Face Recognition Smart Attendance System UsingCnn

Abhishek Kamble
Department of Electronic &Telecommunication Engineering
K.J Somaiya Institute of Technology
Mumbai, India

Ronit Zore
Department of Electronic &Telecommunication Engineering
K.J Somaiya Institute of Technology
Mumbai, India

Ambaji Bodekar
Department of Electronic &Telecommunication Engineering
K.J Somaiya Institute of Technology
Mumbai, India

Harshwardhan Ahire
Department of Electronic &Telecommunication Engineering
K.J Somaiya Institute of Technology
Mumbai, India

***Abstract*— Many applications, such as surveillance, object detection, object tracking, etc., have benefited greatly from the use of convolutional neural networks. There is extensive study for**

CNNs are used for face recognition, a crucial component of surveillance applications. Recent years have seen a significant increase in the application of face recognition technology in university automation systems, smart entry management systems, etc. The procedure of gathering student face data is included in the unique CNN architecture for face recognition system that is proposed in this study. It has been demonstrated experimentally that the suggested CNN design offers 99% accuracy.

The "Smart Attendance Management System (SAMS)," a web-based program that uses face recognition to provide attendance of students in real-time, is also developed utilizing the proposed CNN architecture. The suggested application is simple to deploy and keep up with.

Keywords*— *Facial recognition, Deep Learning, Data Augmentation, Convolutional Neural Networks (CNNs), Smart Attendance Management System (SAMS).

I. Introduction

One of the most used physiological biometric approaches, face recognition, has a decreasingly plausible chance of faking user input. When compared to other physiological bio-metric methods, such as behavioural bio-metric technologies like Keystroke Recognition, Voice Recognition, and Signature Recognition, as well as Fingerprint Recognition, Iris Recognition, and Signature Recognition.

Monitoring techniques like surveillance [1] help to keep track of activities by recording footage of the area where they are positioned. The technology of surveillance cameras has advanced to the point that it may now be very useful for identification, authentication, criminal detection, and other purposes.

The development of deep learning has paved the way for the resolution of many issues that were thought to be challenging for conventional machine learning methods. Particularly, CNNs have transformed how problems involving computer vision and image processing are approached, yielding effective outcomes. Because to its effectiveness in supplying authenticity by automatically detecting the faces, the CNN-based face recognition system has grown in popularity [2], [3].

Contrary to typical machine learning techniques, which require feature selection or feature extraction, convolutional neural network (CNN) designs implicitly extract the features without any prior insight [4]. Due to its ability to provide workable solutions in a generalised manner, CNN has attracted a lot of attention. Issues with computer vision, including picture segmentation, object detection, and face and face recognition [5, 6].

II. Face Recognition Smart Attendance (FRSM)

The suggested web-based application for Face Recognition Smart Attendance using CNNs is described in this section. The difficulties of obtaining sizable train and test datasets for real-time applications are also discussed in this part.

Face Recognition Smart Attendance is the name of the developed web-based programme for attendance posting that is based on face recognition [7]. Python and the flask infrastructure are used in its development.

The main components of the application are:

1) Front-End Interface (Camera)

2) Server

3) CNN Model

4) Database

5) End Result (Display Screen)

A. *front-end interface*

Front end interface is a web page that uses the OpenCV library to receive video frame data from the camera. The back-end of the programme, or server, is what keeps the front-end interface connected. model, store, and monitor. The front-end interface sends the server a video frame, the server detects the face, the model gets that input, and the model forecasts the label. The server confirms whether the attendance has already been uploaded. If not, the date and attendance are entered into the database. The display screen, which is a web page, is used to display the details of the associated student. The images of the application's front-end UI and final product are shown in Fig. 1.

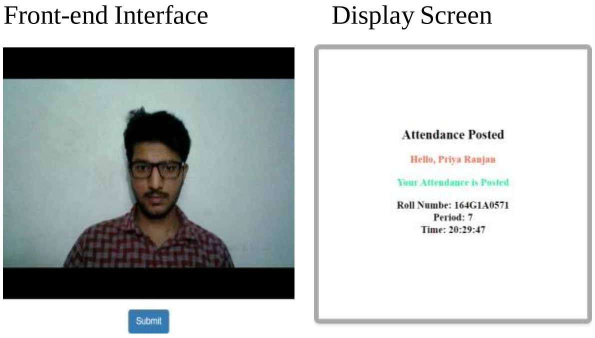


Fig.1: A Sample input image and output

B. Data Collection

An automated system is used to gather the students' face data. It uses the system camera to capture video frames, which it then modifies into a dataset by running the series of processes listed below:

- Locate the face in the frame of the video Frame.
- Remove the face from the photo and turn it into a grayscale picture.
- Attach the label w.r.t to the class of the image andwrite onto a *csv* file.

the representative images based on the data gathered for each subject. In order to comprehend the patterns in the data, a three-dimensional scatter plot is created using Principal Component Analysis (PCA), which reduces the dimensions of the data to three.

