Face Recognition Attendance System

Ch Geethika Lakshmi Ramya, M Abhiram Nadiu
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Abstract—**The face recognition attendance system is an automated attendance management system that employs computer vision technology to recognize and register individuals based on their facial features. This system utilizes a custom dataset of face images captured using the OpenCV module and a convolutional neural network (CNN) model trained on this dataset to recognize faces accurately. This system exemplifies how advancements in deep learning and computer vision technology can be leveraged to automate attendance management systems, ultimately resulting in more efficient and accurate attendance recording. The project also highlights the potential of open-source tools and frameworks like TensorFlow and Keras in building sophisticated computer vision applications. The face recognition attendance system developed in this project is a prime example of how machine learning applications like facial recognition can be applied to real-world problems, making them an essential tool in today's world.**

Keywords- face recognition, attendance system, computer vision technology, OpenCV, convolutional neural network, deep learning, automation, efficiency, accuracy, TensorFlow, Keras, machine learning.

I. Introduction

Attendance management systems have been an essential aspect of various organizations for many years. However, traditional attendance systems that rely on manual attendance recording methods are becoming increasingly inadequate in today's fast-paced world. To address this issue, modern attendance management systems have been developed that leverage computer vision technology and deep learning methods to automate attendance recording accurately and efficiently. The face recognition attendance system is a prime example of such an automated attendance system that employs computer vision technology to recognize and register individuals based on their facial features.

The main objective of this project is to develop an automated attendance system that can recognize and register individuals based on their facial features. This system utilizes a custom dataset of face images captured using the OpenCV module and utilizes deep learning methods to train a convolutional neural network (CNN) model for facial recognition.

The methodology employed in this project involves several steps, which are as follows:

1) Dataset Creation
2) Developing CNN Model
3) Architecture of Proposed CNN
4) Training the CNN
5) Evaluating the CNN model
6) Interpretation of results

II. Background and Previous Work

To understand the face recognition attendance system and implement a new project, one must have a background in computer vision, deep learning, and image processing. Familiarity with programming languages like Python, as well as open-source tools and frameworks like TensorFlow and Keras, is also essential.

Previous work in the field of face recognition systems can provide valuable insights into developing an effective attendance management system. Some previous work in this area includes research on deep learning-based facial recognition algorithms and their applications in various fields like security systems, mobile devices, and social media platforms.

Other related work includes studies on feature extraction and representation techniques, such as Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG), which can be used to extract relevant features from facial images.

Additionally, previous work in attendance management systems can provide insights into the development of a face recognition-based attendance system. This includes research on traditional methods of attendance recording, such as manual recording and barcode scanning, and their limitations, as well as previous work on automated attendance systems using technologies like Radio-Frequency Identification (RFID) and biometrics.

By building on the insights and advancements from previous work, one can develop a robust and effective face recognition attendance system that can streamline attendance recording and enhance accuracy and efficiency.

III. Dataset

The CV2 module is a Python library used for computer vision tasks, and it can be used to capture live images of a person's face. In this scenario, the CV2 module is being used to capture a dataset of images that will be used for training a machine learning model to recognize different individuals.

When a new person is added to the dataset, the CV2 module captures 50 images of their face, which are then stored in a folder named "image data". These images have a fixed size of 100x100 pixels and only contain the face of the person being captured.

Once all the images are captured and stored in the "image data" folder, the data is further processed and mapped to respective classes or labels (in this case, the student IDs). This mapping is done by creating a CSV file where each row corresponds to an image, and the columns represent the image data and its corresponding class label.

By doing this, a machine learning model can be trained to recognize different individuals based on their faces, by using the dataset generated from the cv2 module.

IV. Developing CNN Model

In the face recognition attendance system project, a convolutional neural network (CNN) model was developed to recognize faces accurately. The CNN contains multiple kernels, each with trainable parameters that can convolve on an image to detect features such as edges and shapes. The large number of filters enables the model to capture spatial features from the image based on learned weights through backpropagation. By stacking layers of filters, the model can detect increasingly complex spatial shapes from the features at each subsequent level. This allows the model to transform the input image into a highly abstracted representation, making it easier to predict and identify facial features.

