

Smart attendance system using Convolution Neural Network and Image Processing

Senigala K. ChayaDevi, and Vamsi Agnihotram

Abstract—Smart attendance maintenance system has been a research topic from past few decades. An algorithm has been proposed in this paper using the Convolution Neural Network (CNN) and Image processing. Image recognition is playing an important role in modern living like driver assistance systems, medical imaging system, quality control system to name a few. An Artificial Neural Network along with image recognition used to enhance the reliability of the system. One such update used here is CNN. Deep learning is an emerging technology hence opted to implement the smart attendance system in it. The implementation basically consists of three components: 1) Face scanning and detection using HAAR cascade method 2) Training the CNN-ANN model 3) Recognize the face back and update the attendance. The main motivation of our work is to merge three of the emerging technologies: Machine learning, Image Processing and IOT. Key advantage of this implementation is that a deep learning model increases its accuracy with more epochs of training, and It optimizes the run time.

Index Terms—Deep Learning, HAAR Cascade Classifier, Convolution Neural Network (ConvNet), Artificial Neural Network (ANN), Internet of Things (IoT).

I. INTRODUCTION

Smart attendance based on face recognition is one of the non-intrusive methods. In many countries including India, monitoring student attendance, in almost all the educational sectors be it schools, institutes, colleges or universities, is still a tedious, time-consuming, error-prone, and manual process. The above fact is hard to assimilate in the present era where technological developments are on the cutting-edge and technology-driven solutions have served as the primary key to unlock doors to trouble-free quality life. Thus a technologically new way has to be executed in to take attendance and store it. We chose deep learning to build a model with the prime reason being a deep learning model (may it be artificial, convolutional or recurrent neural network) learns epoch by epoch and goes on getting perfect. At a stage, a CNN is proved to give out 90.83 percent accuracy, given by [10] on classification tasks, which is almost equal to a human brain's accuracy in prediction and classification. Because finally, it is the output accuracy that matters in a digital project. Artificial neural networks (ANN) or connectionist systems are computing systems vaguely inspired by the biological neural networks that

constitute animal brains. Such systems "learn" to perform tasks by considering examples, generally without being programmed with task-specific rules. For example, in image recognition, they might learn to identify images that contain cats by analysing example images that have been manually labelled as "cat" or "no cat" and using the results to identify cats in other images. They do this without any prior knowledge of cats, for example, that they have fur, tails, whiskers, and cat-like faces. Instead, they automatically generate identifying characteristics from the examples that they process.

Image processing was one of the major areas for most of the researchers to work on. It has been playing an increasingly larger role in the modern life like driver assistance system, medical imaging system, quality control system to name a few. Artificial Neural Network models are extensively used for the above purposes due to their reliable success. One such update used here is the convolutional neural network (CNN, or ConvNet). Digital image processing techniques help in the manipulation of digital images through the use of computers. The first step in digital image processing is capturing of documents, images and pages with a scanner to obtain a raw file (ex: raw file format). This raw file coming from the scanner is a file that has not been altered. Say for example we are using a video camera and capturing frames which in turn is the raw file for D.I.P. However, to get an optimized workflow and to avoid losing time, it is important to process images after the capture. Face detection and recognition are popular sub-parts of image processing. Reframing into face boxes makes it easier for the model to detect the face components (like eye, mouth locations).

Discussing the prior implementations, some include: 1) Implementing access cards and biometrics. Biometrics is an automated method of identifying a person or verifying the identity of a person based on a physiological or behavioural characteristic. Examples of physiological characteristics include hand or finger images, facial characteristics. Biometric authentication requires comparing a registered or enrolled biometric sample (biometric template or identifier) against a newly captured biometric sample (for example, captured image during a login). As shown in the picture below, a sample of the biometric trait is captured, processed by a computer, and stored for later comparison. But the model had limitations that it is equally time consuming as students have to come one-by-one for marking attendance.

Published on May 30, 2020.

