Facial Recognition on System Prototype to Verify Users using Eigenface, Viola-Jones and Haar

Robin[1], Aldrick Handinata[2], Wenripin Chandra[3]
[1][2]Universitas Pelita Harapan Medan, Indonesia [3]STMIK Mikroskil, Indonesia
[1]robin.huang@lecturer.uph.edu, [2]ah70016@student.uph.edu, [3]wenripin@mikroskil.ac.id

ABSTRACT

Facial recognition is one of the most popular way to authenticate user into a system. This method is preferable considering the tendency of users for using the same password across multiple sites which made the user has already made his own account securities in vulnerable states. Using biometrics might supply solutions to solve this problem and facial recognition is one of the best biometric methods can be apply as a digital account security solution. This study to design a prototype system implementing facial recognition to verify users to measure how accurate these methods are. The method used here is Viola-Jones for face detection, Eigenface and Haar feature for face recognition from the OpenCV. The system was designed in Java. Based on the test results from the system designed, system can recognize user face with 100% accuracy if faces are shot in a well desirable condition. The system is able to recognize the user's face with various expressions including with or without glasses. However, the system has difficulty in recognizing user's face in facing up, down, sideways position or blocked by accessories or body parts such as hands. After some experiment, it was proven that the system designed is accurate, reliable and safe enough to be implemented to digital authorization process.

Keywords: Eigenface; Facial Recognition; OpenCV; System Prototype; Viola-Jones; Haar Feature

INTRODUCTION

Using username and password as method to verify user account still becomes primary choice due to its practicality. Although this method is usual, it also has its flaw, account owner tends to use same password across multiple sites. This practice eases hackers to gain access for the same user in multiple different sites (Chowhan & Tanwar, 2019). When a hacker had successfully gain certain user's digital security account and password, he can easily gain access on any different account of the same user from the same sites of even in the different sites easily. In this case, the user's digital account security was vulnerable which means putting the user digital account security in a potentially high risk state. To reduce the risk, using biometrics for gaining access to digital account might become one of the best solution which roles in user authentication process before the system granted user access into the system. One of the more popular utilization of biometric for authentication is facial recognition. Face is one of many part from the human body naturally used as the main identifier by human biologically due to its ease to utilize for identification visually.

This paper aims to develop a login system prototype that capable to verify user by using facial recognizing process features before the system decided to grant use access into the system or to deny it. This research also did some experiment on predicting user faces and how the accurate is. To study about the accuracy of facial recognition in the system developed, the Method used to detect face in this research is Viola-Jones and the method used to recognize and to predict faces in this research is the Eigenface which have already included in the OpenCV library and the facial recognition system in this research is designed using the Java programming language.

LITERATURE REVIEW

Image is an object representation as output of an optical data recording system (Sinaga, 2017). In. In order to be processed in computer, image must be digitalized into what we call digital image.

* Corresponding author

A digital image can be represented as a collection of elements called pixels arranged in a matrix with M rows and N columns. The greater values of M and N, the smoother the resulting image. As shows in Fig 1, each pixel has an x, y coordinate formed by intersection between row and column.
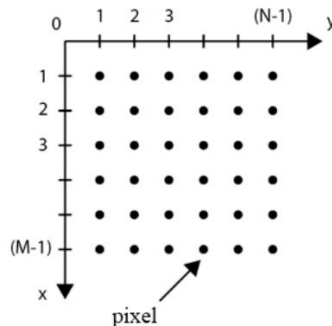

Fig. 1 Digital Image Coordinate

Artificial intelligence is field that study how to make machines have intelligence like humans. In artificial intelligence, machines are expected to solve problems automatically and accurately more than capability of human (Ether, 2017).

