



IoT-Based Student Monitoring System for Smart School Applications

Hemdani Rahendra Herlianto¹, Gede Putra Kusuma²

¹Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480, hemdani.herlianto@binus.ac.id

²Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480, inegara@binus.edu

ABSTRACT

Internet-of-things (IoT) provides a framework for integrated computing devices with sensors, smartphones, and cloud computing platforms for interaction between devices. Smart School is one of the concepts of IoT and is also a solution for education for student monitoring problems. In this study, student monitoring is proposed using an IP-based CCTV system with the ability to detect and recognize people. With the support of the use of beacon chips as student identity, the system can simplify problems of face identification to face verification. The use of fingerprinting techniques is a general technique used to determine the position of objects when using beacon chips based on transmitter signals so that the position of a student's face can be determined based on the user's position. The face verification process is then carried out by combining fingerprinting techniques and facial recognition captured by CCTV cameras with a face comparison method of OpenFace. Experimental results show that the proposed system provides good accuracy on student monitoring for smart school applications.

Key words: Internet-of-Things, Fingerprinting Technique, User Localization, Face Detection, Face Verification.

1. INTRODUCTION

Internet-of-things (IoT) provides a framework for integrated computing devices with sensors, smartphones, and cloud computing platforms for interaction between devices [1]. In an integrated smart multi-agent system the technology can facilitate the geriatric control of the residence [2]. Smart School is one of the concepts of IoT and the solution for education for the problem has been explained previously. Smart School will make it easier for teachers, parents, and schools as users to monitor their students and provide information on their child's whereabouts in the school environment. With the smart school monitoring feature, IoT-based students are expected to be able to solve the problems previously explained.

Research on the implementation of IoT-based passenger monitoring for Smart School Applications [3] proposed the

implementation of IoT using cards with Bluetooth low energy (BLE) technology. Which aims to provide monitoring facilities in the school shuttle process. Students will be monitored from departure from home until arriving home. Based on the algorithm used, the signal strength test results obtained by -54 dB as a selection limiting value and can read valid data, then the duration of time used is 1 minute [4].

Based on previous research conducted on IoT-based passenger monitoring for Smart School Applications, researchers want to improve the authenticity of BLE card users because in previous studies that have been explained there are still problems regarding the misuse of BLE card users that might be used by others and also problems regarding the existence students when carrying BLE cards, which are used to verify their existence.

The main purpose of this system is camera-based monitoring using face recognition methods that can prove the identity of a BLE cardholder. By combining the BLE-based conversion process and the camera, it can simplify the problem in face recognition from problem identification to verification problems. One-to-many identification problems require a lower level of accuracy than one-to-one verification problems. Using a CCTV camera using calibration techniques on the camera and the use of low energy Bluetooth can solve the problem.

Further analysis and comparison with other works will be explained briefly in the following section. Section 2 describes other research on smart school applications using BLE and face detection using CCTV. The description of the dataset, the methods used, the learning model, and the system framework are detailed in Section 3. Section 4 explains our experimental settings and the results of the analysis. Finally, limitations, conclusions, and future scopes are presented in Section 5.

2. LITERATUR REVIEW

Bluetooth low energy (BLE) research was also conducted on the IoT-based Smart Museum using Bluetooth low energy [5]. In his research, a museum admin uploaded all the images of smart objects present in the museum hall and information

content in a multimedia format. When entering the museum everything is provided with one wearable device which will be synchronized with mobile users. IoT-based smart museums use a Raspberry Pi device that will capture user movements and use a background reduction algorithm to process images and get localization information from Bluetooth low energy (BLE) while in the museum. This wearable device will improve the performance of the entire system by sending only frames that are suitable for central cloud processing. So when visitors are in the position of cloud artwork will provide the data requested by visitors.