S. K. ChayaDevi, Asst. Professor, Department of Information Technology, Vasavi College of Engineering.
skchayadevi@gmail.com

V. Agnihotram, Department of Information Technology, Vasavi College of Engineering.
vamsiagnihotram98@gmail.com

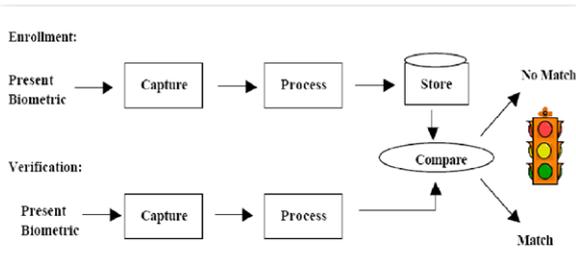


Fig. 1. Block Diagram (Biometric System)

Another solution based on the barcode system is proposed in [4] where students have to show the unique barcode on their ID card to mark attendance and that also is a very time-consuming task. The major drawback of this system is the barcode prone to get damaged. One of the popular method based on RFID [1], refers to a technology whereby digital data encoded in RFID tags or smart labels (defined below) are captured by a reader via radio waves. This technique is similar to barcoding in that data from a tag or label are captured by a device that stores the data in a database. It, however, has several advantages over systems that use barcode asset tracking software. The most notable is that RFID tag data can be read outside the line-of-sight, whereas barcodes must be aligned with an optical scanner. A key limitation has then been found out during this implementation is that Materials like metal & liquid can impact signal. Next solution includes “Attendance management using Bluetooth Low Energy and Android” given by [2]. It uses the Bluetooth Low Energy technology of beacons which communicate with an android application. The application is used to collect the data from the sensors and store it according to the dates. But this approach failed due to multiple reasons. An approach based on fingerprint scanner is proposed in [3]. It has a drawback that students have to stand in a queue to mark their attendance and that takes a very long time for large number of students. A similar approach is speech-based attendance system [5]. Then comes a more technological implementation - NFC (Near-Field Communication) based attendance system [6] is proposed. Disadvantage of this model is it being Expensive to implement. In [11], face recognition based on PCA (Principal Component Analysis) is proposed. The process of face recognition is divided into four steps.

Drawbacks of the previous discoveries have been covered in the implementation explained by this algorithm. This paper highlights the importance of pre-trained neural networks as well as the significance of Deep Learning used in the field of Academics and Advancement which is implemented in python. Smart Attendance Systems involves the image (face) detection and analyses the data accurately. This approach solves the time-consuming traditional method of attendance system and paves way for new advanced technologies. In [7], face recognition Open CV module is proposed.

The process of face recognition is a single algorithm in our implementation which starts by setting up a video frame. Required frames are extracted from the video, which contains only the face part by using cascade classifier as HAAR cascade [8] already as an XML document embedded in it has details on facial features. Then comes the analysis

phase where a build a CNN model. It, being a class of deep neural networks, most commonly applied to analysing visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics. They have applications in image and video recognition, recommender systems, image classification, medical image analysis, natural language processing, and financial time series. CNNs are regularized versions of multilayer perceptrons.

The model is then trained with the scanned set and also can predict the accuracy using the model. Predict function of the keras library. This approach solves the time-consuming traditional method of attendance system and paves way for new advanced technologies.

The remainder of this paper is organized as follows. Section II Introduces the block diagram and methodology of our system. In Section III, we propose the implementation of our smart attendance system with screenshots of each stage. The conclusions and future work are presented in Section IV. References of citations used have been specified in section V.

