IMPLEMENTATION OF FACE RECOGNITION USING GEOMETRIC FEATURES EXTRACTION

[a] Risanuri Hidayat, [b] Muhammad Oka Bagus Wibowo, [c] Brama Yoga Satria, [d] Anggun Winursito

[a,b,c] Department of Electrical Engineering and Information Technology, Faculty of Engineering, Universitas Gadjah Mada, Yogyakarta, 55281, Indonesia
[d] Department of Electronics and Informatics Engineering Education, Faculty of Engineering, Yogyakarta State University, Yogyakarta, 55281, Indonesia
E-mail: risanuri@ugm.ac.id, mokabw@gmail.com, brama.y.s@mail.ugm.ac.id, anggunwinursito@uny.ac.id

Abstract

The face is among the biometric objects used to recognize one's identity. There are various face recognition system methods that can be applied, one of which is geometric features-based face recognition. Geometric features are unique features extraction of one's facial components. These features are obtained by calculating the comparison values of the distance measurement between facial components served as a reference like eyes, nose, and mouth. This research implemented a face recognition system using the geometric features method on a significantly low-spec computer system. This implementation was carried out by building a system, installing it on a computer system, and then testing it using laptops or computer devices and the camera web. The face recognition system would process the facial input images, extract their geometric features, and match the results with the data stored in the database. The research results were a low-spec computer system that could recognize its users by providing real-time feedback in the form of users' names with an accuracy of 98%.

Key words: biometrics, euclidean distance, face recognition, feature extraction, real-time.

INTRODUCTION

Technological proliferation in the world is rapidly progressing. The emergence of advanced televisions, smartphones, and computers is a shred of evidence that technology is constantly evolving. The development of hardware in those electronic and telecommunication devices certainly goes together with their software development. These advances in hardware and software have led to increasingly sophisticated smart devices. The features available on those devices have provided ease of use to store the users' personal, work-related, or other activities data. One thing constantly developed along with the smart devices' enhancement is their usage security, as most users demand their devices be accessible only for themselves or specific people. In other words, these devices will function only after recognizing their users and cannot be accessed by unauthorized users. Therefore, it deems necessary to provide a system that can recognize users by identifying and verifying who is accessing these devices. If the identification and verification results indicate that the users are not authorized, these devices will not work.

The recognition and identification can be conducted using various methods, one of which is by comparing biometric objects such as fingerprints and eye irises [1]. The most common method employed is by comparing faces or known as face recognition. In brief, face recognition means identifying the entire person's face by comparing their faces with the stored data. Over the last few years, research on face recognition has been conducted, so it is expected to provide an efficient system with low error rates [2].

The face is the easiest human body used to identify someone. It has been widely utilized as the computer systems' authentication object that requires user authentication, such as commercial or government applications [3], [4], [5]. This computer system uses the face recognition system to identify the users' faces and distinguish them from other users. Consequently, this system can serve as a good authentication process for a computer system regarding its accuracy.

The face recognition system can use a webcam in its implementation. As an additional device, a webcam is no longer an extraordinary device, given that almost all the newest smartphone models are equipped with a front camera that can be used as a webcam; the same goes with a laptop or PC webcam. A webcam works in real-time, so what is captured by this device will be processed simultaneously [6]. In addition, systems can capture a person's facial image for further detection and identification by using a webcam.

Face detection is the initial stage in the face recognition system. When the system detects face availability, it will be developed to recognize it. Face recognition has numerous methods used to detect and recognize faces. The use of these different method combinations will affect the results in terms of systems' performance and accuracy. Among those face recognition methods is the Viola-Jones method. This method is known for its simple feature usage that enables higher processing speed than other methods [7]. It also has become the most prevalent method due to its ease of implementation. When the Viola-Jones method has been implemented on a computer system, the subsequent stage is implementing a face recognition system to recognize the face owner's identity. Among other methods, the most prominent face recognition method is the geometric feature extraction method.

