

# Haar Cascade dan Algoritma Eigenface Untuk Sistem Pembuka Pintu Otomatis

<sup>1</sup>Elfridus Hasman, <sup>2\*</sup>Indah Purwitasari Ihsan, <sup>3</sup>Herianto S Pailing, <sup>4</sup>Safaruddin

<sup>1,2,3,4</sup>Universitas Fajar Makassar, Indonesia

[elfridhasman@gmail.com](mailto:elfridhasman@gmail.com); [indah.ihsan13@gmail.com](mailto:indah.ihsan13@gmail.com); [heripailing@gmail.com](mailto:heripailing@gmail.com); [safaruddin@gmail.com](mailto:safaruddin@gmail.com)

## Article Info

### Article history:

Received, 04-06-2021

Revised, 09-06-2021

Accepted, 18-06-2021

### Kata Kunci:

Algoritma Eigenface  
Haar Cascade  
Sistem Pembuka Pintu  
Pengolahan Citra  
Otomatis

### Keywords:

Eigenface Algorithm  
Haar Cascade  
Door Opening System  
Image Processing  
Automatic

## ABSTRAK

Teknologi diciptakan untuk mempermudah manusia dalam melakukan segala pekerjaan dan aktifitasnya, termasuk dalam hal mengakses pintu. Menggunakan teknologi pengolahan citra, wajah merupakan salah satu alternatif yang bisa digunakan untuk mengakses pintu dan mengamankannya dari orang yang tidak bertanggung jawab. Hal ini dikarenakan wajah setiap manusia memiliki pola yang berbeda-beda yang bisa ditransformasikan menjadi citra digital dan diolah menggunakan algoritma pengolahan citra. Dalam penelitian ini, mengkombinasikan *haar cascade* dan *algoritma eigenface* untuk mengolah citra wajah. Hasil dari pengolahan citra tersebut digunakan untuk menentukan hak akses dalam mengakses pintu, untuk kemudian diintegrasikan ke mikrokontroler, sehingga pintu dapat terbuka otomatis. Penelitian ini menghasilkan *prototype system* pembuka pintu otomatis dengan pengenalan wajah sebagai penentu hak aksesnya. Dari hasil penelitian, algoritma *eigenface* tidak dapat bekerja pada pencahayaan 0 lux hingga 8 lux dalam jarak 20 cm hingga 60 cm yaitu menghasilkan akurasi 0%, sedangkan pada pencahayaan 36 lux sampai 44 lux dan 160 lux sampai 172 lux algoritma *eigenface* bekerja dengan baik dengan jarak pengambilan gambar 20-60 cm dengan akurasi 80%.

## ABSTRACT

Technology was created to make it easier for humans to do all their work and activities, including accessing doors. Using image processing technology, faces are an alternative that can be used to access doors and secure them from irresponsible people. This is because the face of every human being has a different pattern that can be transformed into a digital image and processed using an image processing algorithm. In this research, combining haar cascade and eigenface algorithm to processing face images. The results of the image processing are used to determine access rights in accessing the door, and then integrated into the microcontroller, so that the door can be opened automatically. This research produces a prototype automatic door opening system with face recognition as a determinant of access rights. From the results of the study, the eigenface algorithm cannot work at 0 lux to 8 lux lighting within a distance of 20 cm to 60 cm which produces 0% accuracy, while at 36 lux to 44 lux and 160 lux to 172 lux lighting the eigenface algorithm works well with a shooting distance of 20 cm to 60 cm with 80% accuracy.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-nc-nd/4.0/) license.



## Penulis Korespondensi:

Indah Purwitasari Ihsan,  
Program Studi Teknik Elektro,  
Universitas Fajar,  
Email: indah.ihsan13@gmail.com

## 1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi pada saat ini membawa banyak perubahan dalam kehidupan sehari-hari. Banyak pekerjaan yang sebelumnya dilakukan secara konvensional, sekarang dilakukan dengan bantuan mesin. Demikian juga dengan kemajuan ilmu dan teknologi dalam pengolahan citra memungkinkan untuk memanfaatkan teknologi tersebut di segala bidang.

Pengolahan citra (*image processing*) adalah teknik mengolah citra yang mentransformasikan citra masukan menjadi citra lain agar keluaran memiliki kualitas yang lebih baik dibandingkan kualitas citra masukan[1]. Citra yang digunakan adalah citra digital, yaitu gambar dua dimensi yang dihasilkan dari analog dua dimensi yang kontinu menjadi gambar melalui proses sampling, citra digital merupakan citra yang dapat diolah komputer [2]. Salah satu yang termasuk dalam pengolahan citra adalah pengenalan wajah (*face recognition*). *Face recognition* merupakan salah satu teknik pengenalan wajah yang sama seperti sidik jari dan retina mata, dimana hasil tangkapan kamera akan dicocokkan dengan foto dan lekuk wajah yang sudah ada di dalam database[3]. Terdapat beberapa algoritma yang bisa digunakan untuk pengenalan wajah yaitu algoritma eigenface, fisherfaces, artificial neural network, local binary pattern, local gaussian structural pattern dan masih banyak algoritma lainnya[4]. Pada penelitian ini menggabungkan *haar cascade* dan algoritma *eigenface*. Ide dari *haar cascade* adalah mengenali obyek berdasarkan nilai sederhana dari fitur tetapi bukan merupakan nilai piksel dari image obyek tersebut[5]. Algoritma *eigenface* adalah suatu metode pengenalan wajah yang berdasarkan pada algoritma Principal Component Analysis (PCA). Secara singkat prosesnya adalah citra direpresentasikan dalam sebuah gabungan vector yang dijadikan satu matriks tunggal. Dari matriks tunggal ini akan diekstraksi suatu ciri utama yang akan membedakan antara citra wajah satu dengan citra wajah lainnya[6].