II. PROPOSED METHODOLOGY

A. Block Diagram

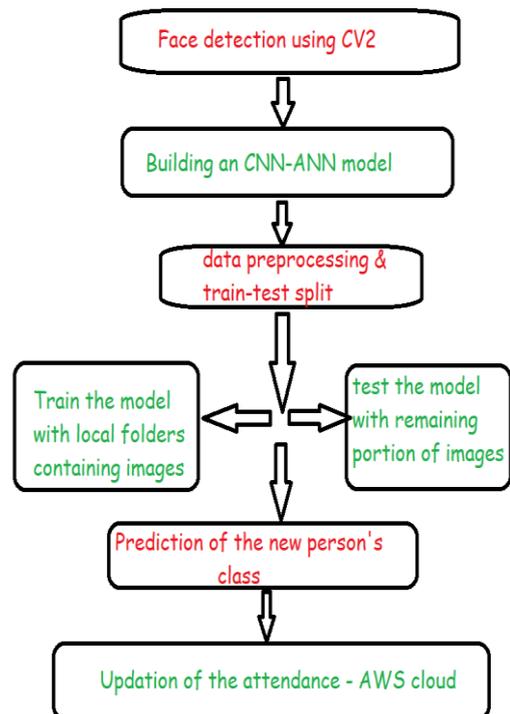


Fig. 2. Block Diagram of algorithm

The block diagram of our smart attendance based on face recognition is mentioned in the Fig. 2.

B. Complete Methodology

For implementing the smart attendance system using face recognition system, we have to follow the following steps in the same order. Those steps are as follows:

1. Enrollment of students (face detection)
2. Data pre-processing and building the model.

3. Train the CNN-ANN model, apply Test image
4. Face recognition
5. Face Encoding (displaying the label of the face)
6. Store attendance in database

1) Enrollment:

The person will be enrolled in the database using their name and 150 face images(with 7:3 train-test split). The information will be stored in the folder. The process of enrollment includes:

- Making a folder with person name entered as the folder name.
- Capture image from pi or USB camera.
- Cut down the face components and if needed figure out the eyes using HAAR cascade face and eye detection steps (as explained in the introduction). Reframing into face boxes makes it easier for the model to detect the face components (like eye, mouth locations).
- The image of the person is captured using a high definition camera and stored in the folder given by the path specified in cv2.imwrite command.

The internal implementation of the cascade goes as follow: Firstly, the detector would load the classifier 6 and determine it is not empty. If it is, then it simply exits with an error message. Then the image in question is loaded and the same procedure is followed. Then the classifier is applied to the image, which outputs an array of rectangles, which correspond to the detected positions of the objects, in this case a face. The program would then draw bright blue rectangles in the locations of the detections and also add a text to the image(if necessary).The rectangular Haar features will be generated to detect various parts like white and black portions of a grey scale image also. A rectangular frame will be produced as a border that helps to crop the face alone from the entire image. It is suitable to detect multiple faces in a given image. It is already mentioned that the pre-processing step converts the RGB image to grey scale image. The pixels which were black were stored, and they were subtracted from the total number of white pixels. The output was compared with a threshold and if the features are matched, then the objective like face will be detected.

2) Data pre-processing and building the model:

CNN - In deep learning, a convolutional neural network, is a class of deep neural networks, most commonly applied to analyzing visual imagery. The main advantage of CNN compared to its predecessors is that it automatically detects the important features without any human supervision. For example, given many pictures of cats and dogs, it learns distinctive features for each class by itself. As any CNN model generally has 4 stages or layers, namely:

- a) Convolution
- b) Max pooling
- c) Flattening
- d) Full connection

In the project, we have implemented two convolution layers with 32 different feature maps or filters with 3/3 dimension. Input image shape = (64,64) and activation function: ReLu (rectified linear unit function). Implemented two max pooled layers with pool size = (2,2). The output

from the second max pooled layer is flattened.

CNN KERNEL:

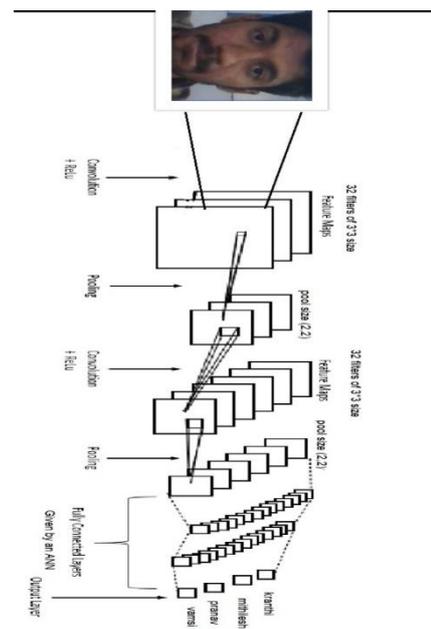


Fig. 3. CNN network figure

The reason why we use ReLu or the rectified linear unit function is, as it is a piecewise linear function that will output as-it-is if the input is positive, otherwise it will output zero.

