

Performance Comparison of Feature Face Detection Algorithm on The Embedded Platform

Ahmad Zarkasi¹, Siti Nurmaini^{1*}, Deris Setiawan¹, Bhakti Yudho Suprpto², Huda Ubaya¹, Rizki Kurniati¹

¹Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya

²Faculty of Engineering, Universitas Sriwijaya, Indonesia

*ahmadzarkasi@unsri.ac.id

ABSTRACT

The intensity of light will greatly affect every process carried out in image processing, especially facial images. It is important to analyze how the performance of each face detection method when tested at several lighting levels. In face detection, various methods can be used and have been tested. The FLP method automates the identification of the location of facial points. The Fisherface method reduces the dimensions obtained from PCA calculations. The LBPH method converts the texture of a face image into a binary value, while the WNNs method uses RAM to process image data, using the WiSARD architecture. This study proposes a technique for testing the effect of light on the performance of face detection methods, on an embedded platform. The highest accuracy was achieved by the LBPH and WNNs methods with an accuracy value of 98% at a lighting level of 400 lx. Meanwhile, at the lowest lighting level of 175 lx, all methods have a fairly good level of accuracy, which is between 75% to 83%.

Keywords: Face Detection Method, Light intensity, The FLP Method, The Fisherface Method, The LBPH Method, and WNNs Method.

1. INTRODUCTION

Humans are living creatures that have characteristics that can be used as a differentiator from other humans. One of the differences between humans and other humans is the face. Every human being has a different face shape [1][2] Each face has its characteristics or distinctive features. The face of each contains different information [3][4][5][6]. Examples are expression, gender, age, and also race [7]. Therefore, in biometrics technology or biological data recognition technology, faces can be used as identification [8][9][10]. The face is a very important part of a person to be recognized, each person has a different facial pattern even in identical twins [11][12]. Therefore, the face is the most accurate indicator for face detection [13][14]. On the machine, human face detection is done by distinguishing the pattern of face shape from other patterns on the camera frame [15][16]. The machine will group numerical and symbolic data on facial features in the digital image automatically to obtain descriptive data on the pattern of a face [17]. The condition that must be found in this face detector is that the face is detected or not detected. In face detection, various methods can be used and have been tested.

Facial landmarks point research, automates the identification of the location of facial points on facial images [18][19][20]. These points are dominant points that

describe the unique location of facial components, for example, the corners of the eyes, nose, or interpolation points that connect dominant points around facial components and facial contours [21][22][23]. Then, research [24][25][26] using the Fisherface method is to increase the ratio of the distance (scatter) between classes to the intra-class distance from the feature vector which is the basis of the LDA method, and reduce the dimensions obtained from PCA calculations [27][28][29]. The greater the ratio between classes, the resulting feature vector is less sensitive to changes in expression and changes in lighting, resulting in a better classification [30][31]. In the Local Binary Pattern Histogram [32][33][34] research, this algorithm will change the texture of a face image into a binary value, and this value represents part of the pixels of a face that forms a circle and has a center as a reference to the values [35]. The distance between the densities of these binary values is called neighbors. In the study of face recognition weightless neural networks (WNNs) [36][37], the use of ram in processing data is an architectural structure in itself in artificial intelligence, where ram itself is a collection of several memory cells. The structure corresponds to the direct use of memory cells [38][39]. However, the face image processing process still uses conventional computer machines, so the test results are still sourced from abundant resources.

Image processing with embedded techniques is a new technique [40][41][42][43]. This technique was developed for systems that have limited resources (hardware and software). Image processing devices in embedded systems are used in real-time systems [44][45]. The available resources are also very limited so the algorithm design and the use of computational image processing must be more efficient. In this study, we propose a performance testing technique for several facial pattern recognition methods, using an embedded platform. This test will compare the performance of each method. So that it can be known in detail, the effect of lighting, test distance, pattern recognition accuracy, and execution time.

2. MATERIAL AND METHODS

This section will discuss the stages of the research in detail. This stage aims so that the research does not go out of the corridor that has been made. The research stage is divided into two main stages, namely the hardware design stage and the system algorithm

2.1 EMBEDDED PLATFORM DESIGN

At the hardware design stage, it aims to determine the devices used in the computing process, starting from the input process to the data validation process. Hardware devices are devices that support an embedded platform because the system must be mobile, real-time, and has a fast execution process. The hardware used in this study is an ARM3 processor with an RPi 3 model B+ processor board. The memory capacity used is 1GB RAM. Then, the system uses two dynamical 360 servo motors. Figure 1 is a schematic of the system hardware.