Different layers that are used to build an efficient CNN:

1) Convolution: Convolution is a fundamental technique that enables us to convert unstructured image data into structured image data. In this technique, we represent the image as a digital matrix of numbers and then apply filters to sweep through the matrix one block at a time. We move the pixel based on the specified stride, and as the filters move through the image, a summarized digital image is formed, known as a feature map or activation map. Through this process, we can extract or map the most significant features from the original image using the filters.
2) Max pooling: Max pooling is a technique used after convolution to extract the most important features from the image data. This process reduces the dimensionality of the data by passing a max-pooling filter through it, which selects the maximum value in the filter as the final feature. The larger the filter used, the more compressed the information in the image becomes. Essentially, max pooling helps to simplify and compress the image data while retaining its important features.
3) Flattening: In CNN, flattening is a process of transforming the output of the convolutional layer into a 1-dimensional array by arranging the values in a sequence. This process is crucial to extract the significant features from the image for further analysis and prediction. Flattening helps to reduce the dimensionality of the data and prepare it for the next step in the CNN model. Essentially, flattening is a way to represent the processed image data in a more manageable format for further processing.

V. Architecture of proposed CNN

The CNN model proposed for this project consists of 12 layers, including 2D Convolutional Layers(Conv2D), Max Pooling Layers, and Dense Layers. The input to this architecture is an image with a shape of (100,100,1), and it predicts the class label of the image as output.

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)           (None, 94, 94, 36)        1800

 max_pooling2d_4 (MaxPooling  (None, 47, 47, 36)       0
 2D)

 conv2d_5 (Conv2D)           (None, 43, 43, 54)        48654

 max_pooling2d_5 (MaxPooling  (None, 21, 21, 54)       0
 2D)

 flatten_2 (Flatten)         (None, 23814)             0

 dense_8 (Dense)             (None, 2024)              48201560

 dropout_6 (Dropout)         (None, 2024)              0

 dense_9 (Dense)             (None, 1024)              2073600

 dropout_7 (Dropout)         (None, 1024)              0

 dense_10 (Dense)            (None, 512)               524800

 dropout_8 (Dropout)         (None, 512)               0

 dense_11 (Dense)            (None, 5)                 2565

=================================================================
Total params: 50,852,979
Trainable params: 50,852,979
Non-trainable params: 0
```

Fig. 1. Model.summary()

VI. Training the CNN

The proposed Convolutional Neural Network model was trained on a dataset consisting of a number of students (i.e., number of classes) and was divided into 80:20 train and test samples.

The paramneters used in the training process:

- Optimizer - Adam
- Learning Rate -0.0001
- Loss function -Cateogeorical cross entropy
- Batch Size -5
- Epochs -15

The accuracy and loss of the CNN model were recorded during the training process with respect to the number of epochs on both the train and test data. The results showed that as the number of epochs increased, the accuracy improved and the loss decreased. Eventually, both the accuracy and loss reached a point where they no longer changed significantly. These findings are represented in a figure.

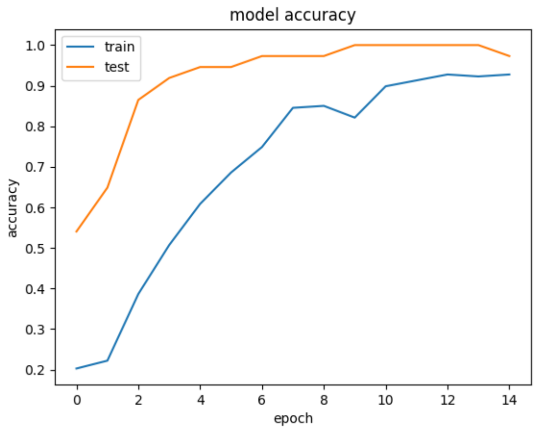


Fig. 2. Model Accuracy

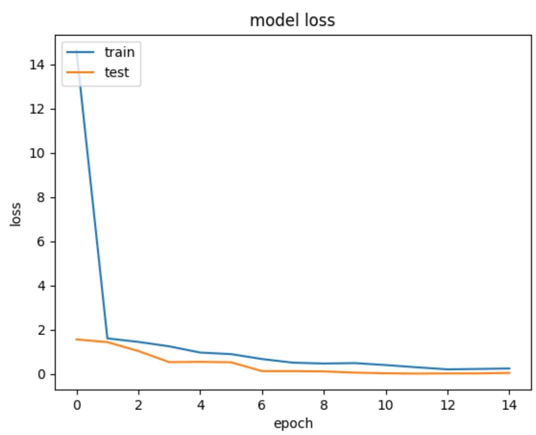


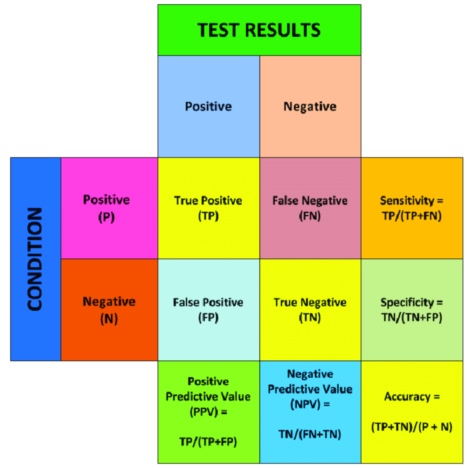
Fig. 3. Model Loss

VII. Evaluation of CNN model

Model evaluation metrics are essential to assess the performance of a face recognition system developed using a CNN model. Metrics such as precision-recall, F1 score, and accuracy can be used to evaluate the system's performance.

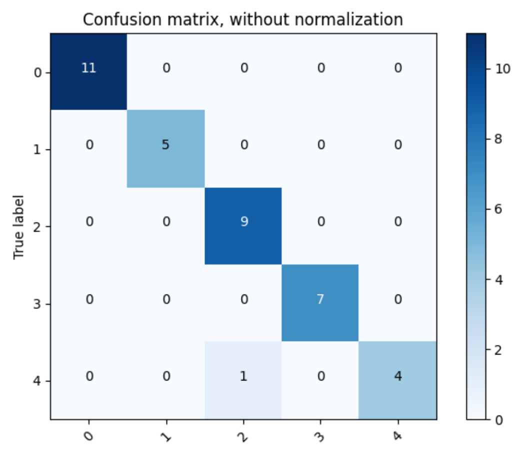
Precision-Recall is a widely used evaluation metric that measures the trade-off between precision and recall. Precision is the proportion of true positives among all predicted positive samples, while recall is the proportion of true positives among all actual positive samples. The F1 score is a metric that takes into account both precision and recall, providing a single value to evaluate model performance.

In the case of face recognition using a CNN model, accuracy can be used to evaluate how well the system can identify and match faces to their respective identities. The accuracy metric measures the percentage of correctly identified faces among all the faces that the system has processed. The higher the accuracy, the better the performance of the face recognition system.



VIII. Results

Confusion Matrix



Precision, Recall, F1 score for each case

