# Prototype of Student Attendance Application Based on Face Recognition Using Eigenface Algorithm

Tio Eko Prabowo[1], Rudy Hartanto[2], Sunu Wibirama[3]

***Abstract*—Prototype of face recognition based attendance application that has been developed to overcome weaknesses in DTETI UGM student manual attendance system has several weaknesses. These weaknesses are a decrease in facial recognition accuracy when operating under conditions of varying environmental light intensity and in condition of face rotating towards z axis rotation centre. In addition, application prototype also does not yet have a database to store attendance results. In this paper, a new application prototype has been developed using Eigenface face detection and recognition algorithm and Haar-based Cascade Classifier. Meanwhile, to overcome prototype performance weaknesses of the previously developed application, a pre-processing method was proposed in another study was added. Processes in the method were geometry transformation, histogram levelling separately, image smoothing using bilateral filtering, and elliptical masking. The test results showed that in the category of various environmental light intensity conditions, face recognition accuracy from developed application prototypes was 16.71% better than previous application prototypes. Meanwhile, in category of face slope conditions at z axis rotation centre, face recognition accuracy from developed application prototype was 38.47% better. Attendance database system was also successfully implemented and running without error.**

***Keywords*— Eigenface, Haar-Based Cascade Classifier, Image Pre-Processing, Attendance Application, Face Recognition.**

## I. Introduction

According to [1], education has long been seen as an activity that is not only mere information delivery, but also requires students' active mental and physical involvement in its process. Therefore, students' presence in class is very important to enable active teaching and learning interactions occurrence between educators and students. In order to monitor students' attendance, a reliable attendance recording system or attendance system is needed.

At present, student attendance system at Department of Electrical Engineering and Information Technology of the University Gadjah Mada (DTETI UGM) is still manual. Attendance process on manual attendance system is done by manually marking attendance list sheet. The advantage of this system is that it does not require high costs. However, this system has several disadvantages, i.e. attendance results transferring process into database is very vulnerable to human error and it is also time consuming. To overcome this weakness, a new automatic attendance system is required. Attendance process in automatic attendance is carried out automatically by electronic devices. Benefits of using automatic attendance system compared to manual attendance is that attendance results already in digital form so that it is easier to process and can directly enter the database. Meanwhile, automatic presence system weakness is that its development and/or implementation requires a higher cost than manual presence system.

Until now, several technologies that can be used in automatic presence systems include Barcode technology [2], Radio Frequency Identification (RFID) [3], fingerprint recognition [4], iris recognition [5], facial recognition [6], and multimodal technology [7]. Fingerprint recognition technology, iris recognition, and face recognition are biometric-based technologies. Biometrics is a method for recognizing one's identity automatically based on physical characteristics and behaviour [8]. Meanwhile, Barcode and RFID technology is based on non-biometrics. Identification base of Barcode technology is a barcode that can be read using a scanner, while identification base of RFID technology is radio waves that can be read using a wave reader. Barcode and RFID technology have several advantages, namely its easy use and very high accuracy. However, Barcodes and RFID also have weaknesses, namely if students forget to bring object identification, then attendance process cannot be carried out. Meanwhile, in biometrics technology, basis of its identification uses physical characteristics and human behaviour. Human physical characteristics and behaviour are always attached to human body unless it is lost/damaged due to aging, accident, or because it is indeed born defective. This makes biometrics technology far more practical than non-biometric technology which requires users to always carry object identification.