Computer vision is a field that study how computer recognize object and extract information from the observed object in form of image (Ramadhan, Jati, & Ahmad, 2016). According to Ramadhan, Jati and Ahmad, in implementing computer vision, there are some steps:

1. Image acquisition
   The process by which an image is captured by digital device that converts analog signals into digital data and is ready to be processed by computer.
2. Image processing
   The process to enhance image quality.
3. Image analysis
   The process to analyze processed image to extract information from the image.
4. Image understanding
   In this process, artificial intelligence applied to understand the image.

Face is one of biometric characteristics beside voice, fingerprint, retina that can be used to identify a human. (Rosyani, 2017). Human can differ or recognize the other by merely looking to his face. Human even can recognize the other human though the face is identical like in twins or changing to the face due to aging, using glasses or changing hair style.

Viola-Jones is one of face detection algorithm utilizing image classification based on feature value of an image. Image classification based on feature value of an image is faster compared to image classification based on pixels (Mahmudi, Z, & Kusrini, 2019).

Viola-Jones has four stages of process (Damanik, Sitanggang, Pasaribu, Siagian, & Gulo, 2018), those are:

1. Haar Feature selection
2. Integral Image creation
3. AdaBoost Training
4. Cascade Classifiers

Haar feature is feature extraction and classification method using rectangular feature. This method can give indication of a specific image. This method used to recognize object based on value of a feature, not by pixel value in input image (Chau, Banjarnahor, Irfansyah, Kumala, & Banjarnahor, 2019).

METHOD

Face images used for image training come from Universitas Pelita Harapan Medan students who were willing to become volunteers where each student was photographed in several different positions and conditions and one other face image was used for testing an unrecognized face.

* Corresponding author

The calculation of the haar feature value serves as a reference value in the Viola-Jones method. This calculation aims to determine whether in an image there is a desired feature. In this study, the features in question are features on the face, namely the eyes, nose, and mouth. The initial stage of this calculation is in the form of an input image that is converted first into a grayscale image. Fig. 2 shows the initial image converted to a grayscale image and then matched with the haar feature.

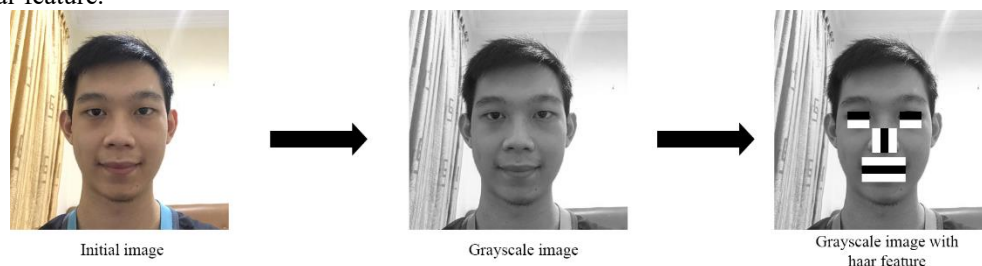Initial image   Grayscale image   Grayscale image with haar feature

Fig. 1 Grayscale Image with Haar Feature

These haar features are then matched over the entire image area. Fig. 3 shows the intensity value of the image converted to a 5x5 pixel image on the mouth as an example.

115	117	124	119	110
116	123	109	108	118
77	82	70	75	71
110	103	119	113	111
102	111	123	127	105

Fig. 2 Image Value by Matching the Haar Feature of the Mouth

Then the image is converted into an integral image in the form of a 5x5 matrix as shown in Fig. 4, making it easier to calculate the value of the haar feature in that section.

115	232	356	475	585
231	471	704	931	1159
308	548	863	1160	1463
418	761	1076	1492	2019
520	974	1289	1955	2587

Fig. 3 Integral Image with Matching Haar Feature of the Mouth

From Fig. 4, the value of the haar feature is calculated as follows:
$$F(Haar) = \sum F_{white} - \sum F_{black}$$

* Corresponding author

$$F(Haar) = [s(A) + s(C)] - s(B)$$

$$F(Haar) = [(1159 + 115 - 231 - 585) + (2019 + 308 - 418 - 1463)] - (1463 + 231 - 308 - 1159)$$

$$F(Haar) = [458 + 446] - 227$$

$$F(Haar) = 677$$

The haar feature value is then compared with a predetermined threshold for object detection. This threshold value is obtained from the Viola-Jones method through the Adaboost Training stage. If the haar feature value obtained is higher than the threshold value, it can be said that the area meets the haar feature criteria. Then all the haar features obtained in the image will be matched with the criteria for facial haar features. If all the criteria for haar facial features are met, then that area contains the object being observed, namely the face.