The geometric feature extraction method is a face recognition system that extracts facial features like eyes, nose, and mouth [8]. The extraction in question compares the distance between the facial features used as a reference for face recognition. Unlike the hybrid method that needs the entire face as the reference, the geometric feature extraction method has been proven to have a lighter computational load. Moreover, this method has been proven essential for finding accurate facial features due to its more lightweight mathematical calculations to process. Since the hybrid method requires the entire face to be processed, the computational load and face recognition process period will also increase significantly along with the number of facial components that must be processed.

One feature-based method example can be found in research [9], namely the geometric feature extraction. In this research, the facial image separator system was enhanced into facial components and was then extracted into the eyes, nose, mouth features, and the facial boundaries in a single still image taken from

the faces' front view. The distance between facial components and other features combination would form a semantic face model. The test results showed that the uniqueness of the facial feature distance required at least five facial features. These results also suggested that determining the facial features and the distance from the facial components would generate a semantic face model used as an identification reference for the face recognition system. Even though the geometric feature is among the methods that can be considered in face recognition system development, face recognition system implementation using geometric feature extraction has never been tested on a low-spec computer system. This method has been proven in the performance test containing mathematical calculation [10], yet no face recognition system on a low-spec computer system has applied this algorithm. In its implementation, previous research has employed this method on the face recognition system for students' attendance applications [11], [12], [13].

This study implemented a face recognition system with a geometric feature extraction method on a low-spec computer system. The expected result is a low-spec computer system that can recognize users through a web camera connected to a computer in real-time.
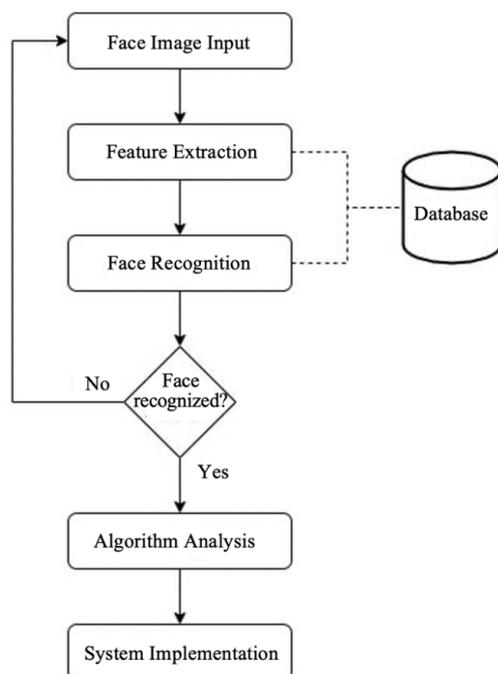
MATERIAL AND METHODS

Fig 1. Procedure flowchart

The research preparation procedure was carried out following the flowchart in Figure 1. The initial step in the research design procedure was inputting the face images via the available webcam connected to a computer system designed for the face recognition system prototype with the geometric feature extraction method. The face recognition system could be employed as the testing prototype when the system has met the research purpose, namely recognizing the users' faces. The face images would be taken as testing samples of this program.

Feature Extractions

According to the feature extraction method, the face recognition system can be divided into three [14]: holistic matching, feature-based, and hybrid. Obtained face images would be processed by the face recognition system using OpenCV. In this system, OpenCV serves as a library program providing various supporting features for the face recognition system. Meanwhile, the main algorithm employed in the face recognition system prototype was the geometric feature extraction method [15]. This method functions to extract facial image features to gain the Euclidean distance value served as input face comparison value of the users' faces stored in the database. The extracted features were the primary results of the geometric feature extraction method. Since these features' values can be used as the calculation reference, they will be used in the face recognition process.

