Class Attendance Management System Using Face Recognition

Omar Abdul Rhman Salim
Department of Electrical and Computer Engineering, Faculty of Engineering
International Islamic University Malaysia, Kuala Lumpur, Malaysia
o.salem92@gmail.com

Rashidah Funke Olanrewaju
Department of Electrical and Computer Engineering, Faculty of Engineering
International Islamic University Malaysia, Kuala Lumpur, Malaysia
frashidah@iium.edu.my

Wasiu Adebayo Balogun
Department of Mechatronics, Lagos State Polytechnic
Ikorodu, Nigeria
balogunwa1999@yahoo.com

***Abstract*—We are living in a world where everything is automated and linked online. The internet of things, image processing, and machine learning are evolving day by day. Many systems have been completely changed due to this evolve to achieve more accurate results. The attendance system is a typical example of this transition, starting from the traditional signature on a paper sheet to face recognition. This paper proposes a method of developing a comprehensive embedded class attendance system using facial recognition with controlling the door access. The system is based on Raspberry Pi that runs Raspbian (Linux) Operating System installed on micro SD card. The Raspberry Pi Camera, as well as a 5-inch screen, are connected to the Raspberry Pi. By facing the camera, the camera will capture the image then pass it to the Raspberry Pi which is programmed to handle the face recognition by implementing the Local Binary Patterns algorithm LBPs. If the student's input image matches with the trained dataset image the prototype door will open using Servo Motor, then the attendance results will be stored in the MySQL database. The database is connected to Attendance Management System (AMS) web server, which makes the attendance results reachable to any online connected web browser. The system has 95% accuracy with the dataset of 11 person images.**

Keywords—attendance system, LBPs, face recognition algorithm, Raspberry Pi

I. Introduction

Most of the universities around the world apply the attendance system to capture students' punctuality. However, the current paper/manual attendance system has many challenges. Passing an attendance sheet from one student to the other to sign takes time as well as causes distraction. Due to such problem, some lecturers delay the attendance till the end of the class, yet some students might be in a hurry to leave the class immediately, hence they might miss signing the attendance sheet. Furthermore, there are some students who never come to the class but sign attendance by proxy. In some cases, lecturers call by names one by one to mark the attendance but this method also consume lots of time. Yet another problem is some students come to class late especially the morning classes. Moreover, opening, and closing of the door during the lecture could be very disturbing for both lecturer and students. Hence, the manual attendance system, are not guaranteed and reliable.

There are two types of face recognition, Two-Dimensional face recognition, this type has some challenges which include variations in illumination, pose facial expression, great occlusion and image resolution [1]. The second type is Three-dimensional face recognition which recently demonstrated their superiority especially in illumination and poses variations. However, these approaches are not directly deployable due to the difficulty and cost of deploying 3D acquisition devices [1]. For this paper, an improved Two-Dimensional face recognition is used. The paper is organized in seven sections. Section II provides with the review of prior works followed by discussing the methodology employed in Section III. Section IV highlights the results and discusses the limitations in Section V. The paper is concluded in Section VI followed by future possibilities mentioned in Section VII.

II. Literature Review

Authors in [2] proposed a method to automate the attendance system by integrating the face recognition technology using Eigen face database and Principal Component Analysis (PCA) algorithm with MATLAB GUI. The architecture of the system first, captures the student image, pre-process it, applied Eigenface generated database then test the captured face image with Eigenface image. When the similarity distance test scored more than the threshold value of 0.3 then the face was not recognized finally attendance marking, was stored in a Microsoft Excel sheet integrated with the MATLAB GUI. The original face database consists of images for 15 persons each has 10 images with different position and direction.

In this paper [3], the authors developed and implemented a classroom attendance system using radio frequency identification (RFID) and face verification techniques. The system recognizes students by using the RFID card and for more confirmation of the student's identity, face recognition technique has been added using Fast Adaptive Neural Network Classifier (FANNC). The classifier was trained and tested to identify human face images. Every student needs to take seven dissimilar head poses images in order for the classifier to identify students' images. The facial system tested on six distinct images of students and it achieved accuracy up to 98% for the front face.

