

# Rancang Bangun Sistem Biometrik Pengenalan Wajah Menggunakan *Principal Component Analysis*

Nicco<sup>1</sup>, Iman Fahruzi<sup>2</sup>

Jurusan Teknik Elektro, Politeknik Negeri Batam  
Parkway Street, Batam Centre, Batam 29461, Indonesia  
E-mail: lim.nicco@gmail.com, iman@polibatam.ac.id

## ABSTRAKS

Sistem keamanan biometrik dapat mengenali penggunanya lebih tepat dibandingkan sistem keamanan berbasis kata sandi. Biometrik memiliki karakteristik tidak mudah hilang, tidak dapat lupa, dan tidak mudah dipalsukan karena keberadaannya melekat pada manusia. Terdapat beberapa macam jenis keamanan dengan menggunakan teknologi biometrik diantaranya pengenalan sidik jari, retina mata, dan struktur wajah. Dalam penelitian ini, penulis mengembangkan pengenalan struktur wajah menjadi sistem keamanan pintu ruangan secara real-time. Penelitian ini menggunakan sebuah webcam untuk meng-capture citra wajah pengguna dan kemudian dibandingkan dengan wajah yang tersimpan di database. Secara umum terdapat 2 metode yang digunakan oleh penulis yaitu metode HaarCascade untuk proses deteksi wajah dan metode PCA atau Eigenface untuk proses pengenalan wajah. Pengujian dilakukan dengan menggunakan 150 data training dan 150 data uji. Pada sistem ini, digunakan parameter jarak sebesar 30 cm untuk mengukur tingkat keakuratannya. Hasil pengujian menunjukkan tingkat keberhasilan pengenalan secara keseluruhan sebesar 83.33%. Sistem ini dirancang untuk bekerja secara real-time dengan harapan mempermudah pengguna dan dapat meminimalisir tindakan kriminal kedepannya.

**Kata Kunci:** biometrik, keamanan, real-time, haarcascade, pca, eigenface

## ABSTRACT

Biometric security system can recognize its user more precisely than password-based security systems. The biometrics have characteristics like not easily lost, can't be forgotten, and not easily counterfeited because its existence inherent in human beings. There are several different types of security by using biometric technologies including fingerprint recognition, eye retina, and facial structure. In this study, the authors have developed a face recognition into real-time security system for the entrance. This research uses a webcam to capture the image of the user's face and then compared with the face that stored in the database. Generally, there are two methods used by the author which is HaarCascade method for face detection and PCA or Eigenface method for face recognition. Experiments were done using 150 training data and 150 test data. In this system, 30 cm were used as parameters of distance to measure the accuracy. The results showed overall recognition success rate of 83.33%. This system is designed to work in real-time with the hope to make it easier for the user and can minimize criminal act in the future.

**Keywords:** biometric, security, real-time, haarcascade, pca, eigenface

## 1. PENDAHULUAN

Dewasa ini, perkembangan teknologi semakin pesat. Banyak terobosan-terobosan baru yang diciptakan dengan basis teknologi canggih. Tujuannya tidak lain adalah untuk mempermudah dan mempercepat kerja manusia. Contoh nyata dapat dilihat pada kasus keamanan. Pada awalnya upaya

pemenuhan kebutuhan akan privasi, keamanan dan kepercayaan dalam penggunaan sistem *e-government* (penggunaan teknologi oleh pemerintah untuk mengoptimalkan proses kerja) di dunia berusaha dipenuhi dengan menggunakan nomor identitas pribadi yang bersifat rahasia yaitu PIN (*Personal Identity Number*) maupun berupa kartu yaitu *smart card* atau *tokens*.

Penggunaan PIN dan *smart card* dapat diakses siapa saja dan mudah dicuri atau disalahgunakan oleh orang lain seperti *hacker* (pengguna yang tidak bertanggung jawab). Hal ini terjadi dalam aplikasi pemalsuan identitas paspor, kartu identitas, dan penanda identitas lainnya. Setelahnya, dimanfaatkanlah teknologi biometrik sebagai sistem pengamanan baru untuk meningkatkan privasi, keamanan, dan kepercayaan penggunaannya. Teknologi biometrik menawarkan autentikasi secara biologis yang memungkinkan sistem dapat mengenali penggunaannya lebih tepat. Sistem ini sangat sulit ditembus dan dipalsukan, tidak seperti pengamanan berbasis kata sandi lain karena identifikasi dilakukan pada bagian tubuh pengguna yakni sidik jari, retina mata, dan struktur wajah. Bagian-bagian tersebut tidak bisa dihilangkan atau dicuri begitu saja.