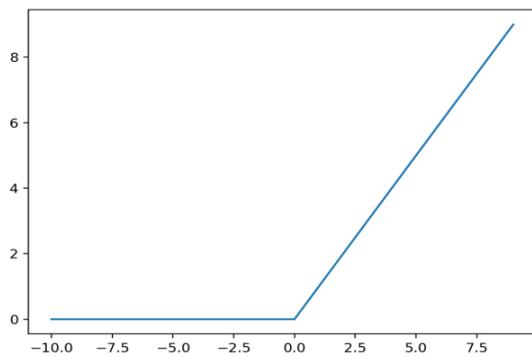


Fig. 4. ReLu activation function

Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "fully-connectedness" of these networks makes them prone to overfitting data.

We inserted three ANN layers in order to classify the flattened data from the CNN. Out of which,

- Two are hidden layers with: A random output dimension (> 100 for more accuracy and less loss %), ReLu as the activation function, and initialized random weights.
- One Output layer with output dimension equal to the number of classes that we wanted to segregate, softmax as the activation function as it is a categorical o/p.

This combination of number of hidden layers and activation functions has given the highest accuracy, and therefore been fixed.

Softmax is the activation function used for the output as it

turns numbers aka logits into probabilities that sum to one. The function outputs a vector that represents the probability distributions of a list of potential outcomes. It is usually the last layer in the classification task.

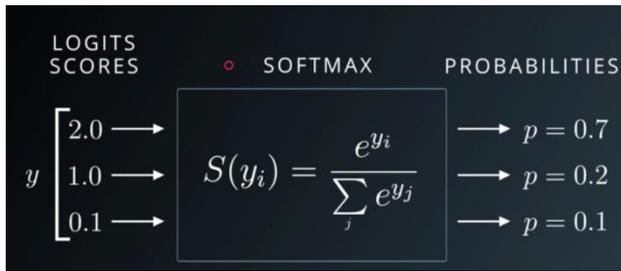


Fig. 5. Softmax activation function

3) Train the CNN-ANN model, apply Test image:

The training algorithm itself has certain components:

- Importing keras libraries
- Data pre-processing and image rectification
- Train-test split with 70-30 division.
- Compiling and running the model with 50 epochs for more accuracy

model.compile function uses optimizer as 'adam'. 'Adam' suggests that a stochastic gradient descent function has been used instead of batch or mini-batch gradient descent because the dataset is big and input image size and number of epochs make it even bigger. In Stochastic Gradient Descent, a few samples are selected randomly instead of the whole data set for each iteration. Suppose you have a million samples in your dataset, so if you use a typical Gradient Descent optimization technique, you will have to use all of the one million samples for completing one iteration. This problem is solved by Stochastic Gradient Descent. In SGD, it uses only a single sample, i.e., a batch size of one, to perform each iteration. The sample is randomly shuffled and selected for performing the iteration. Hence, if a SGD has been used back propagation becomes easier, as cited in [9].

Two of the other parameters which influence the output are loss and metrics. Loss is 'categorical_crossentropy', metrics defines which factors have to be examined while back propagating and updating the initial weights. I have specified 'accuracy' as the metrics because it is of the most priority.