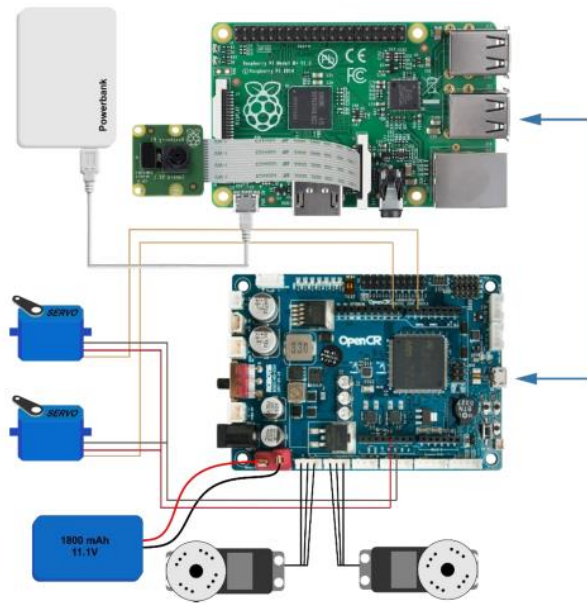


FIGURE 1. Diagram schematic of the hardware

The mobile device has a height of ± 1 meter, divided into 4 levels. The fourth level is used for pi camera components and 2 servo motors, which are installed in the mounted tilt camera. This section serves to move the camera horizontally and vertically. On the third level is the position of the Raspberry pi3 processor. In the second level, it is used as a power position. Then at the lowest level, there are components in the form of a microcontroller, motor driver, 2 dc motors, and 1 battery. Servo is used to find the best facial position. The first servo is used to move vertically and the second servo is used to move horizontally. They also implanted a camera PiCamera V2.1, with a resolution of 8 megapixels. Figure 2 is a block diagram of the system hardware.

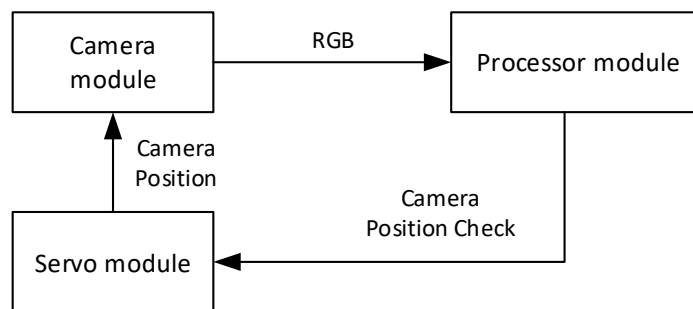


FIGURE 2. Hardware architecture design

The camera module is used to capture environmental images. This image is in RGB format, which is sent to the processor module. The processor module processes the image using a face recognition algorithm. If a face is detected, it will be marked. For the image processing process to be more optimal, the position of the face must be in

**Ahmad Zarkasi, Siti Nurmaini, Deris Setiawan,
Bhakti Yudho Suprpto, Huda Ubaya, Rizki Kurniati**
**Performance Comparison of Feature Face Detection Algorithm
on The Embedded Platform**

the right position. Therefore, it is necessary to check the position of the face first. Figure 3 is an example of checking the position of the user's face.

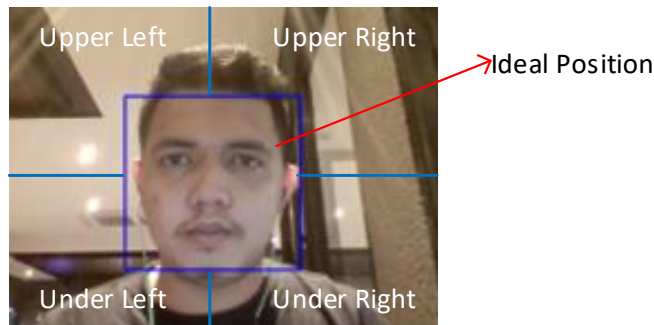


FIGURE 3. Sample ideal face position

There are five positions for image capture that can be processed in the face recognition algorithm, namely upper left, upper right, under left, under right, and ideal position. The five positions are the best positions obtained by the camera. If the image does not exist or is outside the five positions, the servo will certainly move until a complete face image is obtained. So that the face image can be processed further

2.2 LIGHTING MECHANISM DESIGN

The lighting mechanism design aims to measure the performance of face detection methods at different lighting levels in real-time. The device used to measure the level of lighting is a mobile phone that has a light sensor with Light meter software. Figure 4 below is a device for measuring light intensity.



FIGURE 4. Light mater tool

The light measurement mechanism is to uses several LED lights placed in a room. The position of the measuring device is right under the lamp. There are 4 lamps, which are arranged in a square. The measurement distance between the lights and

the mobile departing is about 2 meters. The lights will be turned on one by one, then the intensity of the light is measured. Measurements start from turning on 1 lamp up to 4 lights. Figure 5 is a scheme for measuring light intensity.