Face Recognition

The features generated from the geometric feature extraction were subsequently inputted to the Euclidean distance value calculation obtained from the geometric feature extraction method's advanced process. Feature values were stored and recognized as a person's face in the database used during the running of the prototype. The K prototype captured facial features via a webcam connected to a computer. The Euclidean distance comparison (between the face and the stored feature) was within the predetermined tolerance threshold; the prototype could recognize the face and provide users' names as feedback. Based on the previous literature [16], facial classifications using the Euclidean distance has a predetermined threshold of 0,01. The prototype was tested before and following its installation in the operating system. Thus, it

could be immediately repaired when there was a change in the face recognition system's use and performance. When the prototype could recognize the users' faces, it could be installed in the operating system. In contrast, it should be repaired and the early procedure, namely inputting facial images, should be repeated if the prototype fails to recognize the users' faces.

Algorithm Analysis

A face recognition system generally utilizes a camera as an input device to capture face images used for the authentication process. Although face recognition system integrity is regarded as a biometric system, several issues are considered a major topic of ongoing research [17]. These issues include face-pose, facial expression, and illumination invariance. The face recognition system is analyzed based on its performance and result from several tests that follow previous research's reference. Therefore, in this research, those tests were conducted to strengthen the current results obtained from the previous research. The tests used in this research are defined below.

1. The accuracy test of face recognition system: This test was conducted by comparing the predicted results of the testing dataset with the original results of the training dataset used to train the system. The results were classified as True Positive, True Negative, False Positive, and False Negative to determine the system's accuracy.
2. The precision test of face recognition system: This test was conducted by providing input images with changes in facial expressions and facial tilt.
3. The effect of the lighting intensity test: This test was conducted by capturing images under different lighting conditions so that the changing variable was the light intensity. At the same time, facial expressions and positions were made the same in each test.
4. The computational load test of face recognition system: This test was conducted by increasing the number of training datasets in the database. It would affect the performance since matching the input images with the data in the database was carried out one by one.

These tests were selected as this study sought to determine the feasibility of the geometric feature extraction method for face recognition installed on a low-spec computer system. In this context, the low-spec computer system is a computer with RAM under 2GB. After the tests were carried out, research result data were recorded in a research report, and the system was installed in the computer system.

System Implementation

Face recognition implementation on the computer system was conducted by creating an executable file (.exe) that can be run stand-alone. This file was put together with other files run when the operating system was turned on. As a result, the face recognition and operating system could be accessed simultaneously. The file in question was only a shortcut to the main file containing the database in the same folder, allowing the face recognition system to access that database when running. The steps of the system used in this research are illustrated in Figure 2.

When the users facing the webcam connected to a computer that had previously been installed a geometric feature extraction method, the webcam would capture their facial images. Afterward, these images were further processed as the face recognition input using the geometric feature extraction method in the form of users' facial samples. Images captured by the webcam would also be displayed on the computer screen for the users to see in real-time. Since those images were updated in real-time, changes to the webcam captures would affect images displayed on the computer screen.
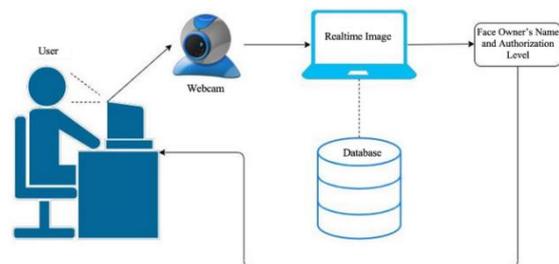

Fig 2. Illustration of the system as the research results

Facial samples would be taken every few seconds, then calculated based on the geometric feature extraction method calculation to obtain feature values used to

compare faces in the database. The values taken were matched with facial samples' feature values stored in the system database. After that, the Euclidean distance of the values matching in the form of facial samples' feature values used as input and the data stored in the database would be calculated simultaneously. They would be subsequently matched according to the specific tolerance threshold values due to the different facial expressions' distance change, even though they are significantly small. This tolerance threshold serves to provide room for the result change flexibility due to facial expression changes.