```
Confusion matrix, without normalization
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        11
           1       1.00      1.00      1.00         5
           2       0.90      1.00      0.95         9
           3       1.00      1.00      1.00         7
           4       1.00      0.80      0.89         5

    accuracy                           0.97        37
   macro avg       0.98      0.96      0.97        37
weighted avg       0.98      0.97      0.97        37

test los 0.0401
test acc 0.9730
```

IX. Future work

The face recognition attendance system that we implemented has shown promising results in recognizing individuals and managing attendance. However, there is always room for improvement and future work can be done to further enhance the system.

One area for improvement is expanding the system to work with larger datasets and diverse groups of individuals. This will require the development of more robust and accurate deep-learning models to handle the increased complexity of the data. Additionally, we can explore ways to optimize the training process to improve accuracy and reduce training time.

Another potential avenue for improvement is to implement real-time recognition capabilities. This will require the integration of the face recognition system with video streams to enable continuous monitoring and attendance management.

Furthermore, the system can be improved to recognize individuals based on their ID photos rather than just their faces. This will require additional data collection and preprocessing efforts to build a comprehensive database of ID photos.

In summary, there is still significant potential for improvement and expansion of the face recognition attendance system. By incorporating these future work areas, we can further enhance the system's accuracy, efficiency, and usability for a wider range of applications.

X. Conclusions

In conclusion, the Face Recognition Attendance System presented in this project demonstrates the potential of computer vision and deep learning in automating attendance management systems. By leveraging the power of convolutional neural networks, the system can accurately recognize individuals based on their facial features and record their attendance. The use of open-source tools and frameworks such as TensorFlow and Keras highlights the accessibility and ease of implementing sophisticated computer vision applications.

The success of this project also suggests that facial recognition technology can have a wide range of applications in various fields beyond attendance management, such as security systems, identity verification, and personalization. However, the use of such technology raises ethical and privacy concerns that need to be addressed carefully.

Overall, the development of the Face Recognition Attendance System showcases the potential of deep learning and computer vision in solving real-world problems and underscores the need for responsible use and development of such technologies.

XI. Statement of Contributions

- Project concept: Abhiram & Geethika
- Data Collection: Abhiram & Geethika
- Coding: Abhiram & Geethika
- Experimentation: Abhiram & Geethika
- Report Writing: Geethika
- Video Making: Abhiram

XII. References

[1]https://www.geeksforgeeks.org/introduction-convolution-neural-network/

[2]https://downloads.hindawi.com/journals/cmmm/2017/9846707.pdf