Camera calibration methods have also been carried out with 6 intrinsic parameters to characterize light field cameras based on traditional two-parallel-plane (TPP) representations. The light camera can capture spatial information and beam angle, enabling 3D reconstruction with a single exposure. The 3D reconstruction geometry is significantly affected by the intrinsic parameters of the light field camera. The results of the study explained that the camera calibration method produces robust verification especially in the initialization and optimization of the distortion improvement.

According to [6], regarding the extensive research of the OpenFace toolkit states that his research can make facial landmarks more accurate for face detection, head pose estimation, recognition of facial action units, and eye gaze estimation. The computer algorithm that represents the core of OpenFace 2.0 shows state-of-the-art generating all the tasks mentioned. Furthermore, OpenFace 2.0 can perform real-time performance and can run from a simple webcam without special hardware [7].

3. RESEARCH METHOD

3.1 System Architecture and Workflow

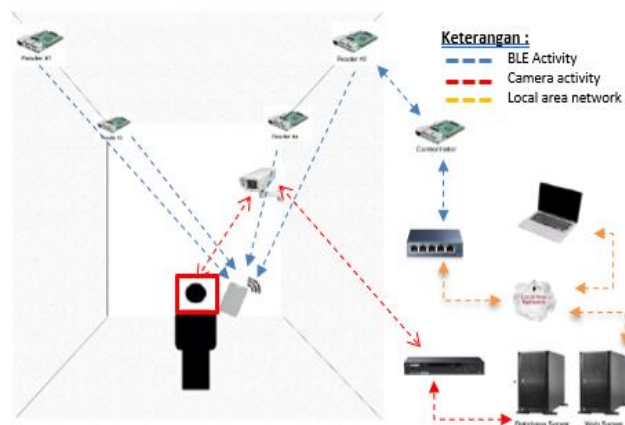


Figure 1: System architecture and workflow

The architecture and workflow of the proposed system are shown in Figure 1. In the first stage, students enter the monitoring room carrying beacon chips with Bluetooth low energy technology. When entering the monitoring area, the signal given by the beacon will be detected by the presence

of a signal receiver that has been installed 4 points. The coordinate value (x, y) generated by the signal will be calculated by fingerprinting techniques to determine the position of students. Next, the second stage will simultaneously take into account the position of the student's face with the calibrated camera to produce a 3D model that describes the location of the student's face. Then it will be extracted in 2D with the convex-hull method to make it easier to find the position of the face. Viola-Jones algorithm is used to detect faces with calculations based on rectangle features that have been explained on the theoretical basis of face detection [8]. After getting the face position the third stage is verifying the beacon chip user by computing the server and database. The use of OpenFace 2.0 will extract the results of faces captured by the camera which will then be integrated with UUID that releases the beacon and the final result is expected to verify the beacon user and the detected face is appropriate. The system workflow can generally be divided into 2 phases, namely the preparation phase and the monitoring implementation phase.

A. The preparation phase

In the initial preparation phase to be carried out is the installation of the system based on the architecture that has been proposed. After the architecture is installed, the next step is to collect the radio map for the preparation of the indoor positioning system (IPS) method with the fingerprinting technique [9]. Radiomap that has been collected will be used to calibrate the camera. This calibration is needed to obtain intrinsic and extrinsic parameters from the camera needed to do projection based on pin-hole camera models [10].

B. The monitoring implementation phase

Monitoring will be carried out when the location of students can be predicted using the BLE-based indoor positioning system (IPS) method. From this position, the camera will predict the position of the head in the real world by creating virtual bounding boxes with a size large enough to ensure the position of the head is inside the bounding box. The position of the head that is already in the Bounding box is then projected onto the camera image based on the pin-hole camera model [11]. The results are produced by the projection in the form of a convex hull in the form of images. Face convex hull will produce face detection on the camera image. The face on the convex hull is thought to be the face of a student whose location has been previously selected. Then in the final stage face verification will be performed on the faces detected in the convex hull using the OpenFace method.