In this paper, a prototype application for biometric technology based attendance was developed. Biometrics technology was selected as basis for identification technology due to its practicality. Meanwhile, employed biometrics technology is facial recognition technology. Research on application development of DTETI UGM students' attendance based on facial recognition technology has been done [9]. In that study, an attendance application prototype was developed using the OpenCV library. The employed algorithms were Haar-based Cascade Classifier algorithm in face detection process and Local Binary Pattern Histogram (LBPH) algorithm in the face recognition process. Meanwhile, image pre-processing was only in a form of image transformation into grey images and image size alteration. In the conducted tests,

[1] *Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada, Jln. Grafika 2, UGM Yogyakarta 55281 (e-mail: tio.eko.prabowo@mail.ugm.ac.id)*

[2,3] *Lecturer, Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada, Jln. Grafika 2, UGM Yogyakarta 55281 (e-mail: rudy@ugm.ac.id, sunu@ugm.ac.id)*

performance of that application prototype was quite satisfactory, although it was not yet 100% successful, it detected and recognized all faces of tested test subjects. The prototype application [9] was then redeveloped to be retested using new data. Test materials were in a form of images and videos. Test result of application prototype [9] in seven test videos with face conditions perpendicular to camera and using training image on same light intensity as test environment light intensity showed very good results, i.e. average face recognition accuracy of 100% However, when tested on different circumstances, this application prototype has decreased the face recognition average accuracy. When application prototypes were tested on different environmental light intensities using training images taken at different light intensity conditions, the average accuracy of facial recognition was reduced to 82.67%. In addition, when application prototypes were tested on faces which rotated by 5° with z axis rotation centre, face recognition average accuracy also decreased, which was to 64.68%. This application prototype [9] also had weaknesses, i.e. it did not have a database system to store attendance results.

This conducted research aimed to create a prototype with a better performance compared to previous application prototype [9]. The performance that will be improved are face recognition accuracy on various environmental light intensity and face recognition accuracy on the face condition which rotates with centreof z axis. In addition, a database system will be added to store attendance results. In this paper, application development is still carried out using OpenCV library. Face detection process still use Haar-based Cascade Classifier algorithm. Meanwhile, for face recognition process, OpenCV library has had three classes of face recognition algorithms, namely Eigenface, Fisherface, and Local Binary Pattern Histogram (LBPH). Eigenface algorithm was selected as a face recognition algorithm in this paper. To increase facial recognition accuracy, both on tilted face caused by rotation of head centred on z axis and in changing light intensity situations, it is necessary to add a pre-processing facial images method. The utilized image pre-processing method was a pre-processing method that has been developed [10]. In addition to pre-processing method, SQL-based database systems were also added to a developed prototype.

After prototype application was successfully developed, its performance was compared to performance of benchmark application prototype [9] which had been redeveloped. It is expected that developed prototype application can provide better performance results compared to benchmark application prototype [9] from previous studies results.

## II. Face Recognition

A biometric technology used in this paper is face recognition technology. Face recognition technology was selected because according to [11], being compared to fingerprint recognition, face recognition is more hygienic because there is no physical contact by many users. Whereas when compared to iris recognition, face recognition is more cost effective because of a cheaper tool. To overcome face recognition weakness in a form of system dependence on environmental light condition changes, face expressions, and pose/face inclination, a compensation by using face recognition algorithms and a good image pre-processing was used. On the other hand, when compared to multimodal technology, face recognition technology is cheaper because multimodal technology uses more than one technologies.

In this paper, application development was still carried out using OpenCV library. When compared with MATLAB, OpenCV library is processed with faster computation time, uses smaller storage resources, and it has unpaid licenses [12]. Haar-based Cascade Classifier algorithm is still used for face detection process. Reference [13] shows that Haar-based Cascade Classifier has better performance compared to several other object detection algorithms such as Successive Mean Quantization Transform (SMQT) and Sparse Network of Winnows (SNOW), Neural Network (NN), and Support Vector Machines (SVM). Meanwhile, for face recognition process, OpenCV library has had three classes of face recognition algorithms, namely Eigenface, Fisherface, and Local Binary Pattern Histogram (LBPH). The Eigenface algorithm was selected as a face recognition algorithm in this paper because according to [14], Eigenface algorithm performance is slightly better than Fisherface algorithm. Meanwhile, according to [15], compared to LBPH algorithm, Eigenface algorithm has a faster but less accurate recognition processing time.