In [4] the authors focused on changing the traditional manual attendance to a digitized system using facial recognition. For face recognition module, the system used MATLAB software to implement the Principal Component Analysis (PCA) algorithm. The code was loaded on an embedded hardware system using Microcontroller PIC which also connected to a servo motor to open the door once the facial authentication was successful. From the experiment, the result was found that the system was so sensitive when there was a change in the background and with different head orientation.

978-1-5386-6992-1/18/$31.00 ©2018 IEEE

Furthermore, authors in [5] proposed a method for class attendance system using face recognition technology. The system was implemented by combing two algorithms Discrete Wavelet transform (DWT) and Discrete Cosine Transform (DCT) to extract the features of student's face. The Discrete Wavelet transform DWT is well known to be a highly flexible and efficient method when it comes to decomposition of a signal. Thus, it decomposes the image into its wavelet coefficients and scaling function. Discrete Cosine Transform DCT take cares of de-correlation, energy compaction of the dedicated image. Facial training image was the set of trained student images which were used to check the identity of the input student's image. The experiments have shown 82% of successfully recognized input, 121 out of 148 successful faces recognition which conducted by involving 16 students.

III. METHODOLOGY

A. *Introduction*

The system not only recognizes students' faces but also controls the door to whether the student is allowed to access or not. Moreover, the system was built as an online Web Server, so the attendance results can be accessible to any authenticated web client. A user interface for the website server is included so the lecturer can check the attendance results from any computer browser connected to the Internet. The facial recognition is done by implementing Local Binary Patterns (LBP).

By facing the camera, the face image is captured. The first processing step is to detect and crop the region of interest ROI which is the human face. It can be done by applying the Haar Feature-based Cascade algorithm. After that, the image features are extracted using LBPs, then LBPs algorithm compares the extracted features with the trained datasets. Based on the results the prototype door will open for the recognized student within the time frame by using a servo motor. Finally, by clicking letter 'c' as in capture on the keyboard system, the attendance results are stored in MySQL database, so it can be accessible to the web server. The architecture of the system is shown in Fig. 1.

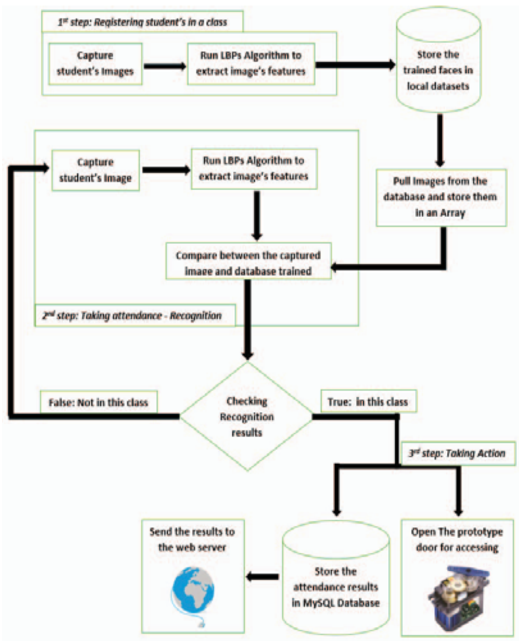


Fig. 1. System architecture

To train the dataset, the system requires at least 12 images for each student in a different facial expression. In this system, 11 students were used as sample test. The total images were 11 * 21 = 231 total face images. For face recognition, Local Binary Patterns LBPs algorithm is used, since it can update the classifier model with new face images in a dynamic way, unlike PCA algorithm which needs to rebuild both the Mean and Eigenfaces every time a new student is added to a class. The number of images might be more than 21 if the algorithm finds it hard to recognize a specific student.

B. *Dataset and LBPs Algorithm*

The first step to implement LBPs algorithm is by dividing the face image into 7 x 7 equal sized cells, as shown in Fig. 2.