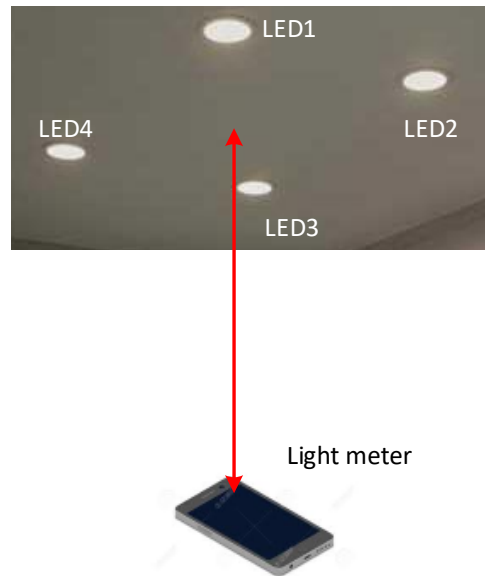


FIGURE 5. Light intensity measurement schematic

2.3 FACE DETECTION ALGORITHMS

This section aims to design a testing algorithm for several face detection methods. Each method will have many parameters that greatly affect the results obtained. Such as the speed of the detection process, accuracy, and level of accuracy of the test object. However, in this research, a test scheme will be designed with different light intensity levels. So that the general test data will be obtained about the effect of lighting on several test methods. Figure 6 below is a schematic design of the system test.

**Ahmad Zarkasi, Siti Nurmaini, Deris Setiawan,
Bhakti Yudho Suprpto, Huda Ubaya, Rizki Kurniati**
**Performance Comparison of Feature Face Detection Algorithm
on The Embedded Platform**

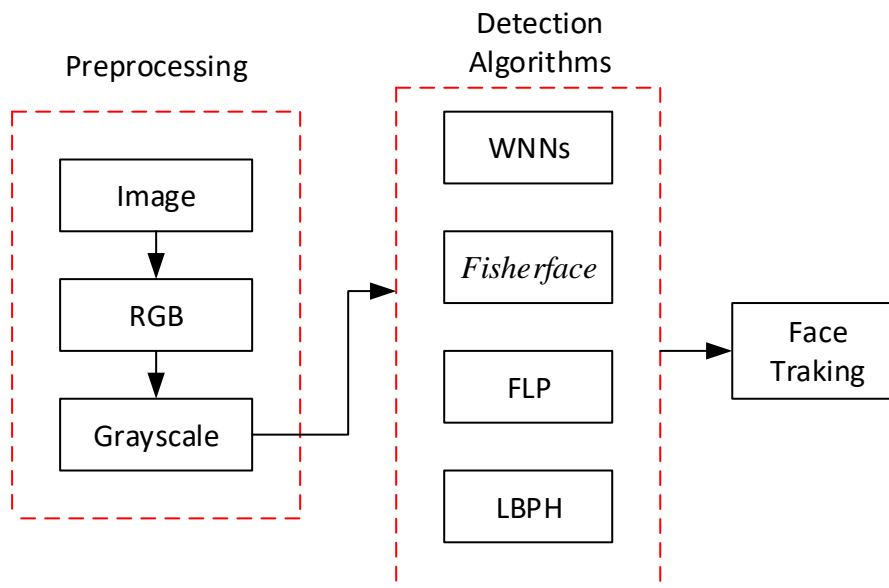


FIGURE 6. Testing diagram schematic

In Figure 6, the design of system testing is carried out in 3 stages. The first is the preprocessing stage. This stage is the first step of the detection process, which is divided into three stages. Image is the original data of the system environment, which is displayed in the camera. All environmental data is displayed in full in red (R), green (G), and blue (B) data formats. Then proceed with converting the RGB data into a grayscale image. Grayscale will change the RGB image from 24 bits format to 8 bits data format. The second stage is the detection algorithm.

This stage will process the data from the previous stage into several algorithms, namely Weightless Neural Networks (WNNs), Figerface, Facial Landmarks Point (FLP), and Local Binary Patterns Histograms (LBPH). Face detection consists of the process of detecting and recognizing facial patterns. Each method has a different process according to the modeling. In the results of this face detection stage, the face position will be obtained in (x, y), so that the servo will move to find the best position of the face.

The test scenario is as follows

- a. The lighting intensity is set to 3 states, namely LOW, MEDIUM, and HIGH.
- b. When the camera captures environmental images continuously, at the same time, the preprocessing process is carried out. Preprocessing is changing the color model from RGB image to grayscale.
- c. The results of this preprocessing, will be input to the face detection stage. Each input will be processed with different light intensity. At this stage, it will be seen how far the effect of light on the sensitivity of the detection results of each method.
- d. The servo movement will be seen as finding the center point of the face, which will predict the best face position.