Meanwhile, to increase facial recognition accuracy, both on tilted face caused by rotation of head centred on z axis and in changing light intensity situations, it is necessary to add an appropriate pre-processing facial images method recognition. In [16] a pre-processing method was developed consisting of four stages, namely cropping, resizing, normalization, and filtering. Test results showed that the pre-processing method succeeded in increasing average accuracy of Eigenface algorithm by 66.32%. On the other hand, in [17] a facial pre-processing method was developed with stages corresponding to stages in [16], but with a more detailed and better method. In a normalization stage, pre-processing method developed in [17] had a geometric transformation process and a histogram levelling process separately. The geometry transformation process was useful for overcoming tilted faces caused by head rotation centred on the z axis, while separate histogram levelling processes were useful for dealing with situations of changing light intensity affecting facial recognition performance. Therefore, pre-processing method in [17] was selected as pre-processing method on application prototypes developed in this paper. In addition to the pre-processing method, SQL-based database systems were also added to application prototypes. SQL was selected over NoSQL because its presence data structure was fixed and in a form of a table. MySQL was selected as a relational database system in this paper because compared to Oracle, MySQL had a non-paid and user-friendly license [18]. Meanwhile, compared to PostgreSQL, MySQL required shorter computation time in simple read operations, it was easier to use and easier to install [19].

## III. Research Method

### A. *Differences between Benchmark Application Prototype and Application Prototype Developed in this Study*

Application prototypes that had been developed before [9] was used as a benchmark application against application prototype developed in this paper. Difference between benchmark application prototype [9] with the developed application prototype is explained in Table I. Meanwhile, differences in pre-processing method between two prototypes are explained by Fig. 1.

TABLE I
DIFFERENCES BETWEEN BENCHMARK APPLICATION PROTOTYPE WITH AN APPLICATION PROTOTYPE DEVELOPED IN THIS PAPER

| **Distinguishing category** | **Prototype of benchmark application [9]** | **Application prototype developed in this paper** |
|---|---|---|
| Facial recognition algorithm | Local Binary Pattern Histogram | Eigenface |
| Face pre-processing method | Resizing | Geometry transformation, separate histogram levelling, smoothing, and elliptical masking |
| Attendance database system | - | MySQL database |

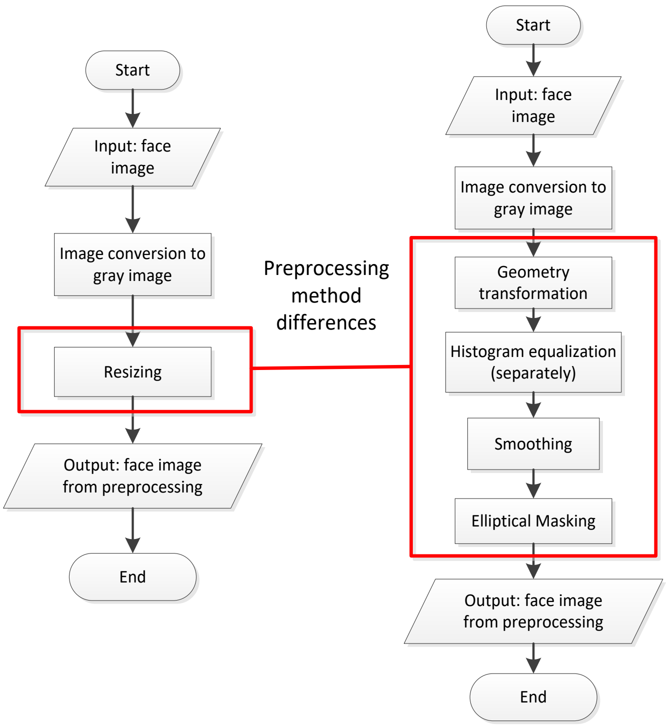



Fig. 1 Pre-processing method differences.

