

Contemporary Study of Face Recognition Systems for Attendance Autonomous Systems

Manu Anmol Singh¹ Vinay Kukreja²

^{1,2}Chitkara University Institute of Engineering and Technology, Chitkara University, India

Abstract

Purpose: Marking the arrival of the age of automation, the sole purpose of this study is to convert the most widely used approach of manual attendance into an autonomous and automatic one.

Values: Fundamental values depicted from this study are to save time and spreading sincerity towards work and studies. This system will enhance the accuracy of an attendance system and its result could lead to students becoming more sincere towards their university lectures.

Design Methodology: To automate the attendance system, LBPH technique has been used for facial recognition. The design system also includes IDE for coding and different packages in order to introduce easy accessibility for face operations.

Findings: The technology is being used to deliver an efficient experience for users. The attendance system not only aids in marking the attendance but also help in surveying various other factors like the time at which the students enter the class and giving real-time feedback about the sincerity of students with respect to grades and attendance.

Keywords: *Face recognition, Face detection, LBPH, Principal Components, Haar classifier.*

1. Introduction

Attendance is a very important tool to keep a check on the students. Attendance is noted down manually in most of the institutes which may lead to inaccuracy of data; other students generally mark attendance for their fellow classmates which leads to false data gathering and later on this result in false statistics. Also, noting down attendance can be really a time-consuming task which may ultimately lead to less teaching and learning time for teachers and students. Many institutes follow a biometric fingerprint attendance approach and other just

relies on traditional paper-based approach. Our system uses face detection and face recognition approach for marking down the attendance of the students.

This approach will increase the accuracy of data about the students attending the class, effects of attendance on the academic performance and preventing huge time consumption carried with manual paper-based attendance marking (Jha, A., 2007).

This autonomous system is specifically designed to overcome the traditional attendance system and revolutionalise the classrooms with automation. Our approach is to come up with an autonomous system that marks down the attendance of candidates, saves time and aids in surveying the class performance and other related aspects. (Chintalapati, S., &Raghunadh, M. V. 2013)

2. Platform Used

This system uses special kind of platforms in order to enhance the mobility and functionality of the software. The IDE provides access to a huge amount of packages. Packages allow users to use a vast amount of pre-defined functions for an efficient coding environment. Following platforms are used to implement face detection and face recognition functions :

2.1 Pycharm

Pycharm is an IDE which is used for the python language. The advantage of using Pycharm over python is that it enhances the lucid implementation of packages; New packages are easy to include in Pycharm and errors are easy to identify.

2.2 OpenCV

OpenCV which is also known as 'Open Source Computer Vision' is a library of programming functions used in python or other programming languages which gives user accessibility to perform a number of operations over an image. These operations include; Reading and writing of the image, Detecting faces, modifying image quality and lots more. OpenCV is simple to learn and is free to use (Bradski, G., &Kaehler, A.,2008).