If the values matched the specified tolerance threshold of 0.01, the database would generate the users' names stored in the database. After a name was retrieved, the face recognition system with the geometric feature extraction method would display the face owners' names next to the real-time images shown on the computer screen. In this research, the face intended to recognize was limited to the users' faces with no comparisons of other faces. Hence, the name retrieval process would only display the users' names or 'unrecognized.'

RESULT AND DISCUSSION

Face recognition system requires input in the form of system users' faces and faces for system tests. The system users' faces function as reference data stored in the database accessed when the system is running. On the other hand, faces for the system test serve as testing data used in the face recognition performance testing. Face image input for the database was explicitly made for one expression, lighting, and same time so that the distance calculation bias could be avoided. The Viola-Jones algorithm was then applied to the face images captured by the webcam to detect a face's presence. The detected face was marked with a square or input images' Region of Interest (ROI). After the ROI was detected, face images were converted to grayscale to ease the subsequent processes. Afterward, the calculation reference points and the geometric patterns of grayscale face images' geometric patterns were determined. The geometric patterns were features with specific values stored in the database as the face recognition reference. There were nine reference points. Of those reference points, ten reference

features of the face recognition were generated, as shown in Figure 3.
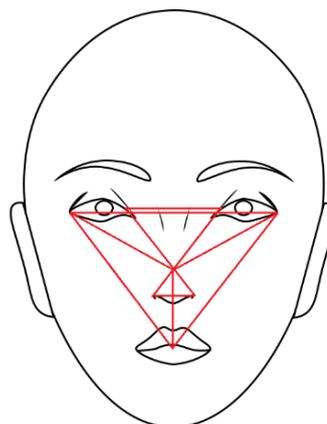
Fig 3. Geometric facial patterns for data calculation

System Accuracy Dataset Test

The dataset test is intended to test the system's accuracy in recognizing the users' faces used as input. Data used for the test was 50 Euclidean distance calculation data obtained from the facial feature extraction of computer users. The test was conducted under the same lighting and expression but at a different distance. Based on the test results, there were 49 True Positive data, 0 True Negative data, 1 False Positive data, and 0 False Negative data. As a result, the accuracy of the face recognition system using geometric feature extraction was about 98%. The calculation error might be due to changes in face position when the webcam captured the input images used for the Euclidean distance calculation. This system accuracy can be further enhanced by adding the number of facial data samples in the database.

The Effect of Lighting Intensity Test

The lighting intensity level test aims to determine the lighting intensity level needed for the face recognition system to run appropriately. This test was conducted because the face recognition system and lighting source are closely related. The face recognition system cannot function without lighting sources (pitch-black) unless an infrared camera is utilized to detect objects in the dark. The test conducted in this research applied various lighting conditions, as illustrated in Figure 4.

(a)   (b)   (c)
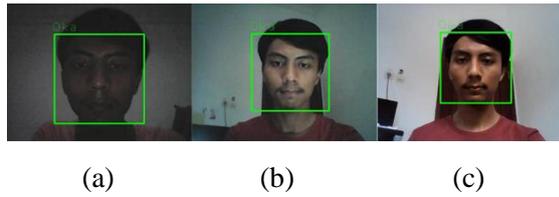
Fig 4. Test with different lighting: (a) dim lighting, (b) sufficient lighting, and (c) bright lighting.

Specifically, this test was conducted by changing the lighting intensity illuminating the system's users whose faces were recognized. Lighting intensity variations were categorized based on a range of values in the lux units: under 20 lux was categorized as dim light, between 20 to 50 lux was categorized as sufficient light, and between 100 to 200 lux was categorized as bright light. Table 1 shows he results of the lighting intensity test on the face recognition system.