3. RESULTS AND DISCUSSION

In this paper, we will analyze hardware and software with embedded platforms. All the tests were carried out on a raspberry pi minicomputer, which had been placed on a mobile robot. Meanwhile, the light intensity was measured using a mobile phone device.

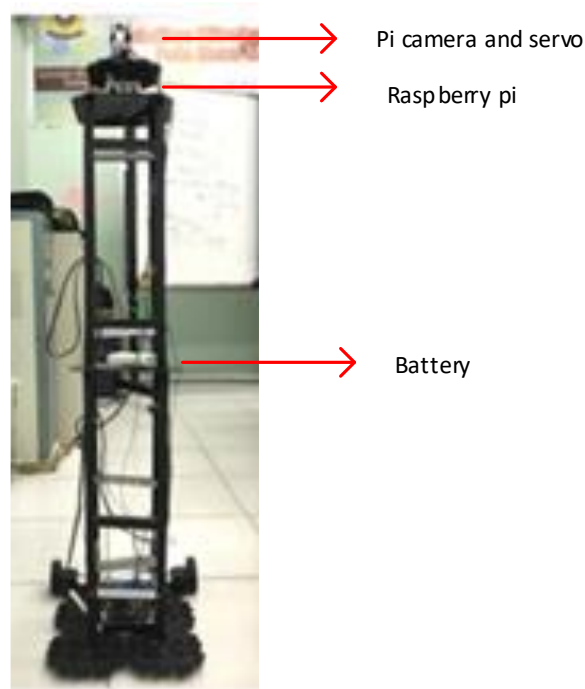


FIGURE 7. Hardware platform

Servo motor rotation analysis will be greatly influenced by the results of face detection. Therefore, the rotation settings and the direction of rotation must match the position of the face capture. In Figure 7, the servo is placed directly under the camera and the raspberry is placed directly under the servo. This allows the data transfer processing process to work quickly. The following is the servo motor setting data.

The direction of rotation of the servo is driven horizontally (H) and vertically (V). This causes the servo to move up, down, right and left based on the position of the face on the camera frame. The servo movement is determined based on the calculation of the face position (x,y). The calculation of the change in Pulse Width is adjusted to the maximum Pulse Width and minimum Pulse Width on the servo. The servo uses a maximum duty cycle of 2500 to 180 degrees clockwise, and a minimum duty cycle of 500 to 0 degrees counterclockwise. the results of testing the servo motor control can be seen in Table 1.

**Ahmad Zarkasi, Siti Nurmaini, Deris Setiawan,
Bhakti Yudho Suprpto, Huda Ubaya, Rizki Kurniati**
**Performance Comparison of Feature Face Detection Algorithm
on The Embedded Platform**

TABLE 1.
Servo Control

Face Position		Servo Pulse	
x	y	H	V
236	134	1600	1550
287	136	1600	1540
271	105	1500	1540
270	107	1500	1550
190	141	1500	1540
140	140	1600	1500
166	63	1600	1540
162	35	1600	1440
165	25	1600	1350
180	125	1600	1350

The next test is to measure the light intensity in real. Figure 8 is the result of taking light intensity. As previously explained, there are three divisions of light intensity areas, namely LOW, MEDIUM, and HIGH. The LOW area is in the range between 166 lx to 200 lx. Then for the MEDIUM area, it is in the range of 202 lx to 280 lx. As for the HIGH area, it is in the range of 300 lx to 400 lx. The values above may change according to the intervention, position, and distance between the test objects.