*1) Geometry Transformation:* A success of face recognition process depends on homogeneity of position and training face image and face image that will be recognized. Geometry transformation process was carried out to homogenize all images. In addition, geometric transformation was also done to cut unnecessary parts such as hair, forehead, and chin. There were three stages in geometry transformation namely rotation, scaling, and translation.

Rotation was carried out to straighten a titling face because z axis rotation. To be able to rotate face, first seek out coordinates of midpoint between two eyes as rotation centre and tilt angle between two eyes as face's tilt angle. Once those two values were obtained, rotation could be done.

Meanwhile, scaling was done to change facial images size but it still used a same pixel size/scale from original image. In this paper, based on an empirical test result, ideal image size as a training image to be trained and recognized was 70x70 pixels. Features calculated as a scale reference was distance between eyes. According to [17], in order to have hair, forehead, and chin be cut off, coordinates of ideal position from left eye according to calculations on paper were 0.16 of total image width (x coordinates) and 0.14 of total image height (y coordinates). From this information, desired distance between eyes was ((1-0.16) * 70) - (0.16 * 70) = 47.6. After obtaining the desired eye distance, scale was obtained using (1).

$$scale = \frac{47.6}{right\_x - left\_x}. \tag{1}$$

Description:
*left_x* = left eye point of input image on x axis
*right_x* = right eye point of input image on x axis.

Translation was the last process of geometry transformation. Translation is a process of position/coordinates shifting aimed to shift midpoint position between eyes towards desired position. According to [17], an ideal position of midpoint between these eyes is 0.5 of image width at x coordinate and 0.14 from image height at y coordinate. Based on this information, translation could be done using (2) and (3).

$$shift\_x = (0.5 * 70) - x = 35 - x \tag{2}$$

$$shift\_y = (0.5 * 70) - y = 35 - y. \tag{3}$$

Description:
*shift_x* = magnitude of midpoint coordinates shifts between eyes on x axis
*shift_y* = magnitude of midpoint coordinates shifts between eyes on y axis,
*x* = coordinates of midpoint between input eyes on x axis
*y* = coordinates of midpoint between input eyes on y axis.

*2) Separate Histogram Levelling:* According to [17], histogram levelling on face images needs to be done separately because light can come from anywhere on one side of face and then shadow on the other side of face. Separate histogram levelling was carried out starting from 0.25 left part image, 0.5 middle part image, and 0.25 right side image.

*3) Smoothing:* Smoothing step was carried out aiming to reduce pixel noise caused by histogram equalization. Smoothing was carried out using bilateral filtering method. According to [17], bilateral filtering is very good at smoothing most images while still keeping image edges sharp.

*4) Elliptical Masking:* This stage was carried out aiming to remove angles on face image such as neck that often got shadows, especially if face was not really facing straight to camera [17]. In order not to interfere with performance, the part was removed by masking face image. Masking was done by first making a white elliptical mask with a black background. The ellipse was made with a horizontal radius of 0.5, a vertical radius of 0.8, and its midpoint coordinates, which were 0.5 on x-axis and 0.4 on y-axis. When applied to face images, ellipse mask background was changed to grey so as not to contrast too much with other colour parts in face image [17].

### B. Test Method

Test was divided into two, i.e. functionality test and performance test. Functionality test aimed to test application prototype if it could run well or not. Functionality test was carried out by testing existing menus. Meanwhile, performance test was carried out aiming to know whether there was an increase or not in developed application prototype, compared to performance of benchmark application prototype [9]. In functionality test, there were two performances categories sought out. These categories were face recognition accuracy and video processing time. Percentage value of face recognition accuracy was calculated using (4).

$$accuracy = \frac{number\ of\ successfully\ known\ faces}{total\ video\ frames} * 100\% \quad (4)$$

Meanwhile, video processing time was calculated using QueryPerformanceFrequency () and QueryPerformanceCounter () functions. Wilcoxon Signed-Rank test was conducted to find out that there was a significant increase or decrease in performance between compared prototypes. This test was chosen compared to T test because histogram from obtained test data was not normally distributed [20]. This test was conducted in one direction (one-tail) with a significance value of 0.05

## IV. Results and Discussion

### A. Application Development Results

The developed application prototype had two interface displays, i.e. main interface display as shown in Fig. 2 and command prompt interface display as shown in Fig. 3.