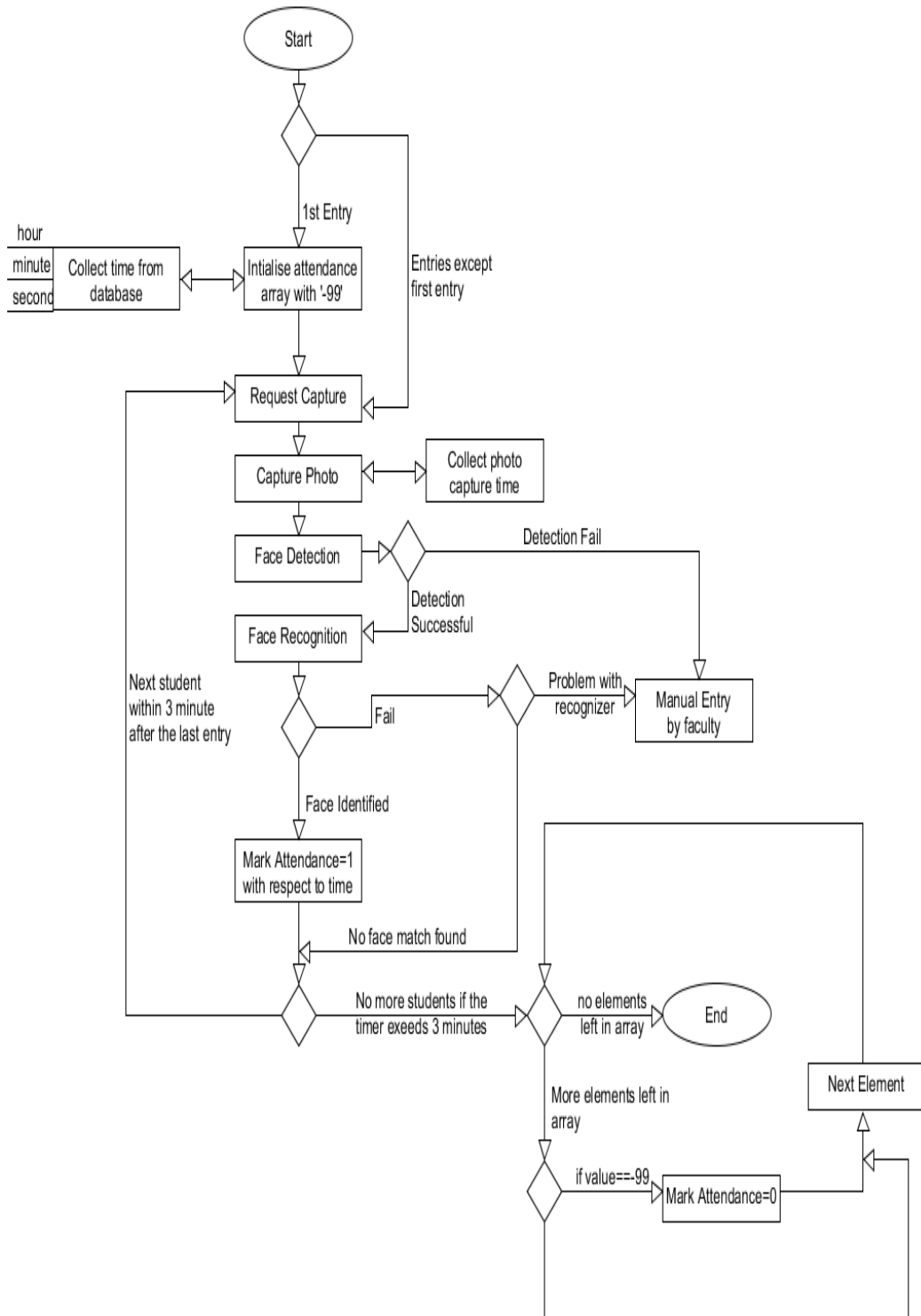
2.3 NumPy

NumPy stands for 'Numerical python' which is a package often included in python or Pycharm in order to access a powerful N-dimensional array or a multidimensional container

for data. This is used to implement the Local Binary pattern histogram technique. (Walt et al., 2011)

3. The Autonomous Attendance System

3.1 Construction of the system



In this paper, our system consists of a camera installed in the classroom. Every student needs to stand in front of the camera after entering the classroom. The camera will capture each student's face one by one and the image captured will be processed in the following way in order to ensure the accuracy of time and attendance:

1. When the algorithm starts, it is important to determine the first entry so that the array is initialized only once. If any other entry is identified later in the process then control directly flows to 'Request Capture' state. If the array is re-initialized the second time then all the attendance entries of the first attempt would be deleted.
2. First, an array is initialized with a value of '-99' for all the students. Second, the lecture starting time and ending time are collected from the database in the format of 'hour:minute:second'. This enables the system to mark attendance of student under the appropriate time period. During the algorithm, the array is modified with a single value for each student which indicates whether the individual is present(1) or absent(0).
3. If there is any new face found, then the control flows into the 'Request capture' state which later on captures the face in the form of a photo.
4. In this step, the time at which the photograph of a student is captured is collected and taken into account in order to determine the time at which the student has entered the lecture hall. This gives access to lecture wise marking the attendance of an individual. Time collection enables the system to determine the specific lecture that should be marked as present for the particular student. (Mike,2010)
5. The face detection unit will detect the presence of a face in the image. A computer program classifies a face image as a positive (+) image and a nonface image as a negative(-) image. The image is processed in grayscale in this unit. If the face is detected then the process moves on to the next step. (Balcoh et al., 2012).
6. This is the most salient step of the Autonomous Attendance System. In this step, the program recognizes a specific face and the name associated with it. The computer recognizes the face in the following way:

6.1 *Data gathering*-Program extracts unique characteristics from a face.