Fig. 2. (7 x 7) equally sized cells of the face image [*Adopted from* [6]].

The LBPs operator focuses on how to able each pixel with its weighted decimal value. Since every centered pixel has 8 pixels-3x3 surrounding neighbors, it can be represented as a binary number. If a neighbor is greater or equal than the central pixel then a logic '1' bit is replaced for that pixel, else logic '0' bit is replaced for that pixel. Then, the 8 bits binary number is converted to decimal which represents the threshold of that centered pixel, as shown in Fig. 3 [7].

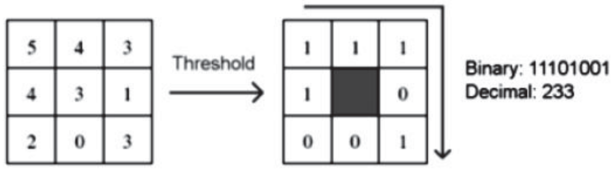


Fig. 3. Threshold of centered pixel 3 [*Adopted from* [7]]

Next, these thresholds are replaced in every cell. These values are turned into histograms based on the common features of these values. For example, looking at the statistics how many times the number 233 come up in the cell or how many times 234 come up in the cell. By computing a histogram for each of the cells, the algorithm is able to encode a level of spatial information such as the eyes, mouth, nose, etc. In other words, the encoding of histograms for each of the cells gives different results based on the weight of each threshold of pixels.

Fig. 4(a) [*Adopted from* [6]]

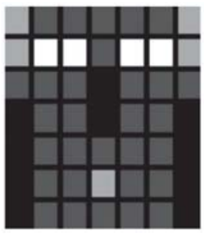


Fig. 4(b) [*Adopted from* [6]]

Fig. 4(a) illustrates the divided face image while Fig. 4(b) illustrates the weighting schemes after encoding the histogram for each of the cells. As it can be seen the white cells represent the eyes are weighed 4x more than other cells because the eyes are brighter than other parts of the face. This means that the white cell regions are multiplied by 4. Meanwhile, Light gray cells like (mouth and ears) contribute 2x more which means the LBP histograms for these cells are multiplied by 2x. For inner cheek and forehead are Dark gray cells that they only contribute 1x. Finally, the outer cheeks and nose are represented by black cells which means they are totally neglected and weighted 0x [6]. These LBP histograms are concatenated together by finding common features to form the final feature vector, as shown in Fig. 5.

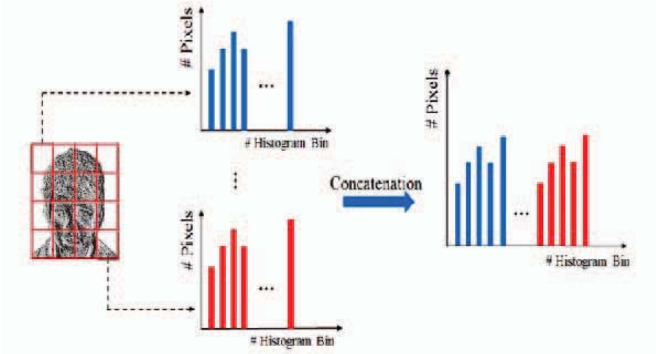


Fig. 5. Concatenation of LBP histograms [*Adopted from* [7]]

Once the final feature vector is found, face recognition can be performed by implementing k-nearest-neighbors KNN with (K=1). KNN is based on a straightforward nonparametric decision. Every input image is tested based on the distance of each of its facial features from the other images' features in the trained dataset [8]. The distance can be found by calculating X^2 distance using the Chi Square distance metric formula [7].

$$\text{distance} = \chi^2(S, M) = \sum_{i=1}^{K\times L}\left(\sum_{j=1}^{P(P-1)+3} \frac{(S_{i,j}-M_{i,j})^2}{S_{i,j}-M_{i,j}}\right) \quad \text{…… (1)}$$

From (1), S and M are the normalized histograms of K by L regions to be compared. S is the known feature vector, M is the test vector. X^2 is simply the distance between the S –the known face in the dataset and M – the unknown input face. X^2 is computed j times where j represents the number of students in the datasets, based on KNN algorithm, the lowest X^2 the most likely S and M faces are of the same object, as shown in Fig. 6.