Fig. 2 is main page display of application prototype that has been developed. This page is equipped with three menus in upper left and guide text at bottom to help users operate application prototypes. If there were faces detected on the frame, indicator would appear in a form of a yellow line on the face. If the detected face had eyes detected, an indicator would appear in a form of a red circle in each eye. Meanwhile, Fig. 3 is a display of command prompt window page. This page serves as an interface for users to interact with application prototypes, especially in inputting data in a form of text. The majority of responses from prototype applications were submitted through this page.

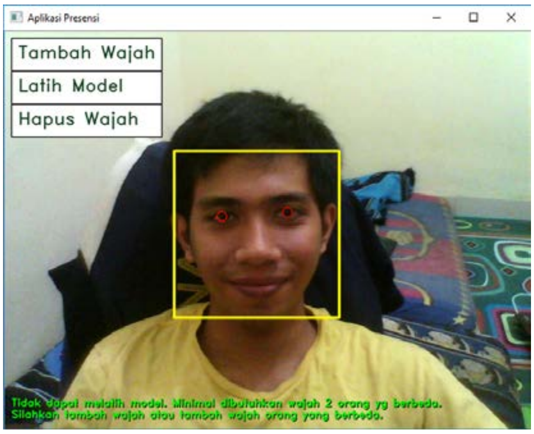



Fig. 2 Main interface display.

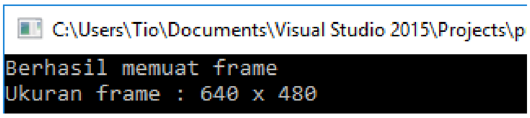



Fig. 3 Command prompt interface display.

### B. Test Results of Attendance Process

Attendance process occurred when application was in face recognition mode. This process started when face was successfully recognized 15 times in a row. Of the 15 times of face recognition, the most recognizable NIMs were taken for subsequent performing attendance to MySQL database through MySQL Connector/C ++ library. By MySQL Connector/C ++ library, NIM was passed on a query to determine whether the attendance would succeed or not. Attendance process would be successful if previously user had been registered in attendance database by taking courses on same day and hour as when the user did the attendance. In addition, at attendance time user must be within course validity period. Success of attendance process was indicated by a notification in a form of sound effect and text notifications in *cmd* as shown in Fig. 4.

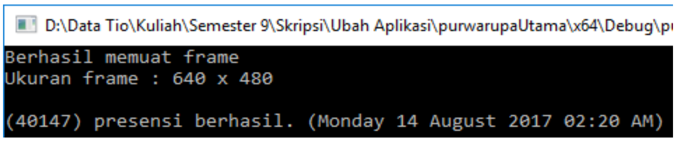



Fig. 4 Application responses to a successful attendance process.

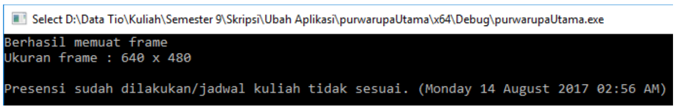
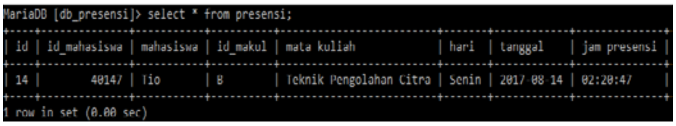



Fig. 5 Application responses to failed attendance process.