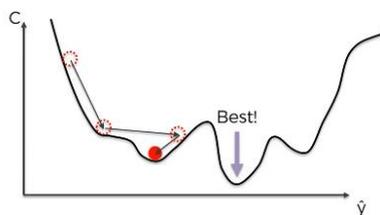


Fig. 6. SGD: Probability distribution function for whichh 'stochastic' predicts the best outcome

After training the system on the database our system is placed in the classroom. The camera is adjusted in such a way that a good amount of lighting falls on the in-comer and the face is captured with minimum sheer angle and horizontally streamlined. A test image is loaded from the test set of every class and checked for more train-perfection

of the model. For this we require high definition camera, say a USB camera. Further stages involve :

- Train-test split
- Compile the model
- Fit transform the model to the dataset (epochs : 50)
- Save the model with.h5 extension

4) Face recognition:

This stage can be implemented using the Open CV module, but this time there is no loop as in the detection algorithm (which needs 150 images in a loop for creating the dataset), it just needs one frame capture. The face component is isolated from the frame and now this image is fed as an input to the CNN layers then to the ANN layers for feature extraction. The extracted features are compared with the trained dataset in order to give out the class of the input face and then followed by encoding.

5) Face Encoding (displaying the label of the face):

For the newly arrived person the model has to encode to which class does he belong to. This prediction is of two stages

- Firstly, the model outputs class index i.e to which class does the person belong to. Through which data is updated in the cloud database.
- Second one gives the accuracy of the newbie with the chosen class.

6) Store attendance in database:

After the CNN model recognizes the in-comer as a temporary action I have created a list of size equal to strength of the class. The index corresponding to the recognized person is incremented as and when he encounters the camera.

As a permanent implementation - A CSV file is created using python with name as one column and attendance in the next column and it is sent to the faculty email id through SMTP protocol with file name as the date on which the attendance is taken.

III. RESULT AND OBSERVATIONS

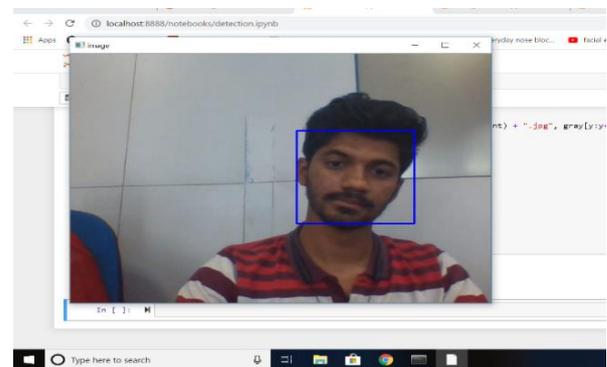


Fig. 7. face detection

Our system has been tested by taking ~ 30 persons' images and creating the dataset. Intake of each scan goes as follows:

150 image frames of a person are captured within 5 seconds and stored in the newly created folder given by input name. The images are automatically saved in 7:3 ratio

in the train and test directories respectively i.e 105 in the train set and 45 frames in the test set.

After the train test split the folders look like this :

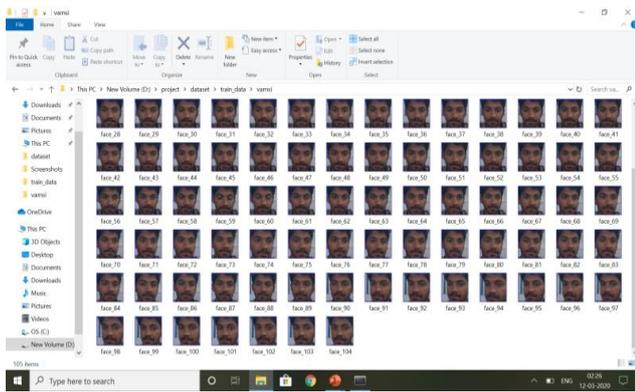


Fig. 8. Storing frames into dataset

Now the CNN-ANN model gets trained with all the train and test images of different classes and indices are given to each member. The classes are arranged in alphabetical order automatically:

```
Found 315 images belonging to 3 classes.
Found 135 images belonging to 3 classes.

In [7]: print(x_train.class_indices) # print class indices
{'kranthi': 0, 'pranav': 1, 'vamsi': 2}
```

Fig. 9. Trained CNN model

When the new face arrives it predicts in two stages, as discussed earlier:

```
[[0.33287457 0.33428073 0.3328448 ]]
[1]
[0, 2, 0]
```

Fig. 10.