Pengolahan citra dalam hal *face recognition* merupakan salah satu alternatif yang bisa digunakan untuk sistem keamanan dan otomatisasi. Hal ini dikarenakan wajah setiap orang memiliki pola yang berbeda-beda yang bisa ditransformasikan menjadi citra digital dalam bentuk *matriks – matriks* yang bisa diolah menjadi sesuatu yang lebih berguna. Pada saat ini, masih banyak ruangan/rumah/kantor yang masih menggunakan kunci manual untuk membuka pintunya. Kunci manual memiliki beberapa kekurangan diantaranya mudah hilang dan gampang diduplikasi sehingga bisa terjadi pencurian. Sehingga dibutuhkan sebuah sistem pembuka pintu otomatis yang memanfaatkan teknologi pengolahan citra.

Beberapa penelitian terkait diantaranya penelitian yang dilakukan oleh Widodo Muda Saputra, Helmie Arif Wibawa, S.Si, M.Cs, dan Nurdin Bahtiar, S.Si, M.T, 2013. Pengenalan Wajah Menggunakan Algoritma Eigenface Dan Euclidean Distance, Hasil pengujian sistem menunjukkan bahwa penggunaan Eigenface dengan Euclidean Distance sebagai classifier dapat memberikan tingkat akurasi yang cukup tinggi. Untuk gambar wajah yang diikutsertakan dalam pelatihan, dapat diperoleh 91% identifikasi benar[7]. Penelitian selanjutnya oleh Derian Indra Bramantio1, Erwin Susanto, S.T., M.T., Ph.D, dan Ramdhan Nugraha, SPd., M.T, 2016. Pada penelitian ini pengenalan wajah akan diimplementasikan pada pintu melalui motor servo, Aplikasi yang dibuat menggunakan bahasa pemrograman python dengan menggunakan Raspberry Pi. Sistem ini dirancang agar bisa mencari sendiri posisi dari wajah di dalam sebuah citra. Algoritma yang digunakan adalah Eigenface atau yang biasa disebut dengan Principle Component Analysis (PCA). Penggunaan citra input diharapkan bisa menjadi lebih fleksible, memiliki akurasi mencapai 90% sehingga hanya orang yang terdapat di database yang dapat masuk ke dalam rumah tersebut. Sedangkan motor servo dirancang agar dapat mengontrol posisi untuk membuka slot kuncinya[8]. Penelitian yang dilakukan oleh Ario Wicaksono, Erwin Susanto, S.T.,M.T., Ph.D. , dan Dr. Ing. Fiky Yosef Suratman, S.T.,M.T. , 2016. Pada penelitian ini dibuat sebuah alat, dimana sistem pintu akan terbuka dan tertutup secara otomatis berdasarkan karakteristik wajah. Alat ini menggunakan metode Jaringan Saraf Tiruan (JST) Backpropagation. Dengan metode tersebut informasi yang diperoleh dari WebCam pada program pengujian diolah oleh Raspberry Pi 2 Model B dan dibandingkan dengan database yang merupakan hasil dari proses program pembelajaran. Hasil dari pengolahan citra akan menjadi inputan ke Motor Servo, jika sesuai maka pintu akan terbuka dan tertutup kembali secara otomatis, namun sebaliknya apabila tidak sesuai maka pintu tidak akan terbuka. Masing-masing pengujian dilakukan sebanyak 25 kali[9].

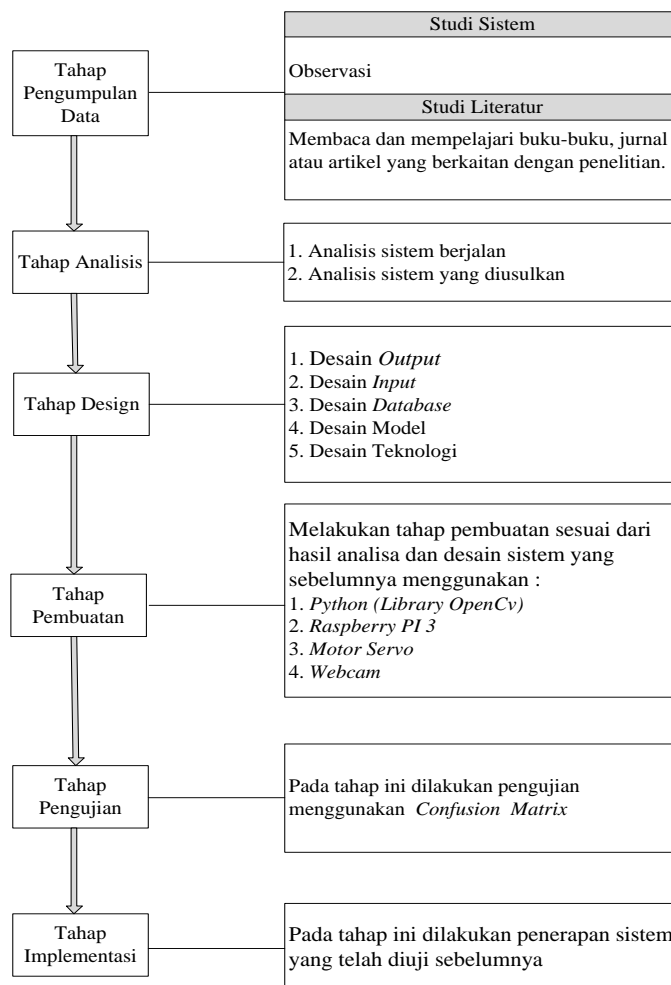
Pada penelitian ini dibuat sebuah prototype pembuka pintu otomatis menggunakan *haarcascade classifier* dan algoritma *Eigenface*. *Haar cascade* sebagai klasifier untuk memisahkan objek dan *background* sehingga yang nantinya ter-*capture* oleh sistem hanyalah wajah saja. Kemudian input berupa wajah tersebut akan diproses menggunakan algoritma *eigenface* dengan membandingkan data *training* dan data *testing* sehingga dapat disimpulkan mana wajah yang memiliki hak akses untuk