Attendance process can only be done once in a course session to avoid multiple data. If there is a user who does attendance but is not registered taking a course at same time as the attendance, then a notification will appear in a form of a sound effect and a notification in a form of text in *cmd* as shown in Fig. 5. The same thing will also happen to users who do attendance twice.

When the attendance process is successful, then attendance will be recorded in attendance database, especially in attendance table. The recorded data is in a form of id, student_ID, student, course_id, subjects, day, date and attendance time as shown in Fig. 6.



```
MariaDB [db_presensi]> select * from presensi;
+----+---------------+-----------+----------+--------------------------+-------+------------+--------------+
| id | id_mahasiswa  | mahasiswa | id_makul | mata kuliah              | hari  | tanggal    | jam presensi |
+----+---------------+-----------+----------+--------------------------+-------+------------+--------------+
| 14 |         40147 | Tio       | 8        | Teknik Pengolahan Citra  | Senin | 2017-08-14 | 02:20:47     |
+----+---------------+-----------+----------+--------------------------+-------+------------+--------------+
1 row in set (0.00 sec)
```

Fig. 6 Table of attendance in attendance database.

## *C. Test Results of Application Prototype Performance*

This section discusses test results of performance comparison between prototype applications that have been developed with benchmark application prototypes [9].

*1) Comparison of Accuracy Performance and Face Recognition Processing Time in Various Light Intensity Conditions:* This test was carried out aiming to find out differences in accuracy performance and face recognition processing time between the developed application prototype with benchmark application prototypes [9], particularly in various light intensity conditions. Light intensity conditions categories used in this test were light intensity of 24 lux, 15 lux, and 7 lux. Test results are shown in Fig. 7 and Fig. 8.

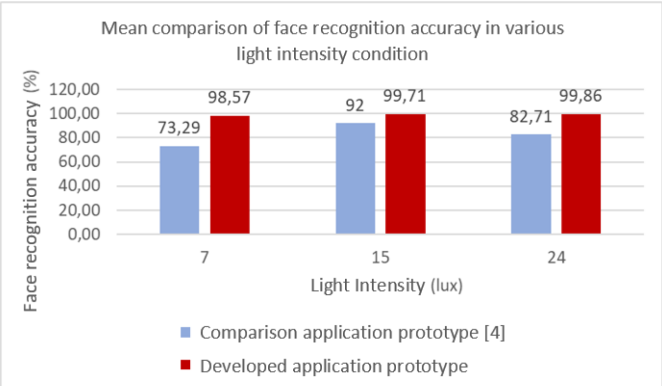



Fig. 7 Graph of face recognition accuracy value comparison.

In Fig. 7 it can be seen that average value of facial recognition accuracy from developed prototype exceeds average value of facial recognition accuracy from the benchmark application prototype [9] in all light intensity conditions. Average increase in facial recognition accuracy performance is ((98.57-73.29) + (99.71-92) + (99.86-82.71))/3 = 16.71%.

Meanwhile, in Fig. 8, it can be seen that average value of total video processing time from benchmark application prototype [9] is higher than average value of total video processing time from prototype application developed in this paper on all light intensity conditions. Average increase in video processing time performance is ((31.24-24.02) + (31.21-25.99) + (31.49-25.25)) / 3 = 6.22 seconds.

To determine significance of performance improvement, a Wilcoxon Signed-Rank test was carried out with a significance level of 0.05. The Wilcoxon Signed-Rank test results showed that there was a significant increase in performance on accuracy and total time of face recognition processing from application prototypes developed in this paper with an average increase of 16.71% and 6.22 seconds respectively.