Berpegang pada realita yang ada, penulis mengembangkan teknologi biometrik khususnya pengenalan wajah untuk dijadikan aplikasi keamanan pada pintu masuk ruangan. Dengan kata lain, penulis berharap sistem ini dapat menggantikan penggunaan PIN maupun *smart card*.

## 2. BAHAN DAN METODE

### 2.1 Biometrik

Apa itu biometrik ? Biometrik adalah kata yang berasal dari bahasa Yunani, yaitu *bios* yang berarti hidup dan *metron* yang berarti ukuran. Dalam artian ilmiah biometrik adalah suatu teknologi mengenai pengenalan makhluk hidup yang berbasis pada karakteristik individu yang unik. Pengembangan teknologi biometrik ini dilatari bahwa pada dasarnya setiap manusia memiliki sesuatu yang unik/khas. Keunikan tersebut tentu hanya dimiliki oleh dirinya sendiri.

Teknologi biometrik dikembangkan karena dapat memenuhi dua fungsi yaitu identifikasi dan verifikasi, disamping itu biometrik memiliki karakteristik seperti tidak mudah hilang, tidak dapat lupa, dan tidak mudah dipalsukan karena keberadaannya melekat pada manusia. Dimana satu manusia dengan yang lainnya tidak akan memiliki ciri khas yang pasti sama 100%, maka keunikannya akan lebih terjamin. Bagian-bagian dari tubuh manusia yang bersifat unik/spesifik diantaranya adalah sidik jari, retina mata, dan struktur wajah.

### 2.2 Pengolahan Citra

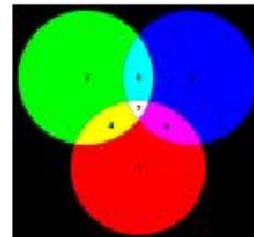
Pengolahan citra atau *image processing* adalah setiap bentuk pengolahan sinyal yang masukannya berupa gambar, sedangkan keluaran dari pengolahan tersebut dapat berupa gambar atau sejumlah karakteristik yang berkaitan dengan gambar. Singkatnya *image processing* adalah segala proses yang digunakan untuk mengolah suatu gambar.

### 2.3 Pengenalan Wajah

*Face recognition* atau pengenalan wajah adalah salah satu teknologi biometrik yang telah banyak diaplikasikan dalam identifikasi, sistem keamanan, dan sebagainya. Dalam aplikasinya, pengenalan wajah menggunakan sebuah kamera untuk menangkap wajah seseorang kemudian dibandingkan dengan wajah yang sebelumnya telah disimpan di dalam *database* tertentu. Ada banyak metode yang telah dikembangkan oleh para ilmuwan untuk dapat melakukan pengenalan wajah secara akurat.

### 2.4 Warna RGB

*RGB* adalah singkatan dari *Red-Green-Blue*, tiga warna dasar yang dijadikan patokan warna secara universal (*primary colors*). Dengan basis *RGB*, kita bisa mengubah warna ke dalam kode-kode angka sehingga warna tersebut akan tampil universal. Warna *RGB* cocok digunakan untuk menampilkan citra / gambar dalam perangkat elektronik seperti televisi dan komputer.



Gambar 2.1 Warna RGB

### 2.5 Warna Tingkat Keabuan (*Grayscale*)

*Grayscale* merupakan warna hitam-putih yang tiap *pixel*-nya memiliki nilai warna dari 0 sampai dengan 255, dimana 0 adalah warna hitam dan 255 adalah warna putih. Nilai ini nantinya akan diproses pada komputer.



Gambar 2.2 Warna Grayscale

### 2.6 Normalisasi Citra

Normalisasi pada pengolahan citra berarti mentransformasikan citra ke bentuk citra normal yang sesuai dengan kebutuhan. Besar dan kecil ukuran citra pada saat beroperasi tidak sesuai dengan ukuran citra normalisasi. Citra hasil normalisasi dapat ditampilkan sesuai dengan keinginan. Penskalaan ini tergantung besar dan kecil ukuran pada citra yang ada pada *database*, artinya citra hasil normalisasi akan dibandingkan dengan citra pada *database* dengan kondisi dan ukuran citra yang dibandingkan adalah sama.

## 2.7 Metode Principal Component Analysis

PCA merupakan metode yang digunakan untuk ekstraksi fitur klasik dan kompresi data. Tujuan dari PCA adalah untuk mereduksi dimensi yang besar dari ruang data (*observed variables*) menjadi dimensi intrinsik yang lebih kecil dari ruang fitur (*independent variables*), yang dibutuhkan untuk mendeskripsikan data lebih sederhana.