3.2 Techniques

3.2.1 Indoor Positioning System

Indoor Positioning System (IPS) is a method to find out the position of objects or people in a room by using radio waves. the received signal is obtained based on the waves emitted or sensors that are interconnected through mobile devices. IPS that uses radio waves has advantages compared to other sensors because of its use can capture signals around it.

Radio wave technology with IPS is commonly used on WLAN and Bluetooth networks. One of the uses of wave technology uses an Arduino microcontroller which can capture the signal strength of Bluetooth [12]. Various techniques and algorithms were adopted to estimate the user's position in wireless-based localization. The Fingerprinting technique is a technique generally used to determine the position of objects based on transmitter signals [13].

There are two phases for determining the location with the fingerprinting technique, as shown in Figure 2:

- a. *In the offline phase*, the location determination is done by calculating the RSSI value based on the grid points in the specified area. The distance between the two closest physical positions is the grid distance and is usually stated in meters or feet. Each grid point has a different RSSI list which is then compiled with enough statistics to create a database [14].
- b. *The online phase* is the phase of calculating the estimated location value using the position algorithm based on RSS values in real-time to the fingerprint database.

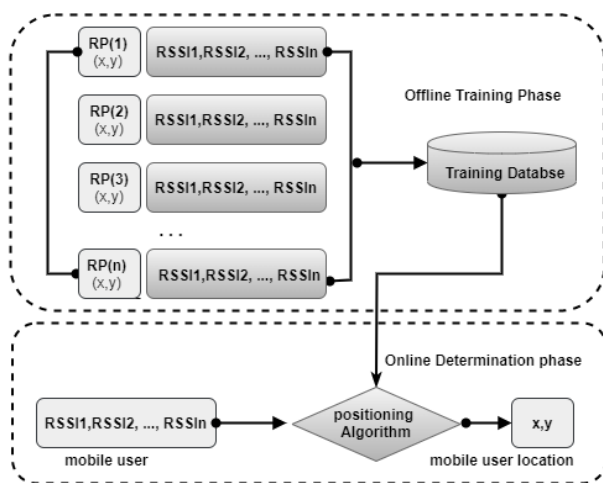


Figure 2: Two phases of the fingerprinting technique

3.2.2 Camera Calibration

One of the most widely used camera calibration techniques is the method proposed by Tsai. The implementation requires appropriate 3D point coordinates and 2D pixels in the image. In the calibration technique there are two stages for calculating coordinates: first, position and orientation, while in the second stage, to determine the camera's internal parameters [15]. Although relatively long, the proposed calibration method is suitable for use in large application areas because it can win coplanar and non-coplanar points which allows it to calibrate internal and external parameters separately [16].

Tsai's proposed method is based on a projection model with a pinhole perspective and eleven parameters for estimating objects [17]. The following parameters are used:

- f = Camera focus length,
- k = Coefficient of radial lens distortion,
- C_x, C_y = The coordinates of the radial lens distortion center,
- S_x = Scaling factor
- R_x, R_y, R_z = Rotation angle for transformation between models and camera coordinates,
- T_x, T_y, T_z = Translational component for transformation between models and camera coordinates.

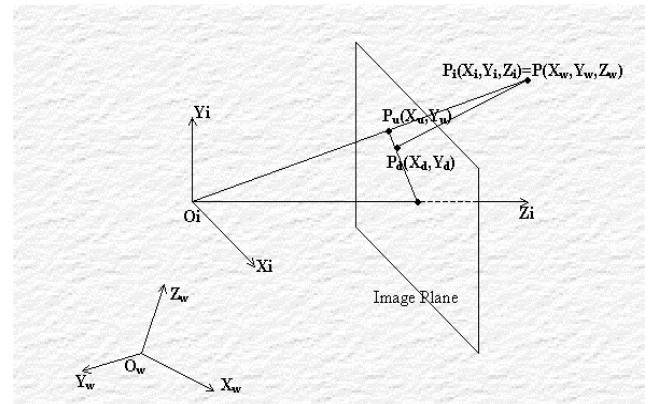


Figure 3: Projection model with Convex-hull perspective

The transformation from the model (X_w, Y_w, Z_w) to the image coordinates (X_i, Y_i, Z_i) with the camera extrinsic parameters (T translation and R rotation) in the equation:

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T \tag{1}$$

Where R and T are 3D transformations from the model to the camera coordinate system and are defined as follows:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{2}$$

with,

$$\begin{aligned} r_{11} &= \cos(R_y) \cos(R_z) \\ r_{21} &= \cos(R_z) \sin(R_x)(\sin R_y) - \cos(R_x) \sin(R_z) \\ r_{31} &= \sin(R_x) \sin(R_z) + \cos(R_x) \cos(R_z) \sin(R_y) \\ r_{41} &= \cos(R_y) \sin(R_z) \end{aligned}$$

$$r_5 = \sin(R_x) \sin(R_y) \sin(R_z) + \cos(R_x) \cos(R_z)$$

$$r_6 = \cos(R_x) \sin(R_y) \sin(R_z) - \cos(R_z) \sin(R_x)$$

$$r_7 = -\sin(R_y)$$

$$r_8 = \cos(R_y) \sin(R_x)$$

$$r_9 = \cos(R_x) \cos(R_y)$$

(R_x, R_y, R_z) = Euler's angle of rotation around three axes.

(T_x, T_y, T_z) = 3D translation parameters from the model to the image coordinates.

The transformation from the 3D position (in the coordinate frame of Figure 3) to the plane of the image is then calculated through the following steps (Figure 3 as a model for calculation):

Step 1:

Transform from the coordinates of the 3D model (X_i, Y_i) to the undistorted plane of the image (X_u, Y_u) using r3 r2 f-1 as a perspective projection with pinhole camera geometry as follows:

$$X_u = f \frac{X_i}{Z_i} \tag{3}$$

$$Y_u = f \frac{Y_i}{Z_i}$$

Step 2:

Transform from the coordinates of the undistorted image (X_u, Y_u) to the coordinates of the

distorted image (X_d, Y_d) as follows:

$$X_d = X_u(1 + kr^2)$$

$$Y_d = Y_u(1 + kr^2) \tag{4}$$

where,

$$r = \sqrt{X_d^2 + Y_d^2}$$

and k is the lens distortion coefficient.

Step 3:

The transformation from the distorted coordinates in the image plane (X_d, Y_d) to the final image

coordinates (X_f, Y_f) are:

$$X_f = \frac{X_d X_d}{d_x} + C_x$$

$$Y_f = \frac{Y_d}{d_y} + C_y \tag{5}$$

with,

(d_x, d_y) : the distance between adjacent

sensor elements in the X and Y directions. d_x and d_y are fixed parameters of the camera that only depend on the CCD size and image resolution, $(X_-(f), Y_-(f))$ is the position of the last pixel in the image.

According to research conducted by [18] regarding facial recognition systems with a high degree of accuracy. In the process of matching faces, to get a matching face with a variety of viewpoints carried out several stages starting from the received image, then enter the preparation period for the image to be detected, and then the image will be detected. After the image can be detected the image will be minimized in size to be able to enter the extraction process [19]. After the image is extracted, the image will do the matching on the database that has been provided.

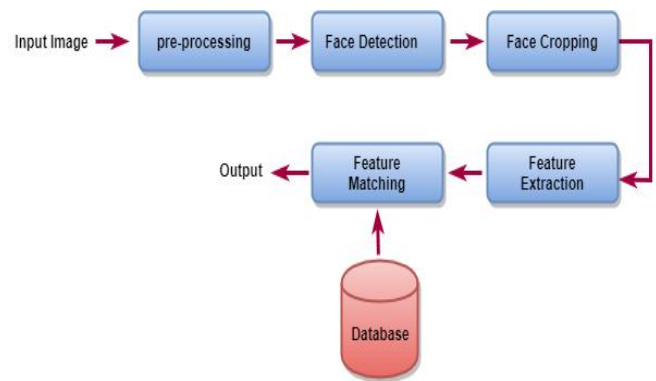


Figure 4: Face recognition framework

4. EXPERIMENTS

This section explains the results obtained in this study including the error value and accuracy of the face verification system.