membuka pintu dan mana yang tidak. Alat yang digunakan yaitu Raspberry PI 3 sebagai komputer mini untuk menyimpan dan mengeksekusi program, motor servo sebagai penggerak dan webcam yang digunakan untuk mengcapture wajah. Data latih (data *training*) yang digunakan yaitu sebanyak 15 data yang sebelumnya sudah disimpan dalam database dan data uji (data *testing*) sebanyak 5 data. Output yang dihasilkan adalah apabila wajah dikenali, maka program akan menjalankan perintah menggerakkan motor servo untuk membuka pintu secara otomatis, namun apabila wajah tidak dikenali, pintu akan tetap tertutup.

**2. METODE PENELITIAN**

Metode yang digunakan dalam pengembangan sistem yaitu menggunakan metode *prototype* dimana proses pengerjaannya akan menghasilkan model fisik kerja sistem dan berfungsi sebagai versi awal dari sistem. Dengan metode prototyping ini akan dihasilkan prototype sistem sebagai perantara pengembang dan pengguna agar dapat berinteraksi dalam proses kegiatan pengembangan sistem informasi. Langkah-langkah dalam *prototyping* meliputi pengumpulan kebutuhan, proses desain yang cepat, membangun prototipe, evaluasi dan perbaikan[10].

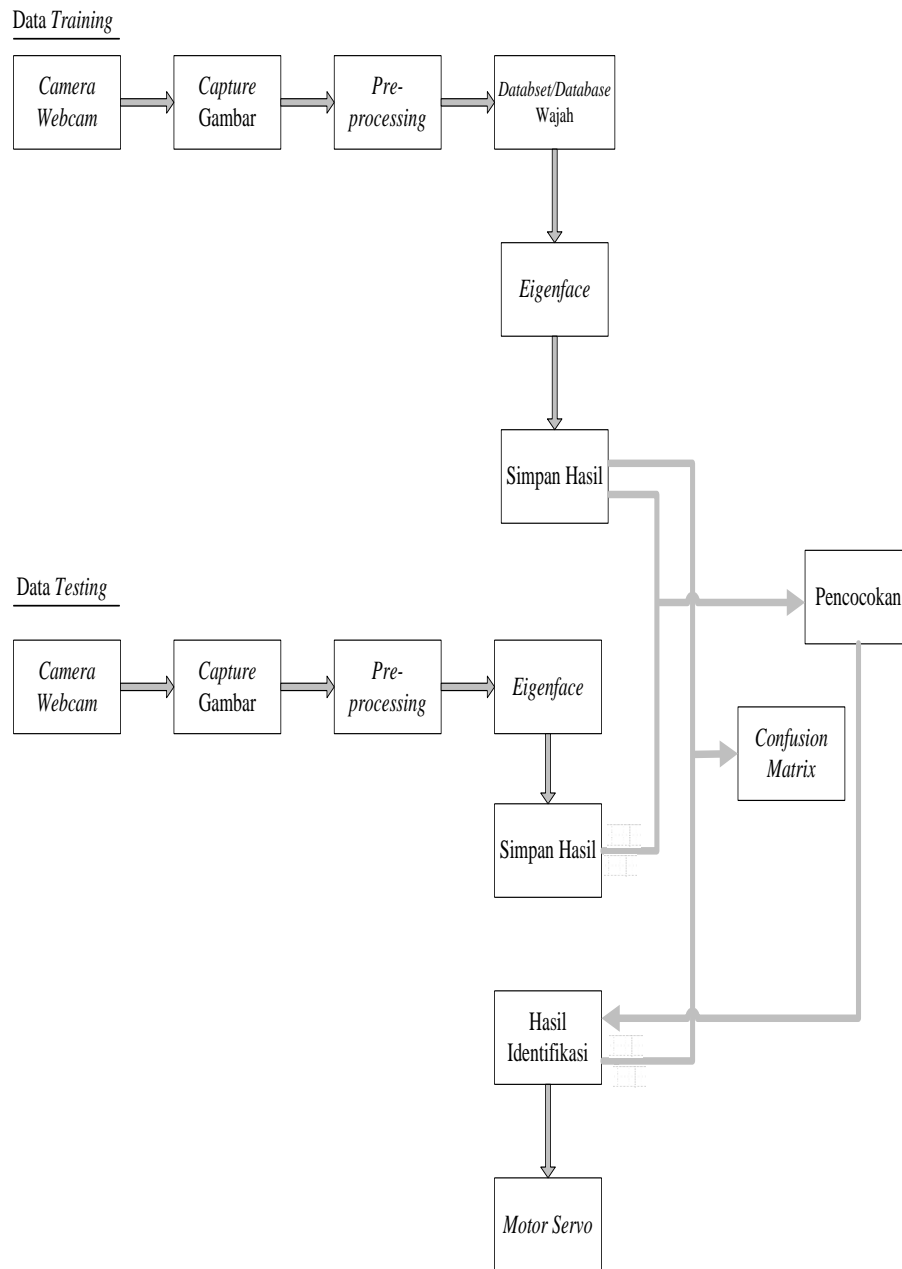
Alat dan bahan yang digunakan dalam penelitian ini yaitu, laptop Intel(R) Celeron(R) 2957U processor 1.40Ghz, raspberry PI 3, memori 512 MB, kamera webcam dengan resolusi 1.3 mp, motor servo, sistem operasi windows 10 Pro, kode editor sublime text 3, dan bahasa pemrograman phyton. Pengumpulan data dilakukan dengan cara kajian teori dan kajian pustaka sebagai bahan referensi, serta dilakukan pengumpulan data dengan cara observasi dan pengamatan secara langsung terhadap kinerja dari fungsionalitas sistem yang telah dibuat. Tahapan yang dilakukan pada penelitian ini digambarkan pada gambar 1 berikut :