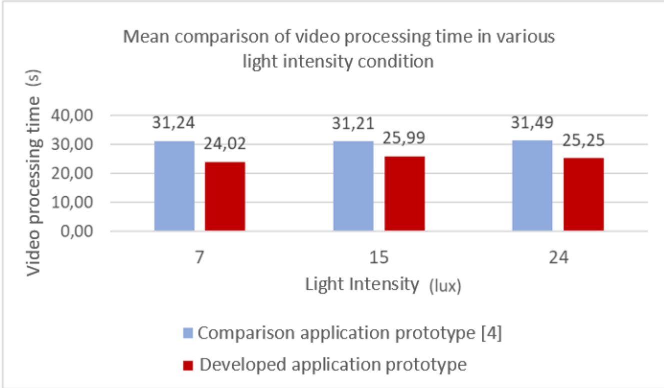



Fig. 8 Graph of video processing time total average value comparison.

*2) Comparison of Face Recognition Accuracy Performance in Tilted Face Conditions due to Z Axis Rotation:* This test was conducted aiming to find out differences in face recognition accuracy performances among application prototypes developed in this paper with benchmark application prototypes [9], especially on tilted face conditions due to z axis rotation. Categories of rotation angle magnitudes employed in this test were 5 °, 10 °, 15 °, and 20 °. Results of this test are shown in the following Fig. 9.

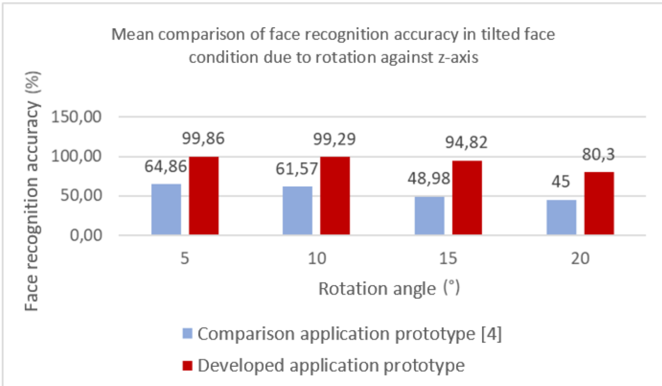



Fig. 9 Graph of face recognition accuracy value comparison.

In Fig. 9 it can be seen that average facial recognition accuracy values from developed application prototype exceeds face recognition average value from benchmark application prototype [9] in all categories of rotation angle. Average increase in facial recognition performance is equal to ((99.86-64.86) + (99.29-61.57) + (94.82-48.98) + (80.3-45)) / 4 = 38.47%.

To determine significance of performance improvement, a Wilcoxon Signed-Rank test was carried out with a significance level of 0.05. The Wilcoxon Signed-Rank test results showed that there was an increase in facial recognition accuracy performance which was significant from developed prototype application in this paper which was 38.47% on tilted face conditions due to z axis rotation.

## V. Conclusions

Based on the tests that have been conducted, some conclusions can be drawn as follows. There was a significant increase in facial recognition accuracy performance from prototype applications that had been developed under conditions of various environmental light intensity with an average performance increase of 16.71%. In addition, there was a significant increase in facial recognition processing time performance from developed application prototypes in this paper on condition of various environmental light intensity represented by video processing time with an average increase in time of 6.22 seconds. Furthermore, there was also a significant increase in face recognition accuracy performance from application prototype which had been developed under conditions of tilted face changes caused by head rotation with z axis rotation centre with an average increase of 38.47%. It can be concluded that attendance database system had been successfully implemented and worked without errors.

## REFERENCES


[1] A. Imron, “Mengatur Kehadiran dan Ketidakhadiran Peserta Didik,” in *Manajemen Peserta Didik Berbasis Sekolah*, 1st ed., Jakarta: Bumi Aksara, 2011, pp. 82–94.

[2] S.A.M. Noor, N. Zaini, M.F.A. Latip, and N. Hamzah, “Android-Based Attendance Management System,” *Proc. - 2015 IEEE Conf. Syst. Process Control. ICSPC 2015*, 2015, pp. 118–122.