4.1 Experimental Design

The research was carried out with preparations on CCTV cameras that produce 2D images derived from real-world images. camera transforms 3D media (world plane) into 2D (image plane). So camera calibration is needed to determine the intrinsic and extrinsic values used to project 3D coordinates into 2D. The first step that must be taken before doing camera calibration is taking some chessboard pattern pictures, in this study 25 sample images were taken with the chessboard pattern 8x6 size. Then perform the edge detection process on the chessboard pattern from the image that has been projected. By knowing the location of the points in the world coordinate and pixel coordinate, a matrix that combines the two can be found.

The experiment was carried out by testing the positioning accuracy of the system using the k-NN method from the RSSI fingerprint training data obtained at the offline data collection stage. Furthermore, manual calculations are carried out using a weighted addition algorithm to determine the position prediction. The test to determine the predictive position used is based on the smallest k average value. The results of the position prediction are used to determine the coordinates of the face position when students use the beacon chip.

Testing accuracy is also carried out in the face detection process on the camera. The face that is captured by CCTV cameras will be determined by the coordinate position of the face in the image using the Viola-Jones algorithm with a face recognition detection system. In the final stage, the test is carried out by calculating the accuracy value from the comparison of the predicted value of the face position when the student holds the beacon chip and the position of the student's face. the same that was caught on the CCTV camera.

The results of the test are carried out in the evaluation stage of the calibration results and evaluation of the results of the algorithm used. Evaluation of the camera calibration results is carried out to ensure that the intrinsic and extrinsic parameters are suitable for the implementation of the indoor positioning system and face verification. while the evaluation of the algorithms chosen in this study was also carried out by comparing the results of the accuracy of the algorithms used. Accuracy can be compared using face prediction results based on the closest position (face matching) with the prediction results of faces with OpenFace.

4.2 Experimental Results

4.2.1 Camera Calibration

The camera can produce 2D images derived from real-world images. camera transforms 3D media (world plane) into 2D (image plane). So, camera calibration is needed to determine the intrinsic and extrinsic values used to project 3D coordinates into 2D. The first step that must be taken before calibrating the camera is taking some pictures of chessboard patterns, in this study 25 sample images were taken with a

chessboard pattern size of 8x6. Then perform the edge detection process on the image chessboard pattern that has been projected. The projection results from 3D to 2D projection are compared with the calibration results with 2D coordinates on the camera. Table 1 shows the results of 3D to 2D projection with an average error of 0.00948.