Gambar 1. Bagan Tahapan Penelitian

**Bagan Kerja Sistem**

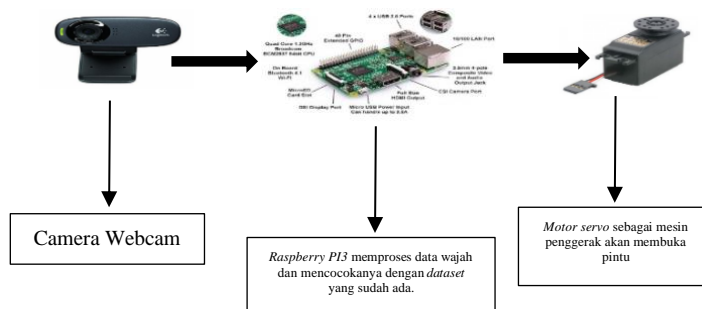
Bagan kerja sistem dapat dilihat pada gambar 2 berikut :



Gambar 2. Bagan Kerja Sistem (Dokumentasi Pribadi)

**Rancangan Rangkaian Alat**

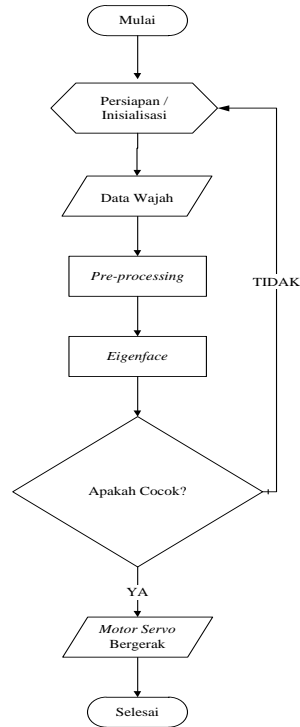
Rancangan rangkaian alat dapat dilihat pada gambar 3 berikut :



Gambar 3. Rancangan rangkaian alat (Dokumentasi Pribadi)

**Flowchart Sistem**

Flowchart sistem ditunjukkan pada gambar 4 berikut :

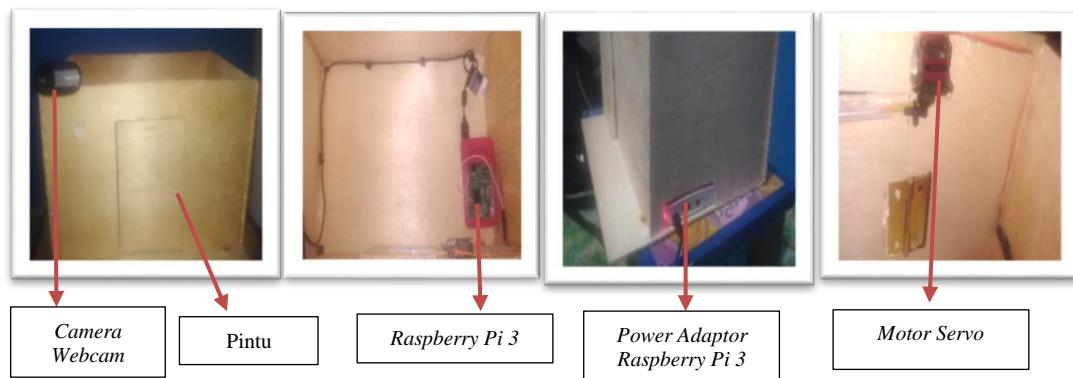


Gambar 4. Flowchart Sistem

**3. HASIL DAN ANALISIS**

**3.1 Prototipe Sistem**

Setelah membuat desain sistem, langkah selanjutnya adalah membuat prototype dari hasil desain tersebut. Prototipe merupakan implementasi awal dari rancangan sistem yang dibuat untuk mendemonstrasikan konsep-konsep dan percobaan[11]. Prototipe rangkaian alat pembuka pintu otomatis dengan menggunakan deteksi wajah dapat dilihat pada gambar 5 berikut:



Gambar 5. Prototipe Sistem (Dokumentasi Pribadi)

**3.2. Proses Haar Cascade**

*Harcascade Classifier* adalah fitur yang terdapat pada *openCv* yang digunakan untuk mendeteksi wajah dengan cara memisahkan antara objek dan *background*. Sistem akan otomatis melakukan capture wajah dengan menggunakan fitur *Cascade Classifier*. Apabila wajah sudah dicapture, maka

akan langsung tersimpan di folder dataset. Dataset merupakan keseluruhan data yang akan diolah meliputi didalamnya data *training* dan data *testing*[12]. Data training digunakan untuk melatih algoritma, sedangkan data testing digunakan untuk mengetahui performa algoritma yang telah dilatih sebelumnya ketika menemukan data baru yang belum pernah dilihat[13].