6.2 *Data comparison*-Program compares the collected unique features with the pre-saved unique pictures.

6.3 *Face recognition*- Program finally comes to the verdict about the specific face recognized.

7. All faces are traversed and If the face is recognized by the program then an attendance is marked in a binary indication of '1' with the time taken into account for marking the attendance for a specific lecture. If the face detection or recognition process fails by any chance, there is an alternative for marking down the attendance by the approval of teacher for that specific lecture. The system goes into the state of *Manual Entry*; all those students encountering an error in their face detection may manually mark the attendance, this will require teacher verifying each student encountering the problem. Every teacher will be provided with a unique 'IP' link which will be directly associated with the attendance database, this 'IP' will only be accessible to the faculty members. In the end, teachers will manually enter the roll number of students and the system will mark the attendance to '1'. (Balcoh et al., 2012)
8. There is a time window of 3 minutes after the attendance of each student is marked. The time window enables the system to accept more faces for marking the attendance. If the timer exceeds, the control of the algorithm flows into the state where there are no more students left. If the time exceeds and there are still more entries to be marked then the algorithm is re-initialized with the existing values of the array; if there is any new entry (not the first entry) then the algorithm will directly start from 'Request Capture' state.
9. In the end, When the attendance process is done then the array of students will be traversed and the remaining fields will be marked as 0 or absent and the algorithm will encounter the end state.

3.2 Preparing Data

Before implementing the system, it is important to prepare the train-data or to acquire the photographs of each and every student and store it to the directory in order to identify each individual in the future. At least 2 photographs of each student in the directory should be saved manually. Separate folders are created for every single student. Folders starting with the label '1'.

3.3 Capturing face and time

In order to make the system work, each student needs to stand in front of the camera so that the camera captures the face and extract unique characteristics from it. On the other hand, the system will also extract the time at which the photo was captured using an 'EXIF' function in

python. Acquiring the creation time of a specific picture will allow the system to mark attendance of an individual for a particular lecture.

3.4 Face Detection

A number of methods for face detection are used in order to figure out whether the image has a face in it or not. Among them, there are two trained classifiers namely 'Haar classifier' and 'Local Binary Pattern classifier' for determining the face in the image. Our system utilizes the LBP classifier for detecting the face which processes images in grayscale. This approach helps the system to identify whether an image contains a face in it or not.

LBP forms a vector that differentiates between a face and a non-face image. LBP processes 9 pixels (3x3) at a time with the primary focus on the centermost bit of the array. LBP is elaborated in the next step (Zhang, G et al., 2004).

3.5 Face Recognition and Face identification

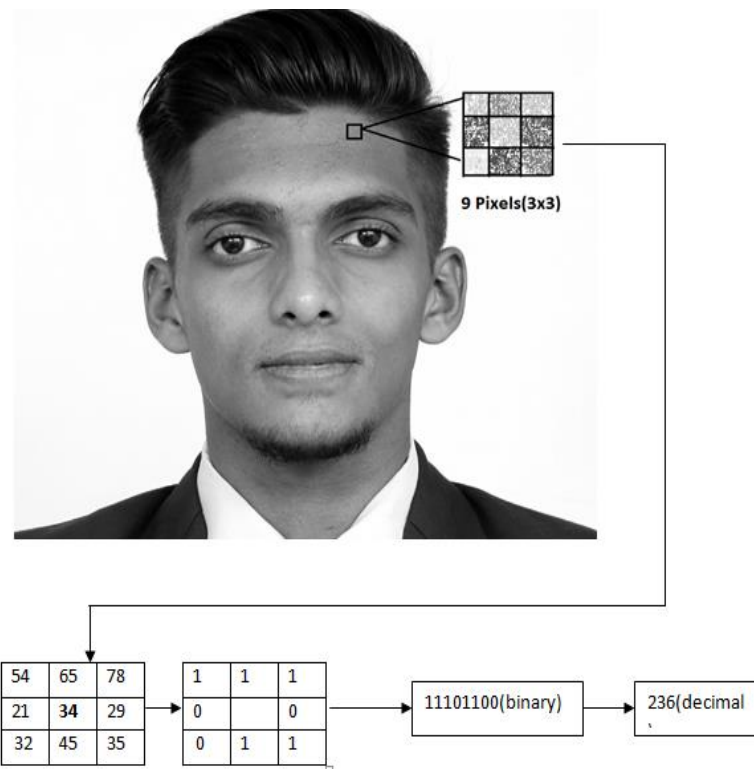
The first step is to train the face recognizer and make it familiar with all operational faces. This step includes to assign the names to each set of pre-saved photographs. This can be done with the help of labels. In this paper, An array of 'face-names' is created with a blank initial value and the actual array is starting from the 2nd element (array[1]) in order to match the names from the array element with the image folders starting from label '1'..(Ramiz Raja, 2017).