## 2.8 Eigenface

*Eigenface* adalah kumpulan dari *eigenvector* yang digunakan untuk masalah *computer vision* pada pengenalan wajah manusia. Teknik ini telah digunakan untuk pengenalan tulisan tangan, pembacaan bibir, pengenalan suara dan pencitraan medis. *Eigenface* juga merupakan sebuah cara sederhana untuk mengekstrak informasi yang terkandung dalam citra wajah yaitu dengan menangkap variasi-variasi penting dalam sekumpulan citra wajah dan menggunakan informasi tersebut untuk mengkodekan dan membandingkan citra wajah.

*Eigenface* ini dianggap sebagai sebuah sederetan ciri yang bersama-sama memberi karakter variasi diantara citra-citra wajah. Setiap titik citra wajah bisa dinyatakan dalam satu atau lebih *eigenvector* sehingga sekumpulan *eigenvector* dapat ditampilkan sebagai sekumpulan wajah. Sekumpulan *eigenvector* yang digunakan inilah yang disebut sebagai *eigenface*.

## 2.9 Algoritma Eigenface

Algoritma pengenalan wajah menggunakan *Eigenface* dimulai dengan membuat matriks kolom dari wajah yang di-*input* ke dalam *database*. Matriks kolom tersebut akan diubah menjadi bentuk *flatvector* untuk dicari Rata-rata *vector* citra dari gambar wajah atau disebut dengan Rataan *FlatVector*. Rataan *FlatVector* dihitung dengan cara membagi penjumlahan dari seluruh *flatvector* citra dengan jumlah banyaknya citra yang disimpan di dalam *database*.

### 2.9.1 Penyusunan Flatvector Matriks Citra

Langkah pertama adalah menyusun seluruh *training image* menjadi 1 matriks tunggal. Misalnya *image* yang kita simpan berukuran H x W *pixel* dan jumlahnya N buah, maka memiliki *flatvector* dengan dimensi N x (H x W). Representasikan semua matriks *training* menjadi matriks dengan bentuk N x 1 atau matriks linier seperti yang ditunjukkan pada persamaan 2.1.

$$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix} \rightarrow [a \ b \ c \ x \ y \ z] \dots \dots \dots (2.1)$$

Contoh pada perhitungan menunjukkan empat wajah citra *training* yang telah diubah menjadi citra

matriks, lalu matriks tersebut diubah ke dalam bentuk rataan *FlatVector*.

$$C_1 = \begin{bmatrix} 10 & 10 & 10 \\ 10 & 10 & 10 \\ 10 & 10 & 10 \end{bmatrix} \rightarrow [10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10]$$

$$C_2 = \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix} \rightarrow [2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2]$$

$$C_3 = \begin{bmatrix} 11 & 11 & 11 \\ 11 & 11 & 11 \\ 11 & 11 & 11 \end{bmatrix} \rightarrow [11 \ 11 \ 11 \ 11 \ 11 \ 11 \ 11 \ 11 \ 11]$$

$$C_4 = \begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 9 & 9 & 9 \end{bmatrix} \rightarrow [9 \ 9 \ 9 \ 9 \ 9 \ 9 \ 9 \ 9 \ 9]$$

### 2.9.2 Hitung Rata-rata FlatVector

Dari *FlatVector* yang diperoleh, jumlahkan seluruh barisnya dan bagi dengan jumlah *image training* untuk mendapatkan Rata-rata (mean) *FlatVector* (lihat persamaan 2.2).

$$\text{Rata-rata FlatVector} = \frac{C_1 + C_2 + \dots + C_n}{n} \dots \dots (2.2)$$

$$\text{Rataan FlatVector} = \frac{C_1 + C_2 + C_3 + C_4}{4}$$

$$= \frac{\begin{bmatrix} 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 \\ 9 & 9 & 9 & 9 & 9 & 9 & 9 & 9 & 9 \end{bmatrix}}{4}$$

$$= \frac{[32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32]}{4}$$

$$= [8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8]$$

### 2.9.3 Tentukan Nilai Eigenface

Dengan menghitung rataan *flatvector* citra, maka nilai *eigenface* untuk matriks *flatvector* yang sudah disusun tersebut dapat dihitung nilai *eigenface*-nya. Caranya dengan mengurangi baris-baris pada matriks *flatvector* dengan rata-rata *flatvector*. Jika didapatkan nilai di bawah nol (nilai *minus*), maka nilainya diganti dengan nol.