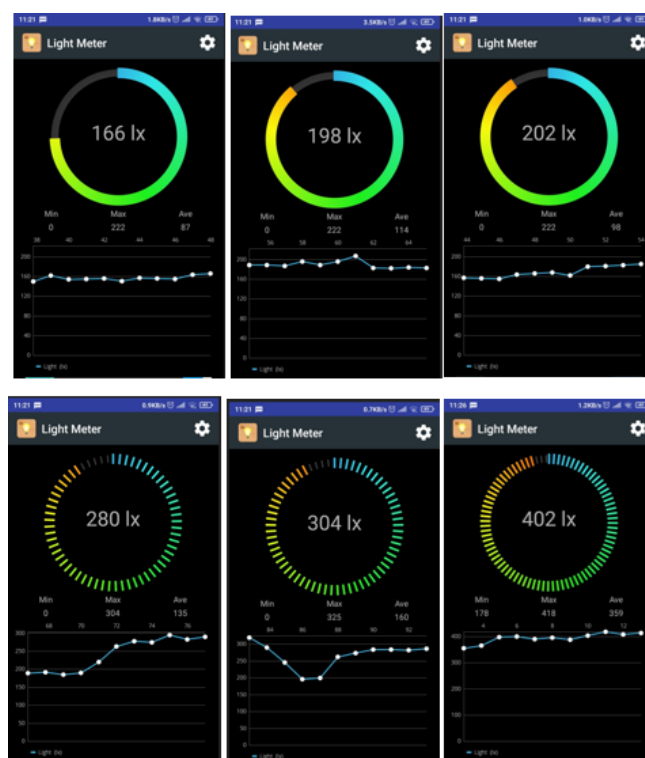


FIGURE 8. Light intensity measurement

In the case of lighting, changes in the data value of a pixel may occur. These changes can be significant or low. So it is necessary to analyze the effect of lighting on the accuracy value obtained. Table 2, is the general result of testing the effect of light intensity on the level of image accuracy from various test methods.

TABLE 2.
Accuracy Testing

Light Intensity	Algorithm				Accuracy level
	<i>Fisherface</i>	FLP	LBPH	WNNs	
400 lx	95%	95%	98%	98%	HIGH
300 lx	85%	90%	95%	95%	HIGH
200 lx	80%	85%	90%	90%	MEDIUM
175 lx	75%	75%	83%	80%	MEDIUM
>150 lx	30%	30%	40%	40%	LOW

In the case of lighting, changes in the data value of a pixel may occur. These changes can be significant or low. So it is necessary to analyze the effect of lighting on the accuracy value obtained. Table 2, is the general result of testing the effect of light intensity on the level of image processing methods that will be the object of testing are the Fisherface, FLP, LBPH, and WNNs methods. The amount of light intensity that becomes the test data, starting from the highest, is 400 lx, 300 lx, 200 lx, 175 lx, and > 150 lx. For HIGH lighting intensity between 400 lx to 300 lx, the Fisherface method has a low accuracy value of about 85%, while the LBPH and WNNs methods are 98%.

Likewise, testing at the MEDIUM lighting level, with a range of 200 lx to 175 lx, the Fisherface, and FLP methods have an accuracy rate of 75%, while the LBPH method has an accuracy rate of 80%. The next test is testing the intensity of the face at a certain distance. This test aims to obtain the level of accuracy of facial resemblance from the various methods being compared. The technique used is to do a direct test of the face image at a certain distance, with a fixed light intensity of 400 lx. Table 3 is the result of testing the level of accuracy at a certain distance. Image accuracy from various test methods.

In the table 3, the image processing method still uses the same method as the previous test. The distance of the face object in a row starting from the lowest is 50 cm, 75 cm, 100 cm, 150 cm, and data 151 cm. For testing distances of 50 cm to 75 cm, the value of all methods has a high accuracy value of 100% and the lowest is 90% in the FLP method.

**Ahmad Zarkasi, Siti Nurmaini, Deris Setiawan,
Bhakti Yudho Suprpto, Huda Ubaya, Rizki Kurniati**
**Performance Comparison of Feature Face Detection Algorithm
on The Embedded Platform**

TABLE 3.
Accuracy against distance at an intensity of 400 lx

Object distance	Algorithm				Accuracy level
	<i>Fisherface</i>	FLP	LBPH	WNNs	
50 cm	100%	100%	100%	100%	HIGH
75 cm	95%	90%	92%	96%	HIGH
100 cm	75%	70%	73%	75%	MEDIUM
150 cm	65%	55%	60%	60%	LOW
>150 cm	23%	30%	30%	25%	LOW

Then, at a test distance of 100 cm, the highest accuracy value was 75% in the Fisherface and WNNs methods, while the lowest accuracy value was 70% in the FLP method. For the test distance of 150 cm, the Fisherface method has the highest accuracy rate of 65%, while the FLP method has the lowest accuracy value of 55%. For the test distance above 151 cm, the results obtained are in a bad category. However, these results may change according to different test equipment and testing techniques.

3. CONCLUSION

In this paper, hardware and software analysis has been carried out according to the embedded platform. The test was carried out on a raspberry pi mini-computer, which had been placed on a mobile robot. Meanwhile, the light intensity was measured using a mobile phone device. The position of the face is detected at the point of the largest pixel $x = 287$, $y = 136$, and the lowest pixel point is at $x = 162$ and $y = 25$. The results of the test for the highest accuracy level of lx are at 400 lx, with the highest accuracy produced in the LBPH and WNNs methods, which is 89%. While at lx 175, the highest accuracy value was produced by LBPH, which was 83%

ACKNOWLEDGEMENTS

This research was funded by Universitas Sriwijaya for funding the research on "Penelitian Sains Teknologi dan Seni 2021". The 1st author would like to thank the Faculty of Computer Science, The Faculty of Engineering, Intelligence System Research Group, and COMNETS Research Group, and also Embedded Systems, Control System, and Robotic Laboratory Universitas Sriwijaya for supporting this research.