In Fig. 10:

- The first line indicates attribute accuracy of the input frame belonging to each class.
- Second line gives to which class does the input frame belongs to.
- Final line gives the updated list after attendance calculation.

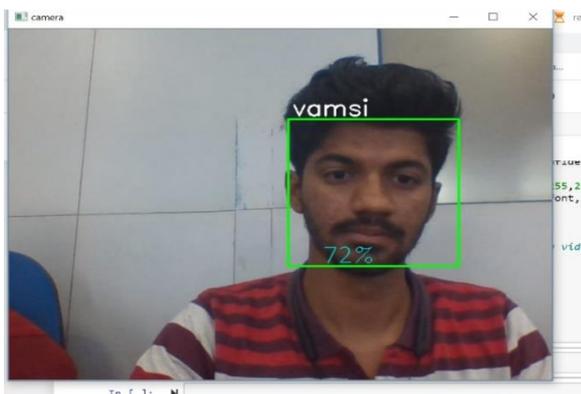


Fig. 11. Recognizing face

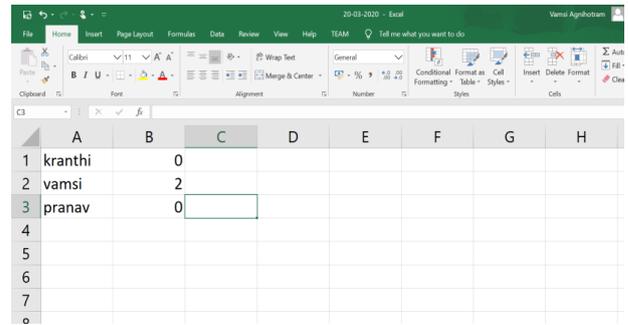


Fig. 12. Storing data to a .csv file

The list data finally gets updated in the.csv file after completion of the day.

If an unknown person encounters the camera the imwrite function gives the output as “shape not found” and a very low accuracy which does not even cross the cut-off value is displayed, as shown in Fig. 13.

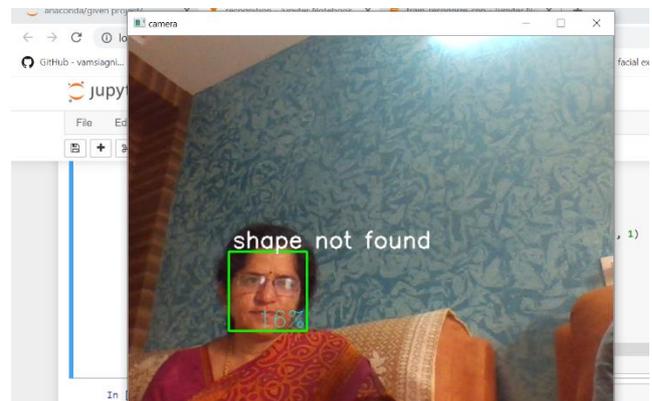


Fig. 13. Rejecting a new (un-trained) face

TABLE I: RESULT COMPARISON TABLE OF PREVIOUS WORK WITH THIS PAPER 'S IMPLEMENTATION

Model	Drawback	Accuracy (maximum)
Barcode scanning system	barcode prone get damaged	50%
RFID	Materials like metal & liquid can impact signal	70%
NFC	Expensive	65%
OUR MODEL	Takes time in training (one-time investment)	75%

IV. CONCLUSION AND FUTURE WORKS

The main motto of our work is to merge three of the emerging technologies: Machine learning, Image Processing and IOT. The smart attendance system using CNN and open cv is proven to be an efficient system for classroom attendance. This system is non-intrusive and it reduces the chances of proxies and fake attendance. After all, a neural network when trained heavily becomes as accurate as a human brain. We have implemented the same successfully, in a classroom and ended with a great accuracy of 72 percent, which further upgrades by tutoring the model. Our system setup is a very simple and easy to use, it requires a mere camera module and a processor to perform facial recognition. Also, our system can be implemented on the raspberry pi with internet-enabled.