The calculation of the Eigenface value aims to predict the face image being tested with the face image dataset that has been trained. The initial stage of calculating the Eigenface value is to find the average value of the face image matrix in the dataset, then look for the Eigenface value of each face image. Fig. 5 shows greyscale pixel value of some part about 5x5 pixels of three sample face images in the dataset.

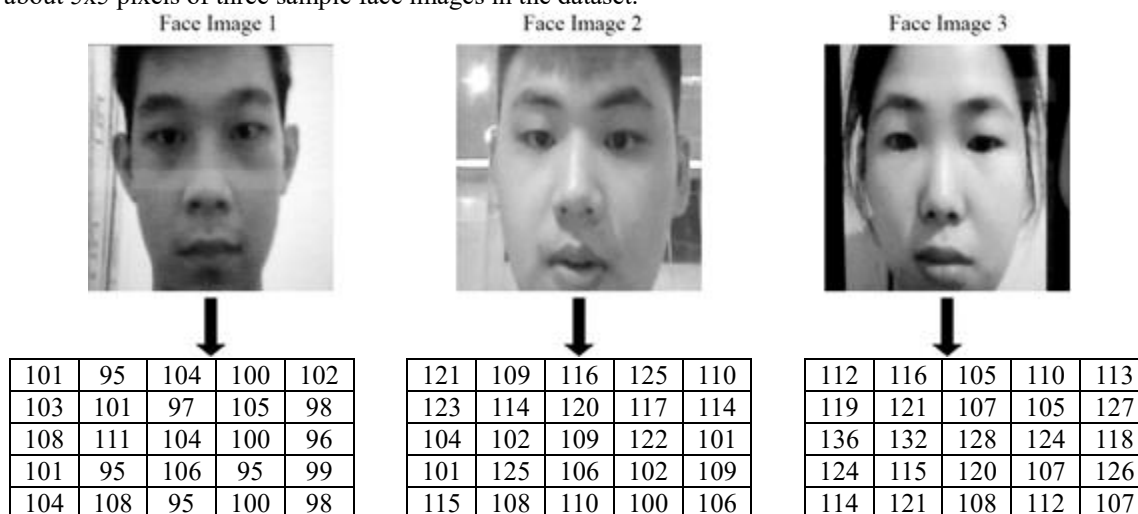| Face Image 1						Face Image 2						Face Image 3				
101	95	104	100	102		121	109	116	125	110		112	116	105	110	113
103	101	97	105	98		123	114	120	117	114		119	121	107	105	127
108	111	104	100	96		104	102	109	122	101		136	132	128	124	118
101	95	106	95	99		101	125	106	102	109		124	115	120	107	126
104	108	95	100	98		115	108	110	100	106		114	121	108	112	107

Fig. 4 Some Part of Grayscale Pixel about 5x5 Pixels Value of Three Sample Face Images.

The image then converted into an eigenvector and the average value of the matrix is calculated. Fig. 6 shows the calculation of the average face image matrix in the dataset.