Gambar 6. hasil *capture* wajah menggunakan *cascade classifier* (a), data set hasil *capture* wajah (b)

### 3.3. Perhitungan Algoritma *Eignface*

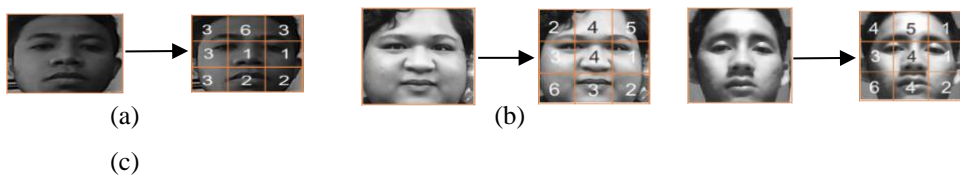
Proses pengenalan wajah dilakukan menggunakan algoritma *eignface* untuk menentukan hak akses. Pada proses pengenalan wajah, data training yang sudah disimpan dipanggil untuk mengenali data uji. Apabila wajah dikenali maka pintu akan terbuka otomatis, sebaliknya apabila wajah tidak dikenali maka pintu tetap tertutup.



Gambar 7 Wajah dikenali (a), Wajah tidak dikenali (b)

Berikut merupakan perhitungan dari algoritma *eignface* :

Proses pengenalan wajah dengan *algoritma eigenface* dilakukan dengan beberapa tahapan. Proses pengujian pada tahap ini menggunakan *matrixs* 3 x 3 dan 3 data *training* sebagai contoh. Dari data *training image* (*T*) tersebut, tahap selanjutnya yang dilakukan adalah menyusun seluruh data *training* menjadi satu *matrixs* tunggal.



Gambar 8. *Training image* 1 (a), *training image* 2 (b), *training image* 3 (c)

Representasikan semua *matrixs training* menjadi *matrixs* linier  $N \times 1$  (*flatvector*).

$$T1 = \begin{bmatrix} 3 & 6 & 3 \\ 3 & 1 & 1 \\ 3 & 2 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 3 \\ 6 \\ 3 \\ 3 \\ 1 \\ 1 \\ 3 \\ 2 \\ 2 \end{bmatrix} \quad T2 = \begin{bmatrix} 2 & 4 & 5 \\ 3 & 4 & 1 \\ 6 & 3 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 2 \\ 4 \\ 5 \\ 3 \\ 4 \\ 1 \\ 6 \\ 3 \\ 2 \end{bmatrix} \quad T3 = \begin{bmatrix} 4 & 5 & 1 \\ 3 & 4 & 1 \\ 6 & 4 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 4 \\ 5 \\ 1 \\ 3 \\ 4 \\ 1 \\ 6 \\ 4 \\ 2 \end{bmatrix}$$

a. Menghitung nilai tengah / *Mean* ( $\Psi$ )

$$\Psi = \frac{1}{3} \left( \begin{bmatrix} 3 \\ 6 \\ 3 \\ 3 \\ 1 \\ 1 \\ 3 \\ 2 \\ 2 \end{bmatrix} + \begin{bmatrix} 2 \\ 4 \\ 5 \\ 3 \\ 4 \\ 1 \\ 6 \\ 3 \\ 2 \end{bmatrix} + \begin{bmatrix} 4 \\ 5 \\ 1 \\ 3 \\ 4 \\ 1 \\ 6 \\ 4 \\ 2 \end{bmatrix} \right) = \begin{bmatrix} 3 \\ 5 \\ 3 \\ 3 \\ 3 \\ 1 \\ 5 \\ 3 \\ 2 \end{bmatrix}$$

- b. Menghitung nilai selisih antara *training image* ( $\Gamma$ ) dengan nilai *mean* ( $\Psi$ )

$$\emptyset 1 = \begin{bmatrix} 3 \\ 6 \\ 3 \\ 3 \\ 1 \\ 1 \\ 3 \\ 2 \\ 2 \end{bmatrix} - \begin{bmatrix} 3 \\ 5 \\ 3 \\ 3 \\ 1 \\ 1 \\ 3 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ -2 \\ 0 \\ -2 \\ -1 \\ 0 \end{bmatrix} \quad \emptyset 2 = \begin{bmatrix} 4 \\ 5 \\ 3 \\ 3 \\ 1 \\ 6 \\ 3 \\ 3 \\ 2 \end{bmatrix} - \begin{bmatrix} 3 \\ 5 \\ 3 \\ 3 \\ 1 \\ 5 \\ 3 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \emptyset 3 = \begin{bmatrix} 4 \\ 1 \\ 3 \\ 1 \\ 6 \\ 4 \\ 3 \\ 3 \\ 2 \end{bmatrix} - \begin{bmatrix} 3 \\ 3 \\ 3 \\ 1 \\ 5 \\ 3 \\ 3 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

