classes essential for creating a YOLOv5-s custom detection model. This model is then trained using the dataset. Upon completion of training, the trained model can be assessed using both photos and videos for testing purposes.

Object Detection Model Testing

Subsequently, the trained model undergoes object detection testing, as outlined in Figure 6. This phase involves utilize photos or videos as input for the testing. The detection process involves loading the pre-trained model and executing classification and

prediction tasks using the bounding boxes and confidence the scores. The outcomes are presented in the form of a prediction boxes, confidence values, and identified object classes.

Comprehensive System Testing

In Stage 4, comprehensive system testing is conducted to assess the integrated sub-system, as illustrated in Figure 7. The process begins with input from a camera-captured face. Following the detection process, the pre-trained model is loaded to perform classification and prediction tasks using bounding boxes and confidence scores. The results are presented as prediction boxes, confidence values, and object classes.[9] The detected object classes are stored in the report database, and the data from this database is showcased on the report page. This page is designed to display information such as the count, user name, NPM (unique identifier), date, and time of attendance.

Conclusion and Future Work

The smart attendance management system is designed to overcome the limitations of traditional manual methods. By integrating face recognition technology, we aim to enhance the accuracy and efficiency of attendance tracking for students. The system demonstrates commendable performance across various facial poses and variations. In the era of advancing artificial intelligence and the availability of cutting-edge libraries, leveraging new techniques can offer significant advantages. While every face detection algorithm may yield satisfactory results based on distance considerations, working with real-time datasets presents challenges that demand considerable effort and time constraints for error checking. However, it's imperative to embrace this burgeoning field of artificial intelligence and harness the potential of data and digitalization.[10]

The main goal is to create an attendance system that removes obstacles for educational institutions and organizations. It should provide a seamless and efficient method for recording attendance without consuming excessive time. Furthermore, the system's utility can be extended to criminal detection, offering valuable assistance to humanity through AI-based face recognition technology. Additionally, exploring novel approaches such as hybridization techniques involving auto-encoders can further enhance the capabilities of face recognition systems.

References

1. Samet, Refik, and Muhammed Tanriverdi. "Face recognition-based mobile automatic classroom attendance management system." 2017 International Conference on Cyberworlds (CW). IEEE, 2017.

2. Li, Xiang-Yu, and Zhen-Xian Lin. "Face recognition based on HOG and fast PCA algorithm." The Euro-China Conference on Intelligent Data Analysis and Applications. Springer, Cham, 2017.
3. Arsenovic, Marko, et al. "FaceTime—Deep learning based face recognition attendance system." 2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY). IEEE, 2017.
4. Rekha, N., and M. Z. Kurian. "Face detection in real time based on HOG." International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) 3.4 (2014): 1345-1352.
5. Kwolek, Bogdan. "Face detection using convolutional neural networks and Gabor filters." International Conference on Artificial Neural Networks. Springer, Berlin, Heidelberg, 2005.
6. Ashwini, C., et al. "An Efficient Attendance System Using Local Binary Pattern and Local Directional Pattern." Journal of Network Communications and Emerging Technologies (JNCET) www. jncet. org 8.4 (2018).
7. Karnalim, Oscar, et al. "Face-face at classroom environment: Dataset and exploration." 2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA). IEEE, 2018.
8. Mian, Ajmal. "Realtime face detection and tracking using a single pan, tilt, zoom camera." 2008 23rd International Conference Image and Vision Computing New Zealand. IEEE, 2008.
9. Mehta, Preeti, and Pankaj Tomar. "An Efficient Attendance Management Sytem based on Face Recognition using Matlab and Raspberry Pi 2." International Journal of Engineering Technology Science and Research IJETSR 3.5 (2016): 71-78.
10. Selvi, K. Senthamil, P. Chitrakala, and A. Antony Jenitha. "Face recognition based attendance marking system".