[3] F.H. Saputra, “Sistem Absensi Menggunakan Teknologi RFID,” Bachelor Thesis, Universitas Indonesia, Jakarta, Indonesia, 2008.

[4] Q. Shafi, J. Khan, N. Munir, and N.K. Baloch, “Fingerprint Verification Over the Network and Its Application in Attendance Management,” *ICEIE 2010 - 2010 Int. Conf. Electron. Inf. Eng. Proc.*, Vol. 2, pp. 555–559, 2010.

[5] S. Kadry and M. Smaili, “Wireless Attendance Management System Based on Iris Recognition,” *Sci. Res. Essays*, Vol. 5, No. 12, pp. 1428–1435, 2010.

[6] D.F. Satria, “Sistem Presensi Otomatis Kelas Berbasis Face Recognition,” Bachelor Thesis, Universitas Gadjah Mada, Yogyakarta, Indonesia, 2017.

[7] E.G. Agulla, E.A. Rúa, J. L.A. Castro, D.G. Jiménez, and L.A. Rifón, “Multimodal Biometrics-Based Student Attendance Measurement in Learning Management Systems,” *ISM 2009 - 11th IEEE Int. Symp. Multimed.*, 2009, pp. 699–704.

[8] K.P. Khushk and A.A. Iqbal, “An Overview of Leading Biometrics Technologies Used for Human Identity,” *2005 Student Conference on Engineering Sciences and Technology*, 2005, pp. 1–4.

[9] N.O. Wijaya, “Prototype Aplikasi Presensi di Kelas Berbasis Face Recognition Menggunakan Closed-Circuit Television,” Bachelor Thesis, Universitas Gadjah Mada, Yogyakarta, Indonesia, 2014.

[10] S. Emami, “Face Recognition Using Eigenfaces or Fisherfaces,” in *Mastering OpenCV with Practical Computer Vision Projects*, Draft., Birmingham, UK: Packt Publishing, 2012, pp. 261–309.

[11] R. Jafri and H.R. Arabnia, “A Survey of Face Recognition Techniques,” *J. Inf. Process. Syst.*, Vol. 5, No. 2, pp. 41–68, Jun. 2009.

[12] U. Sinha (2010) “Why OpenCV?,” [Online], http://aishack.in/tutorials/opencv/, access date: 09-Jul-2018.

[13] K. Dang and S. Sharma, “Review and Comparison of Face Detection Algorithms,” *2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence*, 2017, pp. 629–633.

[14] M. Sharkas and M. A. Elenien, “Eigenfaces vs. Fisherfaces vs. ICA for Face Recognition; A Comparative Study,” *2008 9th International Conference on Signal Processing*, 2008, pp. 914–919.

[15] A. Ozdil and M.M. Ozbilen, “A Survey on Comparison of Face Recognition Algorithms,” *2014 IEEE 8th International Conference on Application of Information and Communication Technologies (AICT)*, 2014, pp. 1–3.

[16] K. Dharavath, G. Amarnath, F.A. Talukdar, and R.H. Laskar, “Impact of Image Preprocessing on Face Recognition: A Comparative Analysis,” *2014 International Conference on Communication and Signal Processing*, 2014, pp. 631–635.

[17] S. Emami, “Face Recognition Using Eigenfaces or Fisherfaces,” in *Mastering OpenCV with Practical Computer Vision Projects*, Draft., Birmingham, UK: Packt Publishing, 2012, pp. 261–309.

[18] (2018) “MySQL vs Oracle,” [Online], https://itxdesign.com/mysql-vs-oracle/, access date: 21-Jul-2018.

[19] O.S. Tezer (2014) “SQLite vs MySQL vs PostgreSQL: A Comparison Of Relational Database Management Systems,” [Online], https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems, access date: 21-Jul-2018.

[20] R. Lowry (2018) “Concepts & Applications of Inferential Statistics.” [Online], http://vassarstats.net/textbook/, access date: 17-Aug-2018.