V. FUTURE WORK

Hence as further implementations we install an IOT device (miniature), say a pi or a USB camera can be connected in the front end for precise capturing as we cannot install a computer at each place. Accessing a cloud database for storing the attendance register file so that it cannot be tampered. The performance goal is to improve the recognition rate of our system when the faces of the students are half-covered or when they are partially visible.

ACKNOWLEDGMENT

We sincerely thank Mr. B.V Doraswamy, scientist 'F', DMRL, DRDO for assistance in Convolution neural networks, and my family for support. With pleasure, we record our sincere gratitude to our Head of the Department Dr. K. Ram Mohan Rao, Department of Information Technology, Vasavi College of Engineering, for permitting us to carry out this research.

REFERENCES

- [1] M. K. Yeop Sabri, M. Z. A. Abdul Aziz, M. S. R. Mohd Shah, M. F. Abd Kadir, "Smart Attendance System by using RFID", Applied Electromagnetics 2007. APACE 2007. Asia -Pacific Conference on, 4-6 Dec. 2007.
- [2] R. Apoorv, P. Mathur, "Smart attendance management using Bluetooth Low Energy and Android", 2016 IEEE Region 10 Conference (TENCON), 2016.K.
- [3] N. I. Zainal, K. A. Sidek, T. S. Gunawan, H. Mansor, M. Kartiwi, "Design and Development of Portable Classroom Attendance System Based on Arduino and Fingerprint Biometric", proc. of The IEEE 5th International Conference on Information and Communication Technology for The Muslim World (ICT4M), pp. 1-4, 17-18 Nov. 2014.

- [4] K.Lakshmi Sudha, Shirish Shinde, Titus Thomas, "Barcode based student attendance system" International Journal of Computer Applications (0975 – 8887) Volume 119 – No.2, June 2015.
- [5] S. Dey, S. Barman, R. K. Bhukya, R. K. Das, B C Haris, S. R. M. Prasanna, R. Sinha, "Speech biometric based attendance system", National Conference on Communications, 2014.
- [6] M. Baykara, U. Gürtürk and E. Karakaya, "NFC based smart mobile attendance system," 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, 2017, pp. 937-941, doi: 10.1109/UBMK.2017.8093571.
- [7] Mrs. Madhuram.M, B. Prithvi Kumar, Lakshman Sridhar, Nishanth Prem, Venkatesh Prasad "Face Detection and Recognition Using OpenCV".
- [8] Sander Soo Institute of Computer Science, University of Tartu "Object detection using Haar-cascade Classifier". <https://pdfs.semanticscholar.org/0f1e/866c3acb8a10f96b432e86f8a61be5eb6799.pdf>
- [9] Sun Ichi Amari "Backpropagation and stochastic gradient descent method" National Conference on Deep learning, 2017.
- [10] Zhipeng yu and Jie yang's "Brain MRI segmentation with patch-based CNN approach" conference paper, July 2016. https://www.researchgate.net/publication/308496682_Brain_MRI_segmentation_with_patch-based_CNN_approach/citation/download
- [11] Wagh, J Priyanka, Agruti Chaudhari, Roshani Thakare, Shweta Patil, "Attendance system based on face recognition using eigen face and PCA algorithms", Green Computing and Internet of Things (ICGCIoT) 2015 International Conference on, pp. 303-308, 2015.



S. K. ChayaDevi working at department of Information Technology, Vasavi College of Engineering, Hyderabad. Graduated from Regional Engineering College Warangal, finished Post Graduation at JNTUH, College Of Engineering Hyderabad and currently pursuing PhD in Medical Image Processing from JNTUH College Of Engineering Hyderabad. Published 13 International and National Journals and conferences.



Vamsi Agnihotram, Student of department of Information Technology, Vasavi College of Engineering, Hyderabad. Place of birth is Hyderabad, Telangana, India.