REFERENCES

- [1] J. Barbosa, W. K. Seo, and J. Kang, “ParaFaceTest: An ensemble of regression tree-based facial features extraction for efficient facial paralysis classification,” *BMC Med. Imaging*, vol. 19, no. 1, p. 30, Apr. 2019, doi: 10.1186/s12880-019-0330-8.
- [2] V. Maruthapillai and M. Murugappan, “Optimal Geometrical Set for Automated Marker Placement to Virtualized Real-Time Facial Emotions,” *PLoS One*, vol. 11, no. 2, p. e0149003, Feb. 2016, doi: 10.1371/journal.pone.0149003
- [3] R. Wang, “Original and Mirror Face Images and Minimum Squared Error Classification for Visible Light Face Recognition,” *Sci. World J.*, vol. 2015, 2015, doi: 10.1155/2015/842084.
- [4] A. Zarkasi *et al.*, “Face Movement Detection Using Template Matching,” in *Proceedings of 2018 International Conference on Electrical Engineering and Computer Science, ICECOS 2018*, 2019, pp. 333–338, doi: 10.1109/ICECOS.2018.8605215.
- [5] D. Jia, J. Cao, W. D. Song, X. L. Tang, and H. Zhu, “Colour FAST (CFAST) match: Fast affine template matching for colour images,” *Electron. Lett.*, vol. 52, no. 14, pp. 1220–1221, Jul. 2016, doi: 10.1049/el.2016.1331.
- [6] Z. Liu, Y. Guo, Z. Feng, and S. Zhang, “Improved Rectangle Template Matching Based Feature Point Matching Algorithm,” in *Proceedings of the 31st Chinese Control and Decision Conference, CCDC 2019*, 2019, pp. 2275–2280, doi: 10.1109/CCDC.2019.8833208.
- [7] H. Liu and L. Wang, “Gesture recognition for human-robot collaboration: A review,” *Int. J. Ind. Ergon.*, vol. 68, pp. 355–367, Nov. 2018, doi: 10.1016/j.ergon.2017.02.004.
- [8] D. Jia, J. Cao, W. D. Song, X. L. Tang, and H. Zhu, “Colour FAST (CFAST) match: Fast affine template matching for colour images,” *Electron. Lett.*, vol. 52, no. 14, pp. 1220–1221, Jul. 2016, doi: 10.1049/el.2016.1331.
- [9] P. Tome, R. Vera-Rodriguez, J. Fierrez, and J. Ortega-Garcia, “Facial soft biometric features for forensic face recognition,” *Forensic Sci. Int.*, vol. 257, pp. 271–284, Dec. 2015, doi: 10.1016/j.forsciint.2015.09.002.
- [10] M. Nordt and S. Weigelt, “Face recognition is similarly affected by viewpoint in school-aged children and adults,” *PeerJ*, vol. 2017, no. 5, p. e3253, May 2017, doi: 10.7717/peerj.3253.
- [11] C. Bi *et al.*, “Supervised filter learning for representation based face recognition,” *PLoS One*, vol. 11, no. 7, p. e0159084, Jul. 2016, doi: 10.1371/journal.pone.0159084.
- [12] L. Sun, X. Liang, and Q. Zhao, “Recursive templates segmentation and exemplars matching for human parsing,” *Comput. J.*, vol. 57, no. 3, pp. 364–377, Mar. 2014, doi: 10.1093/comjnl/bxt100.
- [13] H. Hosoya and A. Hyvärinen, “A mixture of sparse coding models explaining properties of face neurons related to holistic and parts-based processing,” *PLoS Comput. Biol.*, vol. 13, no. 7, p. e1005667, Jul. 2017, doi: 10.1371/journal.pcbi.1005667
- [14] C. Peng, W. Bu, J. Xiao, K. Wong, and M. Yang, “An Improved Neural Network Cascade for Face Detection in Large Scene Surveillance,” *Appl. Sci.*,

**Ahmad Zarkasi, Siti Nurmaini, Deris Setiawan,
Bhakti Yudho Suprpto, Huda Ubaya, Rizki Kurniati**
**Performance Comparison of Feature Face Detection Algorithm
on The Embedded Platform**