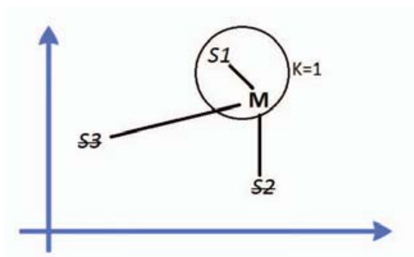


Fig. 6. KNN algorithm with K=1

In face recognition, always K = 1, because each S_j represents different student face and the interest here is to find the unique identity of M. In Fig. 6, the identity of M is equal to S1 because it's the smallest distance X^2. If K = 2 the identity of M would be for both S1 and S2 which does not make sense in face recognition implementation. Moreover, the lowest value of Distance X^2 is called The Confidence.

C. *Attendance Management System website (AMS)*

Once the student's face is recognised, the system directly send the result to MySQL database making the results reachable to the Attendance Management System website. Basically, both recognition system and Attendance Management System is connected to same MySQL database as shown in Fig. 7.

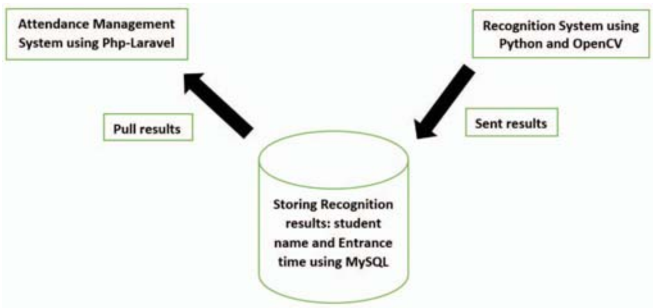


Fig. 7. Connection between the recognition and attendance management systems

For building the (AMS) website, Laravel, Php framework was chosen for its simplicity with connecting to the database and displaying the results. Since the Raspberry Pi 3 can be connected to the internet, the results are no longer local and can be reachable to any web clients' browsers. In this case, the Raspberry Pi 3 acts as a server as shown in Fig. 8.

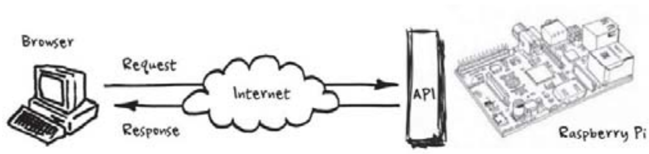


Fig. 8. Connection between web server (Raspberry Pi) and web client (Browser)

D. *The Procedure of System Implementation*

```
Start Program.
Face the camera and press 'c' on the keyboard.
Capture the student's image.
Detect the region of interest using Haar Feature-based algorithm
Crop the ROI of the image to become 130x150.
Convert the cropped image to grayscale.
While ( It is a class time ):
    Apply LBPs algorithm for face recognition.
    IF (the input image matches the trained dataset):
        Open the prototype door.
        Store the results in MySQL Database.
        Display "Recognition Succeed!" on the screen.
        Displaying the result attendance on (AMS) website.
    ELSE
        Display "Recognition Failed!" on the screen.
    End IF.
End while.
IF ( It is not a class time ):
    Display "It's not class time!" on the screen.
End IF.
End Program.
```

IV. Results and Discussion

For face recognition implementation, three programs were developed using Python. The first one to gather selfies by capturing frames of student's face from a video record, while the second one is to train these selfies and store them in a classifier which later on it is used to recognize the students' face. The final program was used to recognize the input face, which as explained before taking the input face and comparing it with the trained data using the classifier. Gathering selfies was made by taking a video record while taking this video the face 'Region of Interest' was deducted, cropped, encoded then stored in a text file as strings, each string line represents a face image of the student. Fig. 9 shows the decoded images of a student.