Table 1: 3D to 2D projection results

Ground Truth of Chessboard 2D Coordinates		3D to 2D Projection Coordinates		Euclidean Distances
X	Y	X	Y	
335,17	292,61	335,15	292,61	0,01984463
383,59	295,70	383,59	295,69	0,00777605
434,06	298,96	434,07	298,95	0,01469423
485,69	302,21	485,71	302,20	0,0203803
537,12	305,30	537,13	305,30	0,0157595
586,96	308,20	586,96	308,20	0,0042785
634,35	310,95	634,35	310,97	0,01653604
326,45	317,83	326,44	317,83	0,01145373
377,91	321,45	377,91	321,45	0,00226153
431,56	325,13	431,56	325,13	0,00808022
486,43	328,59	486,45	328,58	0,01194603
540,99	331,63	541,00	331,62	0,00720642
593,72	334,25	593,71	334,25	0,00770053
643,73	336,63	643,72	336,64	0,01717408
316,95	346,39	316,95	346,40	0,00808859
371,84	350,54	371,84	350,54	0,00356484
428,96	354,63	428,96	354,63	0,00111
487,33	358,25	487,33	358,25	0,00310556
545,24	361,17	545,23	361,17	0,0046806
601,08	363,48	601,07	363,48	0,01129662
654,02	365,46	654,01	365,47	0,01379954
306,54	378,89	306,55	378,90	0,00853246
365,25	383,53	365,25	383,53	0,00594254
426,25	387,94	426,24	387,95	0,00910543
488,39	391,66	488,38	391,67	0,00503934
549,92	394,44	549,91	394,43	0,00379494
609,22	396,42	609,21	396,42	0,00818352
665,42	398,00	665,42	398,00	0,00284007
295,38	415,82	295,41	415,81	0,02172188
357,99	421,20	357,99	421,20	0,00281603
423,31	425,86	423,30	425,86	0,00977657
489,64	429,60	489,63	429,60	0,0053831
555,19	432,20	555,19	432,20	0,00493572
618,34	433,85	618,35	433,84	0,01173382
677,83	434,63	677,85	434,63	0,02156895

4.2.2 Prediction of User Position

The results of the prediction of the (x, y) positions on the beacon chip used by the students are calculated using the weighted addition algorithm. By using the k-Nearest Neighbors (kNN) method, the determination of k-optimal can be seen based on the results of the smallest average error of each manual calculation of position prediction. The resulting research shows the smallest average error results using k-32 with an average error of 92.35.

From the results of the k-32 determination, the prediction for the new coordinate position is obtained. The new coordinates are used as position verification with the beacon chip user. The real *x* and *y* coordinates are determined manually based on a person's position in the research area when taking reference point data. For the new coordinate, which is written in the *x'* and *y'* column is the prediction of a person's location which is generated from real coordinates. Meanwhile, the euclidean value is used to see the performance of the prediction results.

Table 2: Performance of position prediction

Test Point	Average Error	Maximum Error	Minimum Error
1	79,77	126,87	29,88
2	94,37	146,06	22,37
	92,45	141,52	30,66
3	64,58	88,67	21,87
	89,92	155,17	33,22
	91,70	144,33	22,41
4	67,44	106,73	25,77
	120,01	150,79	90,43
	91,70	144,33	22,41
	57,84	99,88	15,28
5	87,75	134,36	30,04
	64,42	114,36	21,53
	145,69	157,90	134,03
	110,18	146,81	52,57
	93,18	126,34	68,49

4.2.3 Face Detection

The biometric system in facial recognition is used to determine facial identity using face analysis algorithms [20]. Detection of the student's face was carried out using OpenFace to determine the (*x*, *y*) coordinates of the student's face in the image captured by the camera and used to verify the correctness between the face position and the student's prediction.

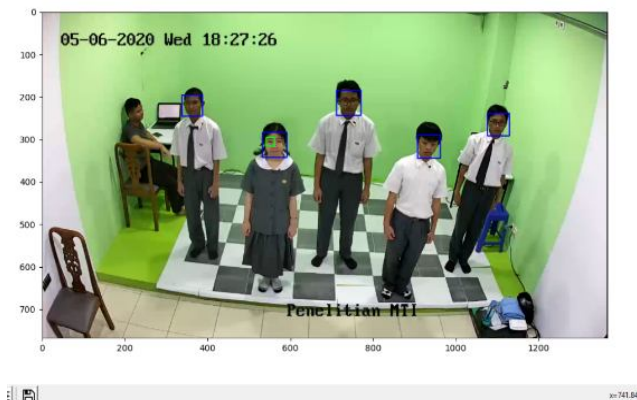


Figure 5: Face detection results

Images for face detection are carried out for test points 1 to 5 with each of the test points recorded in the (*x*, *y*) coordinate position on the student's face according to the position used for the face verification process.