Table 1. Results of Testing The Light Intensity Effect on The Face Recognition System

| No	Lighting Intensity	Lighting Category	Result
1	0	Dim	Face undetected, unrecognized
2	5	Dim	Face undetected, unrecognized
3	10	Dim	Face undetected, unrecognized
4	15	Dim	Face detected, recognized
5	20	Sufficient	Face detected, recognized
6	30	Sufficient	Face detected, recognized
7	40	Sufficient	Face detected, recognized
8	50	Sufficient	Face detected, recognized
9	100	Bright	Face detected, recognized
10	120	Bright	Face detected, recognized
11	150	Bright	Face detected, recognized
12	200	Bright	Face detected, recognized

This test suggests that the lighting level needed for the face recognition system with the geometric feature extraction method to work appropriately is at least 20 lux. In the 20-lux lighting, the system could consistently generate the users' names used for the system's input where the result consistency is the characteristic of an accurate system.

Computational Load Test

The computational load test seeks to prove the low CPU usage of the face recognition system when using the geometric feature extraction method. Face data in the database would be gradually duplicated to determine how heavy the computational load was in each stage. This test result is the CPU usage data represented in Figure 5.
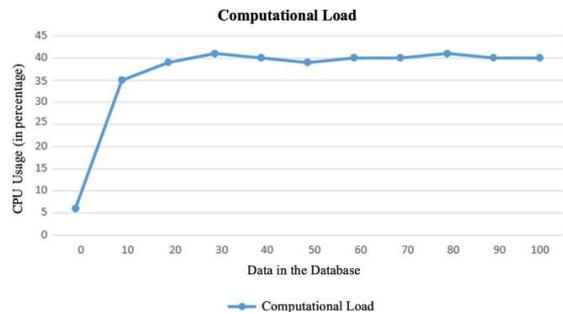
Fig 5. Result of computational load test

This test result indicated that the CPU usage of the face recognition system ranged from 35% to 41%, despite changes in the amount of data in the database. In addition, this test concludes that the data change in the database will not significantly change the computer systems' computational load, regardless of the face recognition system performance, which operates in real-time and periodically accesses data.

System Implementation

The face recognition system implementation will be slightly different for every operating system. It happens since different structures persist in those operating systems. In this research, the face recognition system was implemented in two operating systems, namely Microsoft Windows and Linux Ubuntu.

In Microsoft Windows, the implementation of the face recognition system using the geometric feature extraction method was effortlessly carried out since the interface, and shortcut features made it easier for the implementation. However, the implementation

duration took approximately 20 to 30 minutes since creating an executable file required a long time. In the standby or idle position, this system took up 124 MB of memory. Meanwhile, the face recognition system took up 167 MB of memory in its active position.

Constraints were encountered during the implementation of the face recognition system in Linux Ubuntu and occurred due to this operating systems' unique method. At the same time, the implementation duration took less than 10 minutes because, with high computational speed, the executable file creation process was completed in no time. The face recognition system took up 124 MB of memory in standby or idle position, while it required 167 MB of memory in the active position. In this regard, the face recognition system's performance in the Linux Ubuntu operating system is not much different from that of Microsoft Windows. The implementation of the face recognition system in the Linux Ubuntu operating system is shown in Figure 6.

Fig 6. Implementation of face recognition system on Linux Ubuntu operating system.

CONCLUSION

The face recognition system has been successfully implemented and has run well on a low-spec computer system. It can recognize the users' faces through input taken by a webcam with a high level of accuracy, that is 98%. In addition, the system's accuracy can be improved to minimize errors during the users' face recognition by employing ten features out of ten extractable facial features.

The precision test provides a more significant number of users' facial samples stored in the program database, eventually increasing the accuracy level of the face recognition affected by differences in expression. Meanwhile, the computational load test indicates that the amount of data in the database does not affect the computational load borne by the computer system's CPU installed with a facial recognition system with a geometric feature extraction method.

In the face recognition process, the face tilt and room lighting affect the system performance when there is input through a webcam. Based on the system precision test, the maximum face tilt tolerated by the facial recognition system is approximately 10.39°. At the same time, based on the lighting difference test, the room lighting required by the facial recognition system to run properly is at least 20 lux.