Fig. 9. Decoded trained face images of a student

A. Recognition Results

TABLE I. Results of Different Students' Recognition

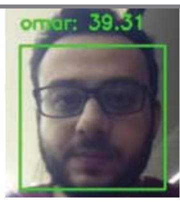
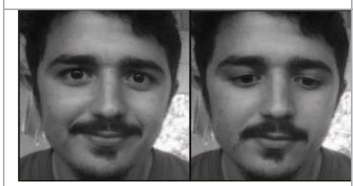
Trained Faces	Tested Face
	
	
	

Table I summarizes the results of three students' recognition attendance. Though each student has more than 21 trained images, only two of them are shown in the first column of the table. The second column shows the recognized face of the student while testing the system. Moreover, failed recognition happens if a student is not registered in the said class. He/She would not be recognized, and the system displays Unknown person as shown in Fig. 10 (a, b and c).

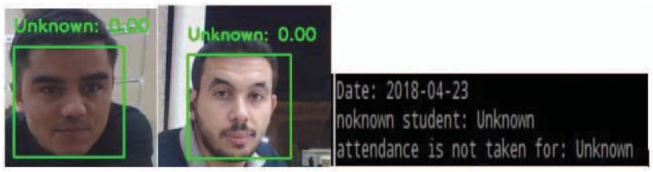


Fig. 10. (a, b) two unknown students, (c) attendance was not taken.

B. Algorithm Analysis

To analyze LBPs Algorithm and find out how efficient it is, the first thing to do is computing the Confusion Matrix on the tested dataset. Which can be done with the help of Sklearn library by using the pre-defined function confusion_matrix(). The eleven students' names were replaced to "S1 -> S11" due to the small space in the plot, as shown in Fig. 11.

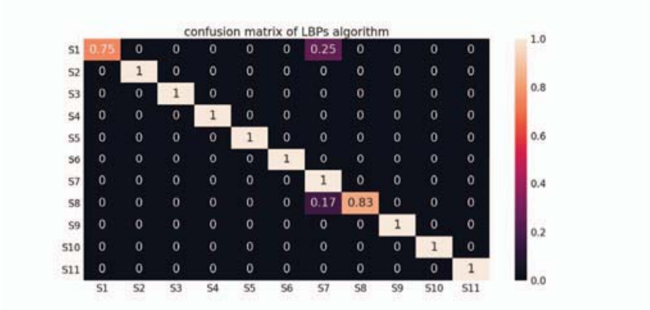
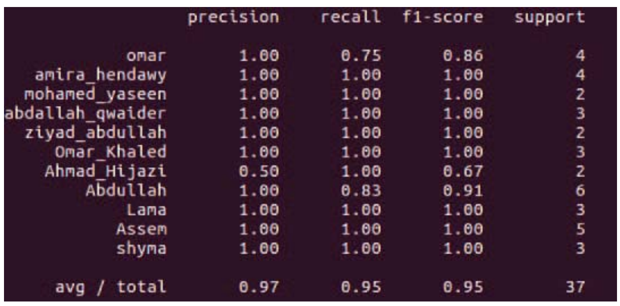


Fig. 11. Confusion Matrix of LBPs algorithm

In Fig. 11, it can be seen that S2 which indicates the student 'Amira Hindawy' has been recognized correctly with 100 % percent accuracy represented by '1'. The same accuracy goes with most of the students except S1 and S8 which are 'Omar' and 'Abdullah' respectively. Both of them had confusion with S7 'Ahmad Hijazi' and the accuracy is 75 % for S1 while for S8 is 83 %. Next is finding the overall accuracy of the algorithm based on the tested database. For this task, Sklearn library was also used that gives an accurate classification report using the built-in function classification_report(). There are types of accuracy management but for this case, F1-Score is considered with 95 % accuracy because the data test for each student was not equal as shown in the Support column. For example, student 'omar' had 4 test samples while 'mohamed_yassen' had 2 test samples, as shown in Fig. 12.