$$C_1 = \frac{\begin{matrix} 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 \\ 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{matrix}}{-}$$

$$C_2 = \frac{\begin{array}{cccccccc} 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}}{-}$$

$$C_3 = \frac{\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 \end{array}}{-}$$

$$C_4 = \frac{\begin{array}{cccccccc} 9 & 9 & 9 & 9 & 9 & 9 & 9 & 9 \\ 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}}{-}$$

### 2.9.4 Proses Identifikasi

Untuk mengenali citra tes (*testface*) pada saat uji, langkah identifikasinya adalah hitung nilai *eigenface* untuk matriks *testface* dengan cara yang sama seperti sebelumnya yaitu dimulai dari awal penentuan nilai *flatvector*, dikurangi dengan rata-rata *FlatVector* (didapat dari citra *training*), dan mendapatkan *eigenface* untuk *testface* (lihat persamaan 2.1).

$$C_t = \begin{bmatrix} 9 & 9 & 9 \\ 9 & 7 & 9 \\ 9 & 7 & 9 \end{bmatrix} \rightarrow [999979979]$$

$$\rightarrow \frac{\begin{array}{cccccccc} 9 & 9 & 9 & 9 & 7 & 9 & 9 & 7 & 9 \\ 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{array}}{-}$$

Identifikasi dilakukan dengan metode *Euclidean Distance* yaitu menentukan jarak (*distance*) terpendek antara nilai *eigenface* dari *training image* di *database* dengan *eigenface* dari *testface*. Proses dimulai dengan menentukan nilai *absolut* dari pengurangan matriks *eigenface training image* dengan *eigenface* dari *testface* dan jumlahkan seluruh elemen penyusun *vector* yang dihasilkan. Kemudian cari nilai paling kecil dari hasil penjumlahan tersebut.

$$|Identify C_n| = \frac{\begin{array}{c} \text{Eigenface Training } C_n \\ \text{Eigenface Testface} \\ a b c d e f g h i \dots \end{array}}{.....} \quad (2.3)$$

$$|C_1| = \frac{\begin{array}{cccccccc} 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 2 & 1 & 1 & 2 & 1 \end{array}}{-}$$

$$C_1 = 1 + 1 + 1 + 1 + 2 + 1 + 1 + 2 + 1 = 11$$

$$|C_2| = \frac{\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{array}}{-}$$

$$C_2 = 1 + 1 + 1 + 1 + 0 + 1 + 1 + 0 + 1 = 7$$

$$|C_3| = \frac{\begin{array}{cccccccc} 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 2 & 2 & 2 & 2 & 3 & 2 & 2 & 3 & 2 \end{array}}{-}$$

$$C_3 = 2 + 2 + 2 + 2 + 3 + 2 + 2 + 3 + 2 = 20$$

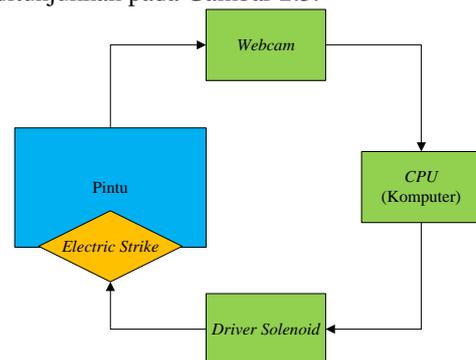
$$|C_4| = \frac{\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{array}}{-}$$

$$C_4 = 0 + 0 + 0 + 0 + 1 + 0 + 0 + 1 + 0 = 2$$

Dari hasil perhitungan diperoleh jarak citra wajah empat memiliki nilai yang terkecil yaitu 2. Karena jarak *eigenface face* empat dengan *eigenface testface* paling kecil, maka hasil identifikasi menyimpulkan bahwa *testface* lebih mirip dengan *face* empat daripada *face* satu, *face* dua, dan *face* tiga.

### 2.10 Perancangan Sistem

Perancangan sistem berisi blok diagram seperti ditunjukkan pada Gambar 2.3.