OpenCV has mainly 3 built-in face recognizers; Eigenfaces, Fisher Faces and Local Binary patterns Histograms (LBPH). In our paper, We are going to implement face recognition with the help of LBPH. This built-in recognizer introduces precision with the data and lucidity in identifying the correct face. After the face recognition step, an attendance is automatically marked in the form of binary number in the array of students maintained separately..(Ramiz Raja, 2017).

3.5.1 Local Binary Patterns Histogram (LBPH)

A 3x3 (9 pixels) window is selected for a particular region in an image and is moved across that image. Every part of the image generates a decimal value of its own. After processing, the image produces many decimal values which are local to every particular area of the image. A histogram is constructed using these decimal values. In the end, one image will generate only a single histogram which will be specific to a single face and photograph. Later

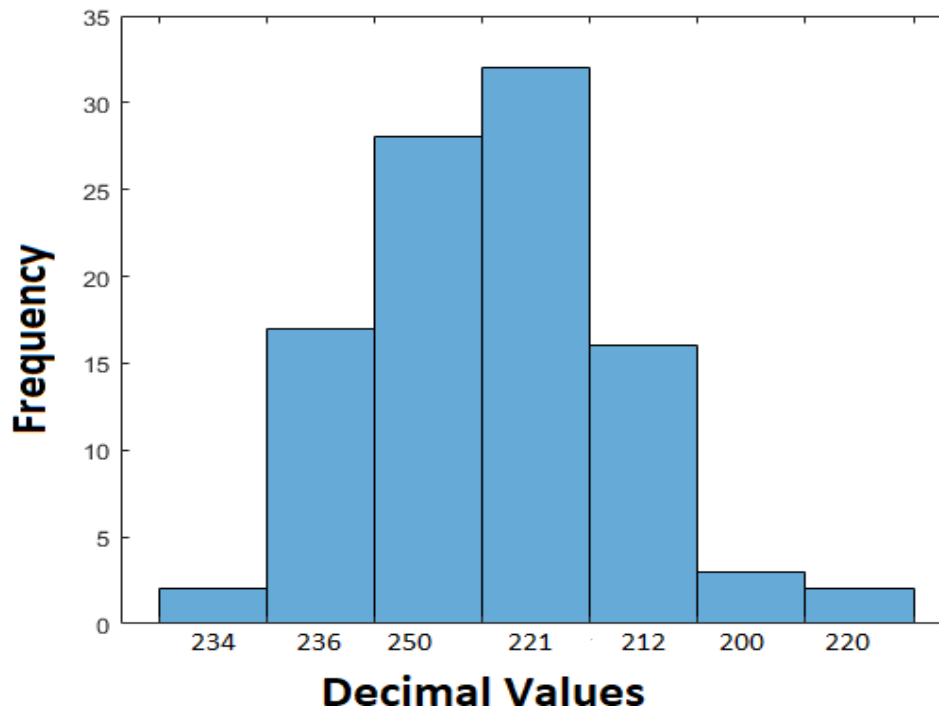
when a new image is fed to the program, it will create the histogram of the new image and compare it with the histograms it already has. Finally, it will return the label and name associated with the best match. (Ahonen et al., 2006).