- vol. 8, no. 11, p. 2222, Nov. 2018, doi: 10.3390/app8112222.
- [15] J. Li and F. Yang, "Research on multi-robot scheduling algorithms based on machine vision," *Eurasip J. Image Video Process.*, vol. 2018, no. 1, 2018, doi: 10.1186/s13640-018-0355-x.
- [16] T. M. Murphy, R. Broussard, R. Schultz, R. Rakvic, and H. Ngo, "Face detection with a Viola-Jones based hybrid network," *IET Biometrics*, vol. 6, no. 3, pp. 200–210, May 2017, doi: 10.1049/iet-bmt.2016.0037.
- [17] M. Kasar, B. Vidyapeeth, M. M. Kasar, D. Bhattacharyya, and T.-H. Kim, "Face Recognition Using Neural Network: A Review Analysis of Multi-View Face Images and Recognition Through ANN View project Face Recognition Using Neural Network: A Review," *Artic. Int. J. Secur. its Appl.*, vol. 10, no. 3, pp. 81–100, 2016, doi: 10.14257/ijisia.2016.10.3.08.
- [18] M. Cummaudo *et al.*, "Pitfalls at the root of facial assessment on photographs: A quantitative study of accuracy in positioning facial landmarks," *Int. J. Legal Med.*, vol. 127, no. 3, pp. 699–706, 2013, doi: 10.1007/s00414-013-0850-7.
- [19] R. H, "Real Time Driver Drowsiness Detection System using OpenCV," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. VI, pp. 3254–3260, 2021, doi: 10.22214/ijraset.2021.35811.
- [20] D. Chen, Q. Chen, J. Wu, X. Yu, and T. Jia, "Face swapping: Realistic image synthesis based on facial landmarks alignment," *Math. Probl. Eng.*, vol. 2019, pp. 1–12, 2019, doi: 10.1155/2019/8902701
- [21] F. Fuentes-Hurtado, J. A. Diego-Mas, V. Naranjo, and M. Alcañiz, "Automatic classification of human facial features based on their appearance," *PLoS One*, vol. 14, no. 1, p. e0211314, Jan. 2019, doi: 10.1371/journal.pone.0211314.
- [22] H. Kim, H. W. Kim, and E. Hwang, "Real-time shape tracking of facial landmarks," *Multimed. Tools Appl.*, vol. 79, no. 23–24, pp. 15945–15963, 2020, doi: 10.1007/s11042-018-6814-7.
- [23] H. A. Shehu, W. Browne, and H. Eisenbarth, "An Adversarial Attacks Resistance-based Approach to Emotion Recognition from Images using Facial Landmarks," *29th IEEE Int. Conf. Robot Hum. Interact. Commun. ROMAN 2020*, pp. 1307–1314, 2020, doi: 10.1109/ROMAN47096.2020.9223510.
- [24] Z. Zhang, D. Lian, and S. Gao, "RGB-D-based gaze point estimation via multi-column CNNs and facial landmarks global optimization," *Vis. Comput.*, vol. 37, no. 7, pp. 1731–1741, 2021, doi: 10.1007/s00371-020-01934-1.
- [25] M. Anggo and La Arapu, "Face Recognition Using Fisherface Method," *J. Phys. Conf. Ser.*, vol. 1028, no. 1, 2018, doi: 10.1088/1742-6596/1028/1/012119.
- [26] X. Y. Jing, H. S. Wong, and D. Zhang, "Face recognition based on 2D Fisherface approach," *Pattern Recognit.*, vol. 39, no. 4, pp. 707–710, 2006, doi: 10.1016/j.patcog.2005.10.020.
- [27] SCAD Institute of Technology and Institute of Electrical and Electronics Engineers, "'A Study of LBPH, Eigenface, Fisherface and Haar-like features for Face recognition using OpenCV' ICISS-2019: proceedings of the International Conference on Intelligent Sustainable Systems (ICISS 2019): 21-22, February 2019," *2019 Int. Conf. Intell. Sustain. Syst.*, pp. 219–224, 2019.