- c. Menghitung nilai matriks kovarian

$$L = \begin{bmatrix} 0 & 1 & 0 & 0 & -2 & 0 & -2 & -1 & 0 \\ -1 & -1 & 2 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & -2 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & -1 & 1 \\ 1 & -1 & 0 \\ 0 & 2 & -2 \\ 0 & 0 & 0 \\ -2 & 1 & 1 \\ 0 & 0 & 0 \\ -2 & 1 & 1 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 10 & -5 & -5 \\ -5 & 8 & -3 \\ -5 & -3 & 8 \end{bmatrix}$$

- d. Hitung nilai *eigen value* ( $\lambda$ ) dan *eigen vector* ( $V$ ) dari matriks kovarian

$$\lambda_i \rightarrow \det(A - \lambda I) = 0$$

dengan I merupakan matriks identitas

Maka nilai  $\lambda$  adalah :

$$\lambda_1 = 1,131, \lambda_2 = 32,378, \lambda_3 = -7,509$$

Nilai *eigen vector* diperoleh dengan cara mensubstitusi *eigen value* ke dalam persamaan  $(\lambda I - L) V = 0$

Untuk  $\lambda_1 = 1,131$  :

$$V = \begin{bmatrix} 1,384 \\ -0,018 \\ 1 \end{bmatrix}$$

Untuk  $\lambda_2 = 32,378$  :

$$V = \begin{bmatrix} 2,459 \\ 0,036 \\ 0,5 \end{bmatrix}$$

Untuk  $\lambda_3 = -7,509$  :

$$V = \begin{bmatrix} 2,276 \\ 0,083 \\ 0,75 \end{bmatrix}$$

$$\text{Maka } C = \begin{bmatrix} 1,384 & 2,459 & 2,276 \\ -0,018 & 0,036 & 0,083 \\ 1 & 0,5 & 0,75 \end{bmatrix}$$

- e. Menghitung nilai *eigenface* ( $\mu$ )

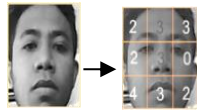
$\mu_1 = v \times \emptyset 1$ , maka

$$\mu_1 = \begin{bmatrix} -4,5520 & -5,8100 & 0 \\ -0,1660 & -0,1730 & 0 \\ -1,5000 & -0,7500 & 0 \end{bmatrix}$$

$$\mu_2 = \begin{bmatrix} 0,8920 & 1,7050 & 2,7680 \\ 0,1010 & 0,0540 & -0,0360 \\ -0,2500 & -0,5000 & 2 \end{bmatrix}$$

$$\mu_3 = \begin{bmatrix} 3,6600 & 4,7350 & -2,7680 \\ 0,0650 & 0,1190 & 0,0360 \\ 1,7500 & 1,2500 & -2 \end{bmatrix}$$

Proses pengenalan wajah dilakukan dengan pengujian terhadap data yang tersimpan di dataset. Untuk mengenali wajah, langkah yang dilakukan sama dengan proses pada gambar *training* mulai dari merubah matriks persegi menjadi *flatvector* hingga memperoleh nilai *eigenface* ( $\mu$ ).



Gambar 9. Testing image

$$\Gamma_{\text{baru}} = \begin{bmatrix} 2 & 3 & 3 \\ 2 & 3 & 0 \\ 4 & 3 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 2 \\ 3 \\ 3 \\ 2 \\ 3 \\ 0 \\ 4 \\ 3 \\ 2 \end{bmatrix}$$

Setelah merepresentasikan gambar uji ke *flatvector*, selanjutnya mencari selisih( $\phi$ ) antara gambar uji dengan nilai *mean* ( $\psi$ ).

$$\Gamma_{\text{baru}} = \begin{bmatrix} 2 \\ 3 \\ 3 \\ 2 \\ 3 \\ 0 \\ 4 \\ 3 \\ 2 \end{bmatrix} - \begin{bmatrix} 3 \\ 5 \\ 3 \\ 3 \\ 1 \\ 5 \\ 3 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} -1 \\ -2 \\ 0 \\ -1 \\ 0 \\ -1 \\ -1 \\ 0 \\ 0 \end{bmatrix}$$

Dari nilai selisih, maka nilai *eigenface* dapat dihitung  $\mu_{\text{baru}}$ , sehingga diperoleh :

$$\mu_{\text{baru}} \begin{bmatrix} -6,1190 & -2,7680 & -2,4590 \\ -0,1010 & 0,0360 & -0,0360 \\ -2,2500 & -2 & -0,5 \end{bmatrix}$$

Setelah nilai *eigenface* untuk gambar uji diperoleh maka langkah selanjutnya yaitu identifikasi dengan menentukan jarak terpendek dengan *eigenface* dari *eigenvector training image*, sehingga diperoleh :

$$\epsilon_1 = \sqrt{20,18014} = 4,492231$$

$$\epsilon_2 = \sqrt{109,0245} = 10,44148$$

$$\epsilon_3 = \sqrt{180,8715} = 13,44885.$$

Dari hasil perhitungan nilai jarak terpendek *Training Image 1*, *Training Image 2* dan *Training Image 3* terhadap gambar *testing*, maka nilai jarak *eigenface* yang terkecil diidentifikasi lebih mirip antara *Training Image 1* dengan gambar *testing*, dibandingkan dengan *Training Image 2* dan *Training Image 3*.