The above figure displays the pixels local to a particular area in an image. A face is comprised of a large number of pixels, These pixels are grouped into a group of 9 pixels each. Each of the group of nine pixels is converted into one single decimal number as following; Every pixel above is converted into a binary digit with respect to the centermost pixel (34 in this case). Denote the neighbor pixel with 1 if it is greater than the center pixel and denote the neighbor pixel with 0 if it is less than the center pixel (Kelvin Salton Do Prado,2017).

In the above example, 9 pixels which are local to a certain part of the image are converted to a binary number(11101100). These binary numbers are grouped in the clockwise direction starting from the leftmost neighbor of the center digit to form a single binary number. The binary number is converted to a single decimal number which represents that part of the image. Like this, Every part of the image generates a decimal value. A single face generates a huge number of decimal value for every 9 pixels processes through the algorithm. The collective decimal values of the image are grouped together and contribute to forming a single *histogram* for an image or a face. (Patil, A., & Shukla, M. 2014).

A histogram is used to plot the accurate representation of numerical data. An example of the histogram:



3.5.2 Marking the binary Attendance

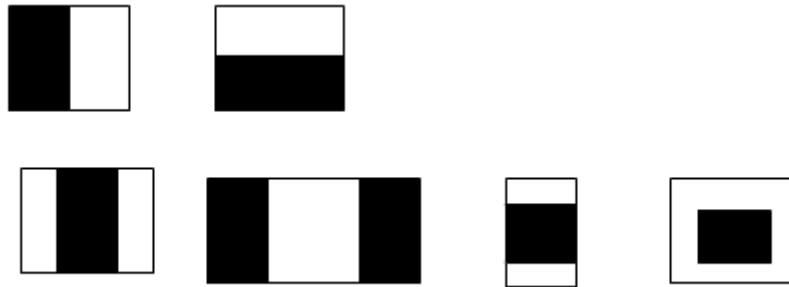
An array is separately maintained for filling in the binary values for attendance. As soon as the recognizer identifies the person, it marks present in the form of 1 against the specific student label in the array. When there is no face left for recognition, the array elements with the value of '-99' in the array should be replaced with zero.

4. Gaps and proposed methodology

The advantages of using LBP classifier outweigh the advantages of using Haar classifier for face detection. Haar classifier has a lesser number of advantages compared to LBP classifier. Our aim in this paper is to focus mainly on the speed and accuracy rate. Although LBP classifier has low detection accuracy as compared to Haar classifier, the computational process of implementation time of LBP is far more simple and fast compared to that of Haar classifier. Despite having high accuracy rate in face detection, Haar classifier has a limitation in processing face in poor lighting conditions. Haar classifier is less accurate for the darker faces whereas LBP has no such issue. It takes more time to train Haar classifier and on the other hand, LBP takes less time to train (Kadir, K et al., 2014).

Haar classifier works on the technique of extracting ‘haar’ features from the image by using windows. Each window is placed on the picture to calculate a single feature. A single value is calculated which is equivalent to the ‘sum of features under the black part’ minus the ‘sum of features under the white part’. These windows mostly focus on black parts of the face. This makes some features irrelevant. (Padilla et al., 2012).

Windows Used in Haar:



Furthermore, EigenFace is one of the widely used recognizers for the application of face recognition. According to this algorithm; not all the parts of the face are equally important, just the distinct features like eyes, nose, forehead, and cheeks. These important features are called *Principal Components*. The main drawback of this algorithm is that it considers illumination as an important component in order to successfully implement face recognition. Getting the right illumination can be somewhat challenging and cannot be feasible all the times which may lead to inaccuracy in data gathering. Too much illumination or too low illumination may result in the recognizer not picking up the principal components. This could lead to false recognition (Turk, M. A., & Pentland, A. P, 1991).

In order to overcome this problem, our paper implements the algorithm of Local Binary Pattern Histograms. LBPH is used to overcome illumination issues and provides precision and accuracy. Apart from that, LBPH is one of the easiest algorithms so far. It is easy to represent all the local features of a face through LBPH.