- [28] C. Y. Low, A. B. J. Teoh, and C. J. Ng, “Multi-Fold Gabor, PCA, and ICA Filter Convolution Descriptor for Face Recognition,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 1, pp. 115–129, Jan. 2019, doi: 10.1109/TCSVT.2017.2761829
- [29] A. Ranftl, F. Alonso-Fernandez, S. Karlsson, and J. Bigun, “Real-time AdaBoost cascade face tracker based on likelihood map and optical flow,” *IET Biometrics*, vol. 6, no. 6, pp. 468–477, Nov. 2017, doi: 10.1049/iet-bmt.2016.0202.
- [30] M. M. Ahsan, Y. Li, J. Zhang, M. T. Ahad, and K. D. Gupta, “Evaluating the Performance of Eigenface, Fisherface, and Local Binary Pattern Histogram-Based Facial Recognition Methods under Various Weather Conditions,” *Technologies*, vol. 9, no. 2, p. 31, 2021, doi: 10.3390/technologies9020031.
- [31] B. W. Yohanes, R. Diaz Airlangga, and I. Setyawan, “Real Time Face Recognition Comparison Using Fisherfaces and Local Binary Pattern,” *Proc. - 2018 4th Int. Conf. Sci. Technol. ICST 2018*, vol. 1, pp. 1–5, 2018, doi: 10.1109/ICSTC.2018.8528608.
- [32] M. Kalakech, A. Porebski, N. Vandenbroucke, and D. Hamad, “Unsupervised local binary pattern histogram selection scores for color texture classification,” *J. Imaging*, vol. 4, no. 10, 2018, doi: 10.3390/jimaging4100112.
- [33] S. Karanwal and M. Diwakar, “OD-LBP: Orthogonal difference-local binary pattern for Face Recognition,” *Digit. Signal Process. A Rev. J.*, vol. 110, p. 102948, 2021, doi: 10.1016/j.dsp.2020.102948.
- [34] L. Li, X. Feng, Z. Xia, X. Jiang, and A. Hadid, “Face spoofing detection with local binary pattern network,” *J. Vis. Commun. Image Represent.*, vol. 54, pp. 182–192, 2018, doi: 10.1016/j.jvcir.2018.05.009.
- [35] Y. H. Chan, Y. Z. Zeng, H. C. Wu, M. C. Wu, and H. M. Sun, “Effective pneumothorax detection for chest X-ray images using local binary pattern and support vector machine,” *J. Healthc. Eng.*, vol. 2018, 2018, doi: 10.1155/2018/2908517.
- [36] M. De Gregorio and M. Giordano, “Background estimation by weightless neural networks,” *Pattern Recognit. Lett.*, vol. 96, pp. 55–65, Sep. 2017, doi: 10.1016/j.patrec.2017.05.029.
- [37] M. Berger, A. F. De Souza, J. de O. Neto, E. de Aguiar, and T. Oliveira-Santos, “Visual tracking with VG-RAM Weightless Neural Networks,” *Neurocomputing*, vol. 183, pp. 90–105, Mar. 2016, doi: 10.1016/j.neucom.2015.04.127.
- [38] D. O. Cardoso, F. França, and J. Gama, “A bounded neural network for open set recognition,” in *Proceedings of the International Joint Conference on Neural Networks*, 2015, vol. 2015-September, pp. 1–7, doi: 10.1109/IJCNN.2015.7280680.
- [39] D. O. Cardoso, J. Gama, and F. M. G. França, “Weightless neural networks for open set recognition,” *Mach. Learn.*, vol. 106, no. 9–10, pp. 1547–1567, Oct. 2017, doi: 10.1007/s10994-017-5646-4.
- [40] A. Kaczmarczyk and W. Zatorska, “Accelerating Image Fusion Algorithms Using CUDA on Embedded Industrial Platforms Dedicated to UAV and UGV,” in *Advances in Intelligent Systems and Computing*, 2020, vol. 920, pp. 697–706, doi: 10.1007/978-3-030-13273-6_65.

**Ahmad Zarkasi, Siti Nurmaini, Deris Setiawan,
Bhakti Yudho Suprpto, Huda Ubaya, Rizki Kurniati**
**Performance Comparison of Feature Face Detection Algorithm
on The Embedded Platform**

- [41] Z. Shi, H. Yu, and L. Kong, "Image stitching algorithm based on embedded system," in *Proceedings of 2018 IEEE International Conference on Mechatronics and Automation, ICMA 2018*, 2018, pp. 1174–1178, doi: 10.1109/ICMA.2018.8484530.
- [42] S. Nurmaini and A. Zarkasi, "Simple Pyramid RAM-Based Neural Network Architecture for Localization of Swarm Robots," *Artic. J. Inf. Process. Syst.*, 2015, doi: 10.3745/JIPS.01.0008.
- [43] C. Qin, W. Zhang, F. Cao, X. Zhang, and C. C. Chang, "Separable reversible data hiding in encrypted images via adaptive embedding strategy with block selection," *Signal Processing*, vol. 153, pp. 109–122, Dec. 2018, doi: 10.1016/j.sigpro.2018.07.008.
- [44] G. Fumera and F. Roli, "Support vector machines with embedded reject option," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2002, vol. 2388, pp. 68–82, doi: 10.1007/3-540-45665-1_6.
- [45] S. Nurmaini, A. Zarkasi, D. Stiawan, B. Yudho Suprpto, S. Desy Siswanti, and H. Ubaya, "Robot movement controller based on dynamic facial pattern recognition," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, p. 733, 2021, doi: 10.11591/ijeecs.v22.i2.pp733-743.