Eigenvector image 1 =
[101 95 104 100 102 103 101 97 105 98 108 111 104 100 96 101 95 106 95 99 104 108 95 100 98]

Eigenvector image 2 =
[121 109 116 125 110 123 114 120 117 114 104 102 109 122 101 101 125 106 102 109 115 108 110 100 106]

Eigenvector image 3 =
[112 116 105 110 113 119 121 107 105 127 136 132 128 124 118 124 115 120 107 126 114 121 108 112 107]

Matrix average =
[[101 95 104 100 102 103 101 97 105 98 108 111 104 100 96 101 95 106 95 99 104 108 95 100 98] +
[121 109 116 125 110 123 114 120 117 114 104 102 109 122 101 101 125 106 102 109 115 108 110 100 106] +
[112 116 105 110 113 119 121 107 105 127 136 132 128 124 118 124 115 120 107 126 114 121 108 112 107]] / 3

* Corresponding author

Matrix average =
[111 107 108 112 108 115 112 108 109 113 116 115 114 115 105 109 111 111 101 111 111 112 104 104 104]

Fig. 5 Average Value of Face Image Matrix Dataset

After getting the average value of the matrix, the next step is to subtract the matrix value of each image to get the Eigenface value of the image. Fig. 7 shows the calculation of the Eigenface value for each face image in the dataset.
Eigenface image 1
= [101 95 104 100 102 103 101 97 105 98 108 111 104 100 96 101 95 106 95 99 104 108 95 100 98] –
  [111 107 108 112 108 115 112 108 109 113 116 115 114 115 105 109 111 111 101 111 111 112 104 104 104]
= [-10 -12 -4 -12 -6 -12 -11 -11 -4 -15 -8 -4 -10 -15 -9 -8 -16 -5 -6 -12 -7 -4 -9 -4 -6]
= -220
Eigenface image 2
= [121 109 116 125 110 123 114 120 117 114 104 102 109 122 101 101 125 106 102 109 115 108 110 100 106] –
  [111 107 108 112 108 115 112 108 109 113 116 115 114 115 105 109 111 111 101 111 111 112 104 104 104]
= [10 2 8 13 2 8 2 12 8 1 -12 -13 5 7 -4 -8 14 -5 1 -2 4 -4 6 -4 2]
= 53
Eigenface image 3
= [112 116 105 110 113 119 121 107 105 127 136 132 128 124 118 124 115 120 107 126 114 121 108 112 107] –
  [111 107 108 112 108 115 112 108 109 113 116 115 114 115 105 109 111 111 101 111 111 112 104 104 104]
= 176

Fig. 6 Eigenface Value of Face Image Dataset

To recognize the test face image, it begins with finding the Eigenface value of the test face image by reducing the value of the test image matrix with the average value of the trained image matrix. Fig. 8 shows the image of the test face in the form of 5x5 pixels matrix.

Test face image | Face image 5x5 pixel

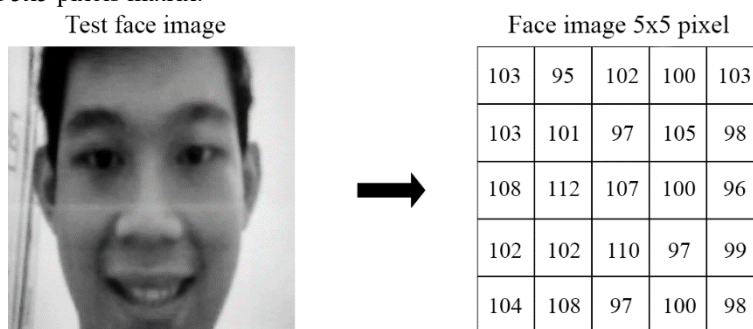| 103	95	102	100	103
103	101	97	105	98
108	112	107	100	96
102	102	110	97	99
104	108	97	100	98