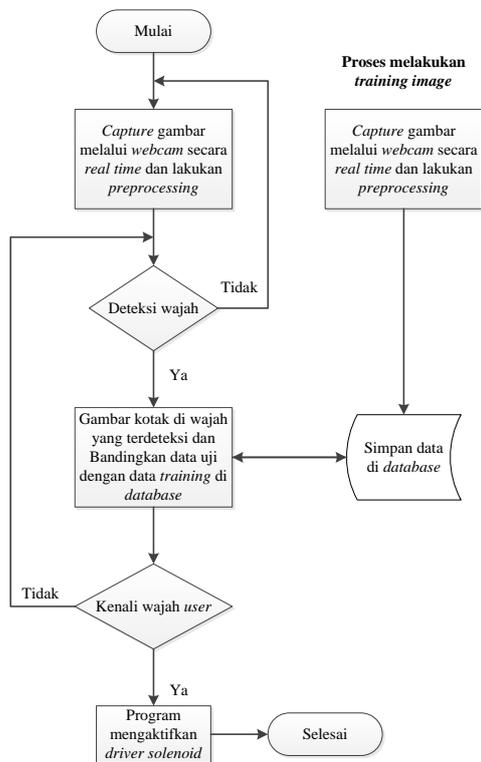
Gambar 2.3 Blok diagram

Pada blok diagram terbagi ke dalam 4 bagian, yaitu :

1. *Webcam* : perangkat ini memiliki fungsi untuk meng-*capture* gambar. *Webcam* memiliki peran sebagai data masukan.
2. *CPU (Komputer)* : pada blok ini dapat berupa komputer *desktop* ataupun *laptop*. Komputer memiliki peran penting sebagai pengolah data.
3. *Driver Solenoid* : blok ini berisi rangkaian elektronika yang berfungsi melakukan *switching* ke *solenoid* yang berada di pintu.
4. *Pintu* : tempat untuk masuk dan keluar dimana terdapat *solenoid* berupa *electric strike* yang berfungsi sebagai pengaman. *Electric strike* diaktifkan dan dinonaktifkan berdasarkan masukan tegangan dari *power supply* dan *RS-232*.

Berdasarkan blok diagram di atas, sistem terbagi menjadi beberapa tahap, yaitu tahap pengambilan gambar, tahap deteksi wajah, tahap

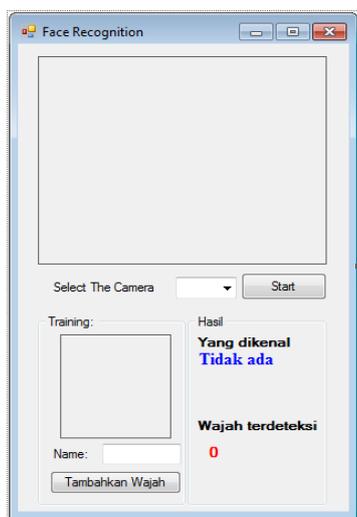
pengenalan wajah, dan tahap pengaktifan solenoid atau secara umum dapat dilihat pada *flowchart* sebagai berikut :



Gambar 2.4 *Flowchart* sistem

## 2.11 Tampilan Interface

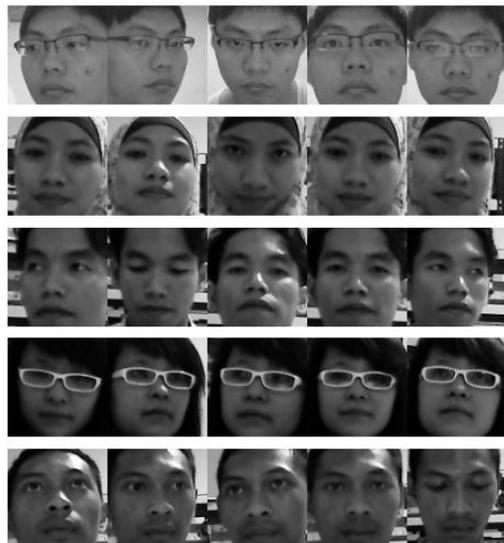
Perangkat lunak dari sistem dibuat berbentuk tampilan antarmuka yang mudah dioperasikan pengguna. Tampilan antarmuka dapat dilihat pada Gambar 2.5.



Gambar 2.5 Tampilan *GUI* pada *PC*

## 2.12 Pengambilan Data Training

Pengambilan citra *training* dilakukan dengan kondisi pencahayaan terang dan merata. Data *training* ini merupakan citra wajah yang dilatih terlebih dahulu oleh komputer yang nantinya akan digunakan sebagai data pembandingan ketika sistem pengenalan diuji. Citra wajah diambil melalui kamera (*webcam external*) yang dihubungkan dengan komputer sebagai pengolah. Ketentuan wajah yang akan dideteksi oleh kamera adalah wajah yang tidak terhalangi sebagian / seluruhnya oleh objek lain.