### 3.4. Hasil Pengujian

Pengujian sistem dilakukan pengujian terhadap 5 data uji yang sudah tersimpan didataset. Pengukuran pencahayaan yang dilakukan menggunakan aplikasi *Lux Light Meter Free*. Pencahayaan dinyatakan dalam satuan *lux*. *Lux* merupakan satuan turunan SI dari pencahayaan dan daya pancar cahaya[14]. Pengujian ini adalah pengujian yang dilakukan pada tingkat pencahayaan yang sama pada saat pengambilan data latih dan data uji, yaitu pada pencahayaan 0-8 *lux*, 36-44 *lux* dan 160-172 *lux* dari jarak 20 cm, 30 cm, 40 cm, 50 cm dan 60 cm dengan posisi wajah 45° ke kiri, 45° ke kanan dan 90°. Hasil pengujian dianalisa menggunakan *confussion matrix*. Confusion Matrix adalah sebuah metode yang biasa digunakan untuk perhitungan akurasi, recall, precision, dan error rate[15]. Berikut hasil analisa dari pengujian yang telah dilakukan :

Tabel 1. *confusion matrix* pada pencahayaan 0-8 *lux*

		Predicted Class	
		Positif	Negatif
Positif	0	0	75



Actual Class	Negatif	0	0
--------------	---------	---	---

$$\text{Akurasi} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) * 100\%$$

$$= \frac{0+0}{0+0+75+0} * 100\% = 0 * 100\% = 0\%$$

$$\text{Precision Positif} = (\text{TP} / (\text{TP} + \text{FP})) * 100\%$$

$$= \frac{0}{0+75} * 100\% = 0 * 100\% = 0\%$$

$$\text{Precision Negatif} = (\text{TN} / (\text{TN} + \text{FN})) * 100\%$$

$$= \frac{0}{0+0} * 100\% = 0 * 100\% = 0\%$$

Tabel 2. *confusion matrix* pada pencahayaan 36-44 lux

		Actual Class	
		Positif	Negatif
Predicted Class	Positif	60	15
	Negatif	0	0

$$\text{Akurasi} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) * 100\%$$

$$= \frac{60+0}{60+0+15+0} * 100\% = 0.8 * 100\% = 80\%$$

$$\text{Precision Positif} = (\text{TP} / (\text{TP} + \text{FP})) * 100\%$$

$$= \frac{60}{60+15} * 100\% = 0.8 * 100\% = 80\%$$

$$\text{Precision Negatif} = (\text{TN} / (\text{TN} + \text{FN})) * 100\%$$

$$= \frac{0}{0+0} * 100\% = 0 * 100\% = 0\%$$

Tabel 3. *confusion matrix* pada pencahayaan 160-172 lux

		Actual Class	
		Positif	Negatif
Predicted Class	Positif	60	15
	Negatif	0	0

$$\text{Akurasi} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) * 100\%$$

$$= \frac{60+0}{60+0+15+0} * 100\% = 0.8 * 100\% = 80\%$$

$$\text{Precision Positif} = (\text{TP} / (\text{TP} + \text{FP})) * 100\%$$

$$= \frac{60}{60+15} * 100\% = 0.8 * 100\% = 80\%$$

$$\text{Precision Negatif} = (\text{TN} / (\text{TN} + \text{FN})) * 100\%$$

$$= \frac{0}{0+0} * 100\% = 0 * 100\% = 0\%$$

Dari hasil pengujian akurasi menggunakan sebanyak 5 data wajah dengan pencahayaan 0-8 lux, jarak 20– 60 cm pada posisi 45<sup>0</sup> ke kiri, 45<sup>0</sup> ke kanan dan 90<sup>0</sup> didapatkan akurasi 0% dengan precision positif sebesar 0% dan precision negatif sebesar 0%. Pada pencahayaan 36-44 lux, jarak 20–60 cm pada posisi 45<sup>0</sup> ke kiri, 45<sup>0</sup> ke kanan dan 90<sup>0</sup> didapatkan akurasi 80% dengan precision positif sebesar 80% dan precision negatif sebesar 0%. Pada pencahayaan 160-172 lux, jarak 20-60 cm pada posisi 45<sup>0</sup> ke kiri, 45<sup>0</sup> ke kanan dan 90<sup>0</sup> didapatkan akurasi 80% dengan precision positif sebesar 80% dan precision negatif sebesar 0%. Sehingga dapat disimpulkan bahwa algoritma eigenface tidak dapat bekerja pada pencahayaan 0-8 lux pada jarak 20-60 cm dikarenakan kondisi pencahayaan gelap, sedangkan pada pencahayaan 36-44 lux dan 160-172 lux algoritma eigenface dapat bekerja dengan baik dengan jarak pengambilan gambar 20-60 cm.