Fig. 7 Test Face Image in the Form of 5x5 Pixels Matrix

The next step is to find the distance or the smallest difference from the Eigenface value of the tested face image with the trained image. This distance calculation is also known as Euclidean Distance. Euclidean distance can also be said as a confidence value. The smaller the value of the Euclidean distance, the more confident the predicted results will be. Fig. 9 shows the calculation of the Euclidean Distance value and the results of the prediction of the tested face image.
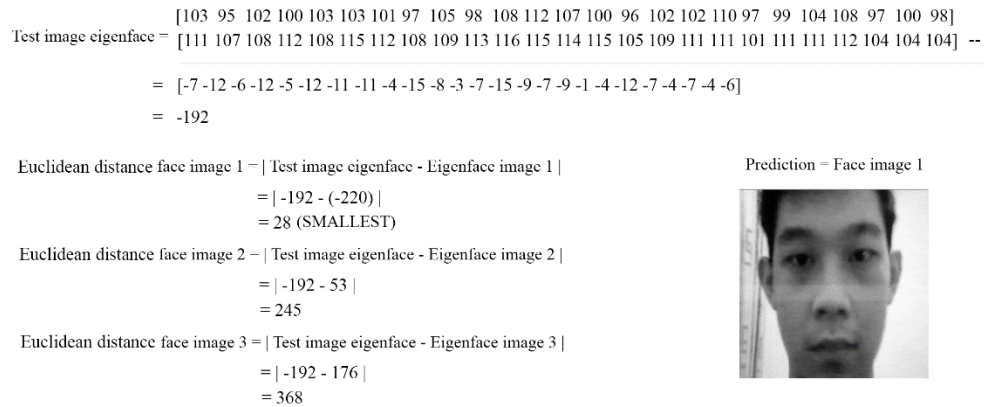
* Corresponding author

217

$$\text{Test image eigenface} = \frac{[103\ 95\ 102\ 100\ 103\ 103\ 101\ 97\ 105\ 98\ 108\ 112\ 107\ 100\ 96\ 102\ 102\ 110\ 97\ 99\ 104\ 108\ 97\ 100\ 98]}{[111\ 107\ 108\ 112\ 108\ 115\ 112\ 108\ 109\ 113\ 116\ 115\ 114\ 115\ 105\ 109\ 111\ 111\ 101\ 111\ 111\ 112\ 104\ 104\ 104]}\ --$$

$$= [-7\ -12\ -6\ -12\ -5\ -12\ -11\ -11\ -4\ -15\ -8\ -3\ -7\ -15\ -9\ -7\ -9\ -1\ -4\ -12\ -7\ -4\ -7\ -4\ -6]$$

$$= -192$$

Euclidean distance face image 1 = | Test image eigenface - Eigenface image 1 |
$$= |-192 - (-220)|$$
$$= 28\ (\text{SMALLEST})$$

Euclidean distance face image 2 = | Test image eigenface - Eigenface image 2 |
$$= |-192 - 53|$$
$$= 245$$

Euclidean distance face image 3 = | Test image eigenface - Eigenface image 3 |
$$= |-192 - 176|$$
$$= 368$$

Prediction = Face image 1

Fig. 8 Calculation of the Euclidean Distance Value of the Test Face Image

Fig. 9 shows the smallest difference in the Euclidean distance value from the test face image is the face image 1. So, the prediction results obtained are face image 1 with a Euclidean distance value of 28. The login workflow on the system prototype is the face dataset training process and the image prediction process. The face can be seen in Fig. 10 and Fig. 11 and the display of the results of the login implementation using face images can be seen in Fig. 12.
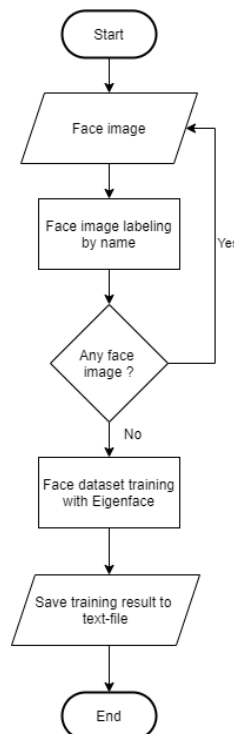