	precision	recall	f1-score	support
omar	1.00	0.75	0.86	4
amira_hendawy	1.00	1.00	1.00	4
mohamed_yaseen	1.00	1.00	1.00	2
abdallah_qwaider	1.00	1.00	1.00	3
ziyad_abdullah	1.00	1.00	1.00	2
Omar_Khaled	1.00	1.00	1.00	3
Ahmad_Hijazi	0.50	1.00	0.67	2
Abdullah	1.00	0.83	0.91	6
Lama	1.00	1.00	1.00	3
Assem	1.00	1.00	1.00	5
shyma	1.00	1.00	1.00	3
avg / total	0.97	0.95	0.95	37

Fig. 12. Classification report of LBPs algorithm

The f1-score accuracy can be interpreted as a weighted average of the precision and recall as illustrated in Fig. 13.

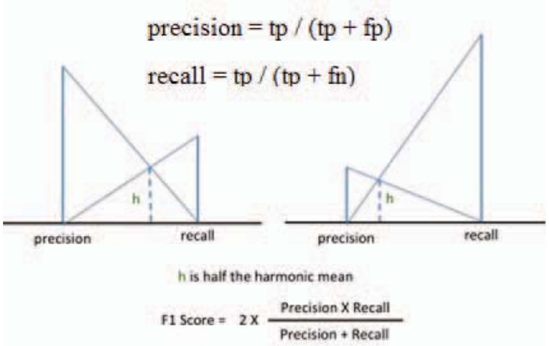


Fig. 13. F1-Score Formula

V. System Limitations

The system is uses Local Binary Patterns algorithm to recognize faces, there are some limitations here. LBPs algorithm has light sensitivity. Since it deals with the value of each pixel in the original image, these pixels change their values with different lighting. The pixel value increases when the light is on while it decreases when the light is off. This rapid change leads to an increment in the Distance X2/Confidence between the trained and input data, making it very difficult to recognize as shown in Fig. 14 (a, b and c).

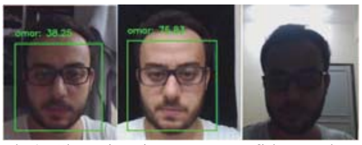
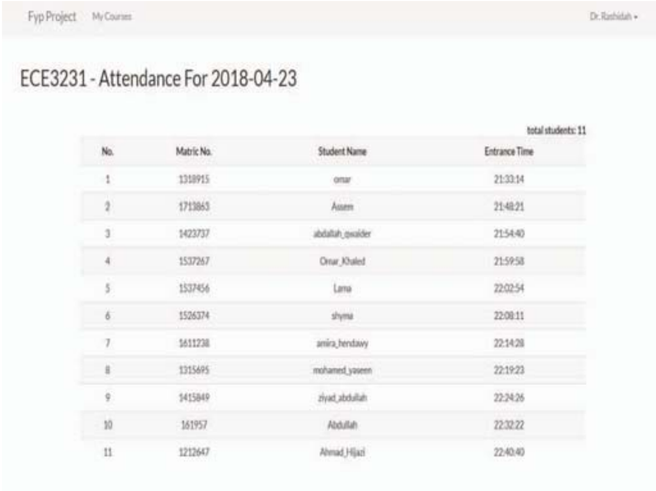


Fig. 14. (a, b and c) Changing in LBPs confidence due to the change of lighting

Fig. 14(a) shows a good confidence distance between the trained and the input face with 38.25 since it was recognized with the same spotting light. in Fig. 14(b) the confidence of input face is far from the trained face with 75.83 because the object changed his place to a darker background and the light was facing his face. The final one did not detect the face due to a very luminous background. This problem can be avoided easily, due to the fact that the system designed to be fixed on the classroom door.

A. Attendance Results

For displaying the attendance results the Attendance Management System (AMS) website was designed to update the attendance sheet every time a student comes to the class. The website is provided with an authentication system, every lecturer can log-in and check his/her own courses attendance sheet from his/her personal computer during the class time to ensure the results are accurate thus no student can bring a photo of his/her friend to take the attendance for him/her as shown in Fig. 15.