Conclusion

We have presented an approach for face operations which consumes less time, is simple and overcomes the problem of the poor lighting condition. This paper introduces an algorithm which overcomes various problems which may be encountered in traditional attendance systems.

The first problem is the time at which the student may enter the class. In this paper, the system is capable of classifying the presence of each student under a suitable time frame and a specific lecture. This feature promotes the accuracy of the system making it suitable for classrooms.

The second contribution solves the problem of two different students entering the classroom at a different time but in the lecture. The system runs a timer after the entry of the last student is done. If the timer runs out then the system is reinitialized for the student entering the class later.

We are working to extend the capabilities of this system, a system developed in today's time should have the ability to adapt and evolve. The facial recognition marks the upcoming age of automation, we have proposed a system and methodology for recognizing candidates inside a classroom. The technique used in this system may be very beneficial to other daily applications. There is a vast amount of applications of this technology that lies inside the classroom; this technology not only enables the user to mark efficient attendance but perform a real-time survey on students. Using this technology one can determine punctuality, sincerity and monitor class performance of a particular student if required.

References

- [1]. Ramiz Raja. (2017). Face Detection using OpenCV and Python. Source:<https://www.superdatascience.com/opencv-face-detection>. Accessed on November 12, 2018
- [2]. Ramiz Raja. (2017). Face Recognition using OpenCV and Python. Source: <https://www.superdatascience.com/opencv-face-recognition/> . Accessed on November 12, 2018
- [3]. Kar, N., Debbarma, M. K., Saha, A., & Pal, D. R. (2012). Study of implementing automated attendance system using face recognition technique. *International Journal of computer and communication engineering*, 1(2), 100.
- [4]. Balcoh, N. K., Yousaf, M. H., Ahmad, W., & Baig, M. I. (2012). Algorithm for efficient attendance management: Face recognition based approach. *International Journal of Computer Science Issues (IJCSI)*, 9(4), 146.
- [5]. Chintalapati, S., & Raghunadh, M. V. (2013, December). Automated attendance management system based on face recognition algorithms. In *Computational Intelligence and Computing Research (ICCIC), 2013 IEEE International Conference on* (pp. 1-5). IEEE.
- [6]. Patil, A., & Shukla, M. (2014). Implementation of classroom attendance system based on face recognition in class. *International Journal of Advances in Engineering & Technology*, 7(3), 974.

- [7]. Zhang, G., Huang, X., Li, S. Z., Wang, Y., & Wu, X. (2004). Boosting local binary pattern (LBP)-based face recognition. In *Advances in biometric person authentication* (pp. 179-186). Springer, Berlin, Heidelberg.
- [8]. Kelvin Salton Do Prado. (2017). Face Recognition understanding LBPH algorithm. Source: <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b>. Accessed on November 13, 2018
- [9]. Padilla, R., Costa Filho, C. F. F., & Costa, M. G. F. (2012). Evaluation of haar cascade classifiers designed for face detection. *World Academy of Science, Engineering and Technology*, 64, 362-365.
- [10]. Bradski, G., &Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc."
- [11]. Turk, M. A., &Pentland, A. P. (1991, June). Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on* (pp. 586-591). IEEE.
- [12]. Ahonen, T., Hadid, A., &Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (12), 2037-2041.
- [13]. Mike. (2010). Getting photo Metadata(EXIF) using python. Source:<https://www.blog.pythonlibrary.org/2010/03/28/getting-photo-metadata-exif-using-python/>. Accessed on November 12, 2018
- [14]. Walt, S. V. D., Colbert, S. C., &Varoquaux, G. (2011). The NumPy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2), 22-30.
- [15]. Jha, A. (2007). Classroom attendance system using facial recognition system. *The International Journal of Mathematics, Science, Technology and Management*, 2(3).
- [16]. Kadir, K., Kamaruddin, M. K., Nasir, H., Safie, S. I., &Bakti, Z. A. K. (2014, August). A comparative study between LBP and Haar-like features for Face Detection using OpenCV. In *Engineering Technology and Technopreneuship (ICE2T), 2014 4th International Conference on* (pp. 335-339). IEEE.