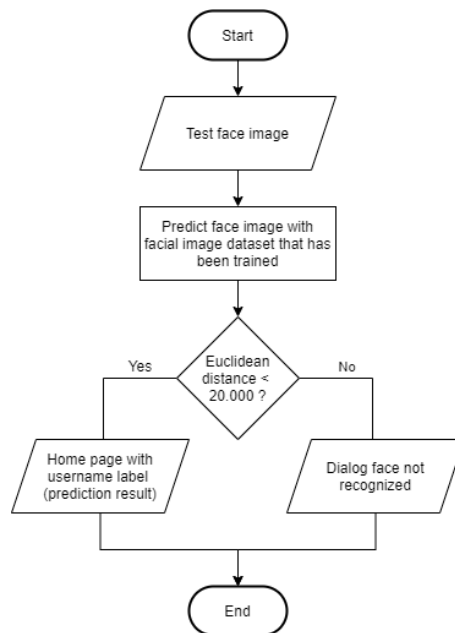Fig. 9 Flowchart of Face Image Dataset Training

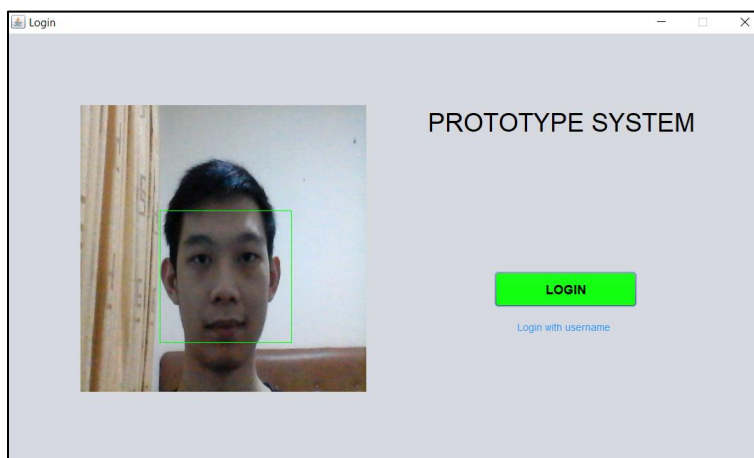* Corresponding author

Fig. 10 Flowchart of Face Image Predictions

Fig. 11 The System Prototype Interface in Recognizing Faces

* Corresponding author

RESULT

The complete face image dataset is shown in the Fig. 13:

Fig. 12 Training Face Images Dataset

DISCUSSIONS

The facial recognition experiment result from the designed system can be seen in Table 1.

Table 1 Face Recognition Results from the Designed System

Identity by Sequence from Fig. 13	Shot close to camera (shot in well condition)	Shot from far (unusual condition)	Shot when facing other side (unusual condition)	Shot when facing down (unusual condition)	Shot when facing up (unusual condition)	Shot wearing medical face mask (unusual condition)
Aldrick	√	√	X	√	X	√

* Corresponding author

Calvino	√	√	X	√	X	X
Griselda	√	√	X	X	X	X
Reynaldo	√	√	X	X	X	X
Wesley	√	X	X	√	X	X
Anton	√	X	X	X	X	X
Cynthia	√	√	X	X	X	√
Levina	√	√	X	√	X	√
Vincent	√	X	X	√	X	√
Fermin	√	√	X	X	X	X
Accuracy	100%	70%	10%	50%	10%	40%

From the experimental results in Table 1, it can be seen that if the face images are taken in good condition, namely with the correct distance and the condition of the face is close to the camera, the facial recognition results have an accuracy rate of 100%, whereas if the face image is taken in unusual conditions, the accuracy obtained is very unsatisfactory with an average accuracy of 36% by calculating the average from all accuracy result in unusual condition experiment. Experiments were also carried out on Vielbert facial image which were not previously entered into the training dataset and the system gave the result that Vielbert's face was not recognized at all.

CONCLUSION

The system designed is 100% accurate and reliable if the face images were shot in a good condition and correct distance close to the camera. Face recognition accuracy dropped when the face images shot in unusual condition such as facing up, facing down, facing beside, wearing medical face mask or face covered by something else.