REFERENCES

[1] S. Rajib and B. Debotosh, "Memory Efficient Human Face Recognition Using Fiducial Points," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, no. 1, 2012.

[2] J. George and T. B., "Face Recognition Technique for Authentication in Smart Devices – Comparative Study," International Journal of Applied Information Systems (IJAIS), vol. 12, no. 1, 2017.

[3] W. Zhao, R. Chellappa, P. J. Phillips, and A. A. Rosenfield, "Face Recognition: A Literature Survey," ACM Computing Surveys, vol. 35, no.4, pp. 399-458, 2003.

[4] Santoso, Hadi. *Pengenalan Banyak Wajah di Dalam Ruang Kelas Secara Realtime Menggunakan Modifikasi Adaboost Viola-Jones dan Fitur 2D DWT-PCA dengan Struktur Indeks KD Tree*. 2018. PhD Thesis. Universitas Gadjah Mada.

[5] A. E. Widjaja, H. Hery, and D. Habsara Hareva, "The Office Room Security System Using Face Recognition Based on Viola-Jones Algorithm and RBFN", *intensif*, vol. 5, no. 1, pp. 1-12, Feb. 2021.

[6] J. A. Andre, "Sistem Security Webcam Dengan Menggunakan Microsoft Visual Basic (6.0)," Jurnal Teknologi dan Sistem Informasi UNIVRAB, 2016.

[7] D. A. Prasetya and I. Nurviyanto, "Deteksi Wajah Metode Viola Jones Pada OpenCV Menggunakan Pemrograman Python," Simp. Nas. RAPI XI FT UMS, pp. 18-23, 2013.

[8] Kholistianingsih, "Ekstraksi Ciri Geometris Untuk Pengenalan Wajah," Fakultas Teknik Universitas Gadjah Mada, Yogyakarta, 2012.

[9] B.S. Hantono and R. Hidayat, "Ekstraksi Ciri Geometris pada Pengenalan Wajah," Fakultas Teknik Universitas Gadjah Mada, Yogyakarta, 2010.

[10] T. E. Prabowo, R. Hartanto and S. Wibirama, "Pengembangan Purwarupa Aplikasi Presensi Mahasiswa Dteti UgmBerbasis Pengenalan Wajah Menggunakan Algoritme Eigenface," Departemen Teknik Elektro dan Teknologi Informasi, Yogyakarta, 2019.

[11] A. L. Machidon, O. M. Machidon and P. L. Ogrutan, "Face Recognition Using Eigenfaces, Geometrical PCA Approximation and Neural Networks," *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, pp. 80-83, 2019.

[12] Y. Yuliana and I. Nurhaida, "Rancang Bangun Aplikasi Pengenalan Wajah menggunakan Metode Viola-Jones dan Algoritma PCA," J. Telekomun. dan Komput., 2018.

[13] S. Nagamalla and BC. Dhara, "A Novel Face Recognition Method using Facial Landmarks," Seventh International Conference on Advance in Pattern Recognition, pp. 445-448, 2019.

[14] D. Y. Liliana, M. R. Widyanto and T. Basaruddin, "Geometric Facial Components Feature Extraction for Facial Expression Recognition," *2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pp. 391-396, 2018.

[15] A. H. Mulyadi, L. Talangko and A. Lawi, "Pengenalan Citra Wajah Menggunakan Metode Analisis Komponen Utama Dan Jarak Euclidean," Fakultas MIPA, Universitas Hasanudin, Makassar, 2018.

[16] M. O. B. Wibowo, "Implementasi Pengenalan Wajah dengan Ciri Geometris pada Sistem Komputer," Fakultas Teknik Universitas Gadjah Mada, Yogyakarta, 2020.

[17] S. Z. Li and A. K. Jain, "Handbook of Face Recognition," Springer, 2011.

.