#### 4. KESIMPULAN

Pada penelitian ini Haar cascade digunakan sebagai face detection untuk membedakan antara objek dengan background, sehingga yang ter-capture oleh kamera hanyalah wajah saja, sedangkan algoritma eigenface digunakan dalam proses face recognition untuk menentukan hak akses. Pengujian akurasi menggunakan confusion matrix dilakukan untuk menguji keakuratan dari sistem dalam mengenali wajah menggunakan algoritma eigenface. Berdasarkan penelitian yang telah dilakukan dapat disimpulkan bahwa pencahayaan sangat penting dalam proses pengenalan wajah. Pada

pencahayaan 36-44 *lux* mendapatkan tingkat akurasi 80% dengan precision positif sebesar 80% dan precision negatif sebesar 0%, pada pencahayaan 160-172 *lux* diperoleh akurasi 80% dengan precision positif sebesar 80% dan precision negatif sebesar 0%. Oleh karena itu, untuk melakukan pengambilan data latih dan data yang akan diuji harus dilakukan pada intensitas cahaya yang sama yaitu 36-44 *lux* dan 160-172 *lux* agar wajah dapat dikenali dengan baik. Sistem pembuka pintu otomatis dengan pengenalan wajah telah terintegrasi dengan baik, pintu otomatis terbuka apabila wajah dikenali dan akan tetap tertutup apabila wajah tidak dikenali. Proses pengenalan wajah membutuhkan berberapa data latih dimana semakin banyak data latih semakin tinggi tingkat akurasi pengenalan wajah tersebut.

## REFERENSI

- [1] A. R. Putri, "Pengolahan Citra Dengan Menggunakan Web Cam Pada Kendaraan Bergerak Di Jalan Raya," *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 1, no. 01, pp. 1–6, 2016, doi: 10.29100/jipi.v1i01.18.
- [2] N. Z. Munantri, H. Sofyan, and M. Y. Florestiyanto, "Aplikasi Pengolahan Citra Digital Untuk Identifikasi Umur Pohon," *Telematika*, vol. 16, no. 2, p. 97, 2020, doi: 10.31315/telematika.v16i2.3183.
- [3] Munawir, L. Fitria, and M. Hermasyah, "Implementasi Face Recognition pada Absensi Kehadiran Mahasiswa Menggunakan Metode Haar Cascade Classifier," *InfoTekJar J. Nas. Inform. dan Teknol. Jar.*, vol. 5, no. 1, pp. 40–43, 2020, [Online]. Available: <https://doi.org/10.30743/infotekjar.v5i1.1997>.
- [4] S. Sriyati, A. Setyanto, and E. E. Luthfi, "Literature Review: Pengenalan Wajah Menggunakan Algoritma Convolutional Neural Network," *J. Teknol. Inf. dan Komun.*, vol. 8, no. 2, 2020, doi: 10.30646/tikomsin.v8i2.463.
- [5] M. Syarif and Wijanarto, "Deteksi Kedipan Mata Dengan Haar Cascade Classifier Dan Contour Untuk Password Login," *Techno.com*, vol. 14, no. 4, pp. 242–249, 2015.
- [6] M. R. Muliawan, B. Irawan, and Y. Brianorman, "Implementasi Pengenalan Wajah Dengan Metode Eigenface Pada Sistem Absensi," *J. Coding, Sist. Komput. Untan*, vol. 03, no. 1, pp. 41–50, 2015, [Online]. Available: <http://jurnal.untan.ac.id/index.php/jcskommipa/article/viewFile/9727/9500>.
- [7] W. M. Saputra, H. A. Wibawa, and P. Wajah, "Pengenalan wajah menggunakan algoritma," vol. 2, no. 1, pp. 102–110, 2013.
- [8] D. I. Bramantio, E. Susanto, D. Ph, and R. Nugraha, "Perancangan Dan Implementasi Keamanan Pintu Berbasis Design and Implementation Security Door With Face," vol. 3, no. 1, pp. 79–83, 2016.
- [9] S. F. Y. Wicaksono Ari, Susanto Erwin, "Rancang Bangun Pintu Pintar Pada Ruang Kerja Dengan Mendeteksi Wajah Berbasis Pengolahan Citra," *e-Proceeding Eng.*, vol. 3, no. 3, p. 3, 2016.
- [10] D. Purnomo, "Model Prototyping Pada Pengembangan Sistem Informasi," *J I M P - J. Inform. Merdeka Pasuruan*, vol. 2, no. 2, pp. 54–61, 2017, doi: 10.37438/jimp.v2i2.67.
- [11] W. Nugraha and M. Syarif, "Penerapan Metode Prototype Dalam Perancangan Sistem Informasi Penghitungan Volume Dan Cost Penjualan Minuman Berbasis Website," *JUSIM (Jurnal Sist. Inf. Musirawas)*, vol. 3, no. 2, pp. 94–101, 2018, doi: 10.32767/jusim.v3i2.331.
- [12] R. Ferdiana, F. Jatmiko, D. D. Purwanti, A. S. T. Ayu, and W. F. Dicka, "Dataset Indonesia untuk Analisis Sentimen," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 8, no. 4, p. 334, 2019, doi: 10.22146/jnteti.v8i4.533.
- [13] Z. A. Fikriya, M. I. Irawan, and S. Soetrisno., "Implementasi Extreme Learning Machine untuk Pengenalan Objek Citra Digital," *J. Sains dan Seni ITS*, vol. 6, no. 1, 2017, doi: 10.12962/j23373520.v6i1.21754.
- [14] R. Rohadi and I. Yulianti, "Uji Efektifitas Pencahayaan Ruang Kuliah Menggunakan Software Calculux Indoor 4.12," *Unnes Phys. J.*, vol. 6, no. 1, pp. 50–53, 2017.
- [15] A.- Arini, L. K. Wardhani, and D.- Octaviano, "Perbandingan Seleksi Fitur Term Frequency & Tri-Gram Character Menggunakan Algoritma Naïve Bayes Classifier (Nbc) Pada Tweet Hashtag #2019gantipresiden," *Kilat*, vol. 9, no. 1, pp. 103–114, 2020, doi: 10.33322/kilat.v9i1.878.