REFERENCES

Afiyat, N. (2017). Analisis Restorasi Citra Kabur Algoritma Wiener Menggunakan Indeks Kualitas Citra. *Nusantara Journal of Computers and Its Applications*, 24-34.

Alexander, L. W., Sentinuwo, S. R., & Sambul, A. M. (2017). Implementasi Algoritma Pengenalan Wajah Untuk Mendeteksi Visual Hacking. *E-Journal Teknik Informatika*.

Chau, S., Banjarnahor, J., Irfansyah, D., Kumala, S., & Banjarnahor, J. (2019). Analisis Pendeteksian Pola Wajah Menggunakan Metode Haar-Like Feature. *JITE (Journal of Informatics and Telecommunication)*, 69-76.

Chowhan, R. S., & Tanwar, R. (2019). Password-Less Authentication: Methods for User Verification and Identification to Login Securely Over Remote Sites. *Machine Learning and Cognitive Science Applications in Cyber Security*, 190-212.

Damanik, R. R., Sitanggang, D., Pasaribu, H., Siagian, H., & Gulo, F. (2018). An Application of Viola Jones Method For Face Recognition For Absence Process Efficiency. *Journal of Pysics*, 1-8.

Detila, Q. M., & Wibowo, E. P. (2019, Desember). Perbandingan Metode Eigenface, Fischerface, dan LBPH pada Sistem Pengenalan Wajah. *Jurnal Ilmiah KOMPUTASI, 18*, 315-322.

Ether, W. (2017). *Introduction to Artificial Intellingence.* Ravensburg: Springer.

Krasmala, R., Purba, A. B., & Lenggana, U. T. (2017). Kompresi Citra Dengan Menggabungkan Metode Discrete Cosine Transfor (DCT) dan Algortima Huffman. *JOIN*, 1-9.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2020). Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, 261-318.

Mahmudi, Z, M. F., & Kusrini. (2019). Implementasi Metode Viola Jones Untuk Mendeteksi Wajah Manusia. *Jurnal INFORMA Politeknik Indonusa Surakarta*, 54-60.

Manurung, E. F. (2017). Implementasi Metode Median Filter dan Histogram Equalization Untuk Perbaikan Citra Digital. *Jurnal Pelita Informatika*, 67-71.

Raharja, B. D., & Harsadi, P. (2018). Implementasi Kompresi Citra Digital dengan Mengatur Kualitas Citra Digital. *Jurnal Ilmiah Sinus*, 71-77.

Ramadhan, B. W., Jati, A. N., & Ahmad, U. A. (2016). Implementasi Eigenface Untuk Pengenalan Wajah Pada Building Security System Berbasis Embedded. *e-Proceeding of Engineering*, 764-770.

* Corresponding author

Rosyani, P. (2017). Pengenalan Wajah Menggunakan Metode Principal Component Analysis (PCA) dan Canberra Distance. *Jurnal Informatika Universitas Pamulang*, 118-121.

S, A. B., Suma'inna, & Maulana, H. (2016). Pengenalan Citra Wajah Sebagai Identifier Menggunakan Metode Principal Component Analysis (PCA). *Jurnal Teknik Informatika*, 166-175.

Sinaga, A. S. (2017). Implementasi Teknik Thresholding Pada Segmentasi Citra Digital. *Jurnal Mantik Penusa*, 48-51.

Sulistyawati, D. H., & Utomo, H. S. (2016). Perbaikan Citra Dengan Noise Mission Block Menggunakan Implementasi Algoritma Projection Onto Convex Sets (POCS). *KONVERGENSI*, 11-19.

Utari, C. T. (2016). Implementasi Algoritma Run Length Encoding Untuk Perancangan Aplikasi Kompresi dan Dekompresi File Citra. *Jurnal TIMES*, 24-31.

Yang, H., & Wang, X. A. (2016). Cascade Classfier for Face Detection. *Journal of Algorithms & Computational Technology*, 187-197.

* Corresponding author