Fyp Project My Courses Dr.Rashidah

ECE3231 - Attendance For 2018-04-23

total students: 11

No.	Matric No.	Student Name	Entrance Time
1	1318915	omar	21:33:14
2	1713863	Assem	21:48:21
3	1423737	abdallah_qwaider	21:54:40
4	1537267	Omar_Khaled	21:59:58
5	1537456	Lama	22:02:54
6	1526374	shyma	22:08:11
7	1611238	amira_hendawy	22:14:28
8	1315695	mohamed_yaseen	22:19:23
9	1415849	ziyad_abdullah	22:24:26
10	161957	Abdullah	22:32:22
11	1212647	Ahmad_Hijazi	22:40:40

Fig. 15. Displaying attendance for all students

VI. Conclusion

The aim of this research paper is to develop an automated attendance system to be used in educational institutions, which can produce more accurate results than the manual attendance sheet. The system is based on Raspberry Pi as the hardware. The system is programmed using both Python for face recognition system and PHP for attendance management system website. Moreover, it is provided with a prototype door using Servo motor which would open for the recognized student to pass every time the recognition is successful. The attendance is stored in MySQL Database and with internet connection provided, the results are accessed from any computer web browser. Each lecturer required to log-in to the AMS website to access his/her attendance sheets. Raspberry Pi is chosen for its small size and affordable price. The small and light size is important because the whole product with the camera is supposed to be fixed on a classroom electric door.

VII. Future Improvements

For the future work, there are some promises to improve and enhance the performance of the system. First, replacing RASPBERRY PI 3 MODEL B with ODROID-XU4 Portable Computer. This Computer has Cortex-A7 Octa core CPUs and 2Gbyte LPDDR 3 RAM which is more advanced than Raspberry Pi 3 hardware. Since the face recognition algorithms are heavy, the performance can be improved rapidly using ODROID-XU4. Second, enhancing the Attendance Management System website by generating automated warning letters for the students who do not attend their classes.

Acknowledgment

This work was partially supported by Ministry of Higher Education Malaysia (Kementerian Pendidikan Tinggi) under Research Initiative Grant Scheme (RIGS) number RIGS-17-048-0623.

References

[1] X. Zhao, W. Zhang, G. Evangelopoulos, D. Huang, S. Shah, Y. Wang, I. Kakadiaris and L. Chen, "Benchmarking Asymmetric 3D-2D Face Recognition Systems." *Pattern Analysis and Machine Intelligence,* pp. 218–233, 2013.

[2] E. Rekha and P. Ramaprasad, “An efficient automated attendance management system based on Eigen Face recognition.” *7th International Conference on Cloud Computing, Data Science & Engineering - Confluence*, pp. 605-608, 2017.

[3] K. H. Pun, Y. S. Moon, C. C. Tsang, C. T. Chow, S. M. Chan, “A face recognition embedded system.” *Biometric Technology for Human Identification,* vol. 5779, no. 2, p. 390, 2005.

[4] J. Joseph and K. P. Zacharia, “Automatic Attendance Management System Using Face Recognition.” *International Journal of Science and Research (IJSR), ISSN (Online),* pp. 2319-7064, 2013.

[5] S. Lukas, A. Mitra, R. Desanti and D. Krisnadi, “Student attendance system in classroom using face recognition technique.” *International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1032-1035, 2016.

[6] V. Mohanraj, V. Vaidehi, S. Vasuhi and R. Kumar, “A Novel Approach for Face Recognition under Varying Illumination Conditions.” *International Journal of Intelligent Information Technologies*, pp. 218–233, 2018.

[7] J. Khorsheed and K. Yurtkan, “Analysis of Local Binary Patterns for face recognition under varying facial expressions.” *24th Signal Processing and Communication Application Conference (SIU)*, pp. 2085-2088, 2016.

[8] H. Ebrahimpour and A. Kouzani, “Face Recognition Using Bagging KNN.” 2018.