C. Data Augmentation

Collecting a significant amount of data in the actual world is a challenging and tiresome undertaking. Large volumes of training data are needed to obtain high accuracy. hence, data

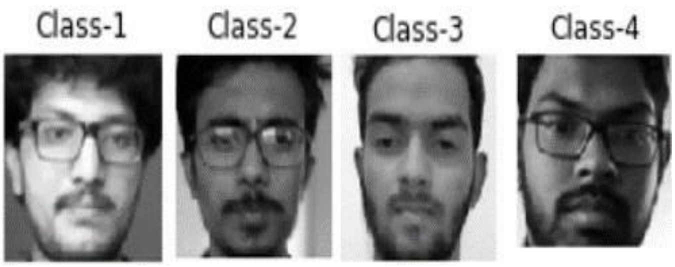


Fig. 2: Sample Images of the Dataset with their Class Labels

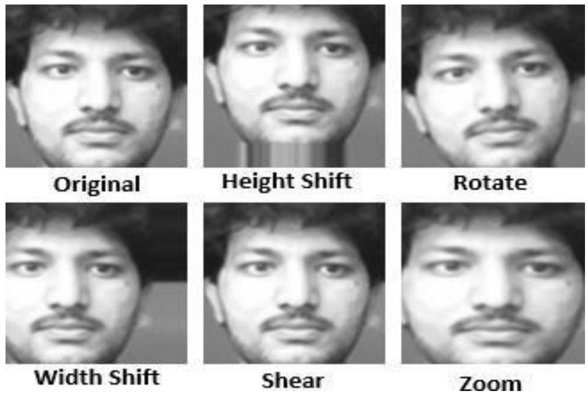


Fig.3: Sample Images after Data Augmentation

By modifying the existing data, additional samples can be produced using the augmentation approach. innovative Face-specific data augmentation for unrestricted face recognition was presented more recently. By using the subsequent methods, synthetic images are randomly created in the current study.

- Zoom
- Shear
- Height shift
- Rotation
- Width shift

The synthetic images generated after data augmentation areshown in Fig-3.

III. Proposed CNN

For this specific application, FRSA, the proposed CNN architecture is presented in depth in this section. The explanation of the many layers that were utilized to construct the CNN model is also included.

The proposed CNN architecture consists of 20 layers whichincludes:

- Two Dimensional Convolutional Layer (Conv2D)
- Batch Normalization Layer
- Max Pooling Layer
- Dense Layer

The batch normalization layer is used to normalize the input and also overcome the issue of disappearing gradient [8] and exploding gradient. The two dimensional convolutional layer is used to retrieve the features from the previous input. The dropout layer is used to prevent the over fitting issue while the max pooling layer is used to lower the input's dimensionality.

The architecture uses a grayscale image with the shape (100, 100, 1) to forecast the image's class. The CNN has a total of 7,658,629 parameters, of which 7,656,197 are trainable and

2,432 are not. Table-I displays a full explanation of the architecture.

TABLE I: Architecture of the proposed CNN

Layer	Output Shape	#parameters
Conv2D	(None, 98, 98, 64)	640
Batch Normalization	(None, 98, 98, 64)	256
Conv2D 1	(None, 96, 96, 64)	36928
Batch Normalization 1	(None, 96, 96, 64)	256
Conv2D 2	(None, 96, 96, 64)	102464
Batch Normalization 2	(None, 96, 96, 64)	256
Max Pooling2D	(None, 48, 48, 64)	0
Dropout	(None, 48, 48, 64)	0
Conv2D 3	(None, 46, 46, 128)	73856
Batch Normalization 3	(None, 46, 46, 128)	512
Conv2D 4	(None, 44, 44, 128)	147584
Batch Normalization 4	(None, 44, 44, 128)	512
Conv2D 5	(None, 44, 44, 128)	409728
Batch Normalization 5	(None, 44, 44, 128)	512
Max Pooling2D 1	(None, 22, 22, 128)	0
Dropout 1	(None, 22, 22, 128)	0
Conv2D 6	(None, 20, 20, 256)	295168
Batch Normalization 6	(None, 20, 20, 256)	1024
Max Pooling2D 2	(None, 10, 10, 256)	0
Dropout 2	(None, 10, 10, 256)	0
Flatten	(None, 25600)	0
Dense	(None, 256)	6553856
Batch Normalization 7	(None, 256)	1024
Dense 1	(None, 128)	32896
Batch Normalization 8	(None, 128)	512
Dense 2	(None, 5)	645