4.2.4 Face Verification

For face verification at each test point, the level of face correctness with the predictive position is calculated based on the closest face with an accuracy rank that shows potential accuracy can be achieved according to the number of face verifications for each test data. Rank 1 means that the correct face is the closest face to the predicted face position, rank 2 means that the correct face is the two closest faces to the predicted face position until it reaches rank *n* which is the *n* closest face to the predicted face position.

In face matching, the face verification process is performed manually with rank conditions. 2D image coordinates are reference points for face coordinates that have been determined based on face detection for determining the closest face based on rank. Meanwhile, the coordinate of the 3D-2D projection results is the result of the calibration of the student's position predictions when carrying the beacon chip. The rank determination that has been previously explained will be given a number based on the closest order and it is considered true to the face of the soul according to the rank provisions. The results of all ranks are presented in Table 3, which shows the correct level of facial accuracy based on the comparison of each test point with the total of all test points.

Table 3: Rank face verification accuracy with face matching

Rank	Test Point	True Total Accuracy
1	10 : 150	33 %
2		66 %
3		80 %
4		93 %
5		100 %

The user verification process is also carried out using the OpenFace method based on the face similarity score. The rank assignments that have been previously described will be numbered according to the closest sequence and are deemed to correspond to the face of the soul according to the terms of rank. The results of all rankings are presented in Table 4, which shows the correct level of facial accuracy based on the comparison of each test point with the total of all test points.

Table 4: Rank face verification accuracy using OpenFace

Rank	Test Point	True Total Accuracy
1	10 : 150	17 %
2		44 %
3		65 %
4		87 %
5		100 %

The face verification process is done by OpenFace to get the truth of the face of the user. It is done by comparing the similarity scores of the two faces' similarities by calculating the L2 squared distance between their representations. A lower score indicates two faces are more likely than the same person. Because the representation is in the hypersphere unit, scores range from 0 (same picture) to 4.0.

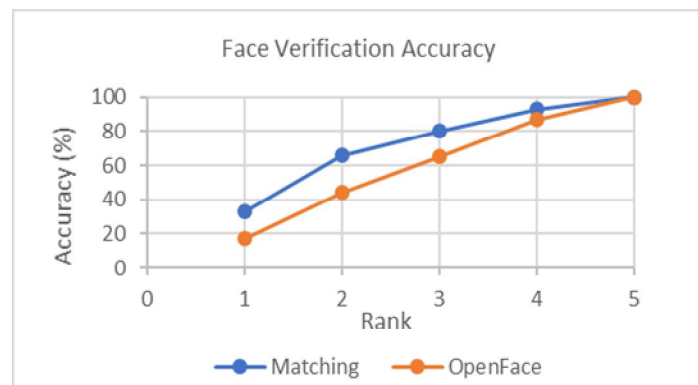


Figure 6: Comparison of face verification accuracy using face matching and OpenFace

This figure shows the comparison of predictive facial truths manually and by using the OpenFace method showing the level of truth of the face of the beacon chip user with an average difference of **11.8%** similarity level by considering the difference between the average level of facial similarity approaching **0%**.

5. CONCLUSION

The use of fingerprint and camera calibration techniques can be done with a statically mounted camera. The position of a person can be determined by using a beacon as a transmitter which gives a signal to the receiver to determine the position of a person when passing through the monitoring area, and verifying the identity can be done when the face of the beacon holder is caught on a CCTV camera. The combination of two beacons (Bluetooth low energy) and CCTV to verify the presence of students with real faces can be done using the indoor positioning system method and calibration of CCTV cameras. So that it can simplify the problem on face identification to face verification.