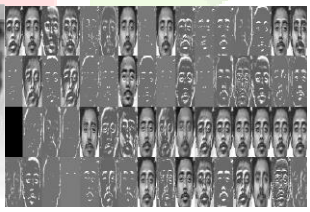
During training with respect to an input image, the intermediate layer visualisations of the convolutional layers Conv2D 1, Conv2D 2, Conv2D 3, Conv2D 4, and Conv2D 5 of the CNN model.

IV. Results

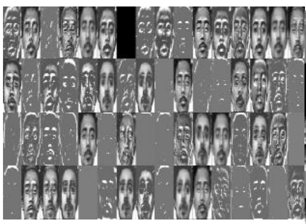
Tensor flow, an open source Python library, is used to create the proposed Convolution Neural Network (CNN) [9]. The system is configured as follows when the experiments are run.

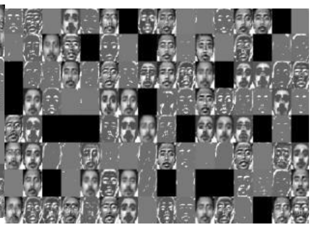
- The employed GPU is a 1X Tesla K80 with 2496 CUDA cores and 12GB GDDR5 VRAM.
- 1X single core hyper threaded Xeon processors, 45MB Cache, 12.6 GB RAM, and 320 GB disc are employed as the central processing unit (CPU).

Input Image conv2D_1

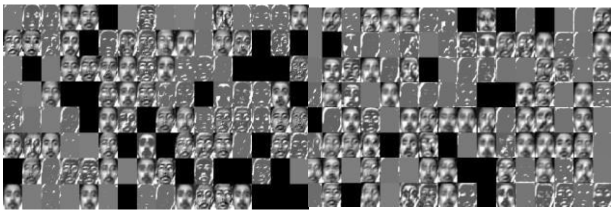


Conv2D_2 Conv2D_3





Conv2D_4 Conv2D_5



V. Conclusion

In this work, CNN's function in face recognition and its adaptation for attendance posting are formally introduced. TheFace Recognition Smart Attendance web (FRSA)application's workflow is described in depth. The process of gathering data and enhancing existing data to create the CNN model is outlined. This research also suggests a revolutionary CNN face recognition model that will be employed in the creation of FRSA. The experimental findings demonstrate the effectiveness of the suggested CNN model and the FRSA online application. FRSA is simple to deploy and keep up with. The paper's future objectives include the development of a dependable programed for real-time smart attendance management for a larger number of students.

References

[1] K. Sun, Q. Zhao, J. Zou, and X. Ma, “Attendance and security system based on building video surveillance,” in *International Conference on Smart City and Intelligent Building*. Springer, 2018, pp. 153–162.

[2] M. Coşkun, A. Uçar, Ö. Yildirim, and Y. Demir, “Face recognition based on convolutional neural network,” in *2017 International Conference on Modern Electrical and Energy Systems (MEES)*. IEEE, 2017, pp. 376– 379.

[3] M. He, J. Zhang, S. Shan, M. Kan, and X. Chen, “Deformable face net for pose invariant face recognition,” *Pattern Recognition*, vol. 100, p. 107113, 2020.

[4] G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Z. Li, and T. Hospedales, “When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition,” in *Proceedings of the IEEE international conference on computer vision workshops*, 2015, pp. 142–150.

[5] M. Swapna, Y. K. Sharma, and B. Prasad, “A survey on face recog- nition using convolutional neural network,” in *Data Engineering and Communication Technology*. Springer, 2020, pp. 649–661.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779– 788.

[7] Z. Zhiyao and B. M. Di, “Multiplet selection: a practical study of multi- faces recognition for student attendance system,” in *Proceedings of the 2nd International Conference on Image and Graphics Processing*, 2019,pp. 42–46.

[8] J. Kolbusz, P. Rozycki, and B. M. Wilamowski, “The study of architec- ture mlp with linear neurons in order to eliminate the “vanishing gra- dient” problem,” in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2017, pp. 97–106.

[9] L. Yuan, Z. Qu, Y. Zhao, H. Zhang, and Q. Nian, “A convolutional neural network based on tensorflow for face recognition,” in *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. IEEE, 2017, pp. 525–529