REFERENCES

1. K. M. Alam, M. Saini, and A. El Saddik, **Toward social internet of vehicles: Concept, architecture, and applications**, *IEEE Access*, 2015.
2. S. Adhikari, A. Ghosh, P. Das, S. Roy, and R. Bose, **An image based activity recognition**, *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 5, pp. 1873–1878, 2020.
3. J. Zambada, R. Quintero, R. Isijara, R. Galeana, and L. Santillan, **An IoT based scholar bus monitoring system**, in *2015 IEEE 1st International Smart Cities Conference, ISC2 2015*, 2015.
- 4.
5. Y. Wang, Q. Yang, G. Zhang, and P. Zhang, **Indoor Positioning System Using Euclidean Distance Correction Algorithm with Bluetooth Low Energy Beacon**, *2016 Int. Conf. Internet Things Appl.*, pp. 243–247, 2016.
6. K. Sornalatha and V. R. Kavitha, **IoT based smart museum using Bluetooth Low Energy**, *Proc. 3rd IEEE Int. Conf. Adv. Electr. Electron. Information, Commun. Bio-Informatics, AEEICB 2017*, pp. 520–523, 2017.
7. T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L. P. Morency, **OpenFace 2.0: Facial behavior analysis toolkit**, *Proc. - 13th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2018*, pp. 59–66, 2018.
8. L. Li, Z. Jun, J. Fei, and S. Li, **An incremental face recognition system based on deep learning**, *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. MVA 2017*, pp. 238–241, 2017.
9. M. Nehru and S. Padmavathi, **Illumination invariant face detection using viola jones algorithm**, *2017 4th Int. Conf. Adv. Comput. Commun. Syst. ICACCS 2017*, pp. 4–7, 2017.
10. D. P. Yudha, B. I. Hasbi, and R. H. Sukarna, **Indoor Positioning System Berdasarkan Fingerprinting Received Signal Strength (Rss) Wifi Dengan Algoritma K-Nearest Neighbor (K-Nn)**, *Ilk. J. Ilm.*, vol. 10, no. 3, p. 274, 2018.
11. L. Song, W. Wu, J. Guo, and X. Li, **Survey on camera calibration technique**, *Proc. - 2013 5th Int. Conf. Intell. Human-Machine Syst. Cybern. IHMSC 2013*, vol. 2, pp. 389–392, 2013.
12. Q. Zhang, C. Zhang, J. Ling, Q. Wang, and J. Yu, **A Generic Multi-Projection-Center Model and Calibration Method for Light Field Cameras**, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PP, no. c, p. 1, 2018.
13. V. S. Kumar, G. Spandan, B. Chakri, and D. Babu, **Fingerprint authentication and mobile app based monitoring of vehicles using IOT**, *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 5, pp. 1785–1789, 2020.
14. H. Chabbar and M. Chami, **Indoor localization using Wi-Fi method based on Fingerprinting Technique**, *2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS)*, Fez, 2017, pp. 1–5, doi: 10.1109/WITS.2017.7934613. T. Han and L. Ding,
15. **An indoor positioning technology in the BLE mobile payment system**, *AIP Conf. Proc.*, vol. 1839, no. May, 2017.
16. F. Jin and X. Wang, **An autonomous camera calibration system based on the theory of minimum convex hull**, *Proc. - 5th Int. Conf. Instrum. Meas. Comput. Commun. Control. IMCCC 2015*, pp. 857–860, 2016.
17. R. Tsai, **Versatile Camera Calibration Techniaue for High-Accuracy 3D Machine Vision Metrology**, *Robot. Autom.*, no. 4, 1987.
18. T. Gee *et al.*, **Tsai camera calibration enhanced**, *Proc. 14th IAPR Int. Conf. Mach. Vis. Appl. MVA 2015*, pp. 435–438, 2015.

19. M. S. I. Sameem, T. Qasim, and K. Bakhat, **Real time recognition of human faces**, *ICOSST 2016 - 2016 Int. Conf. Open Source Syst. Technol. Proc.*, pp. 62–65, 2017.
20. D. Li, H. Zhou, and K. M. Lam, **High-resolution face verification using pore-scale facial features**, *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2317–2327, 2015.
21. S. H. Masood and S. Riza, **Artificial Intelligence Face Recognition for applicant tracking system**, *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 1, pp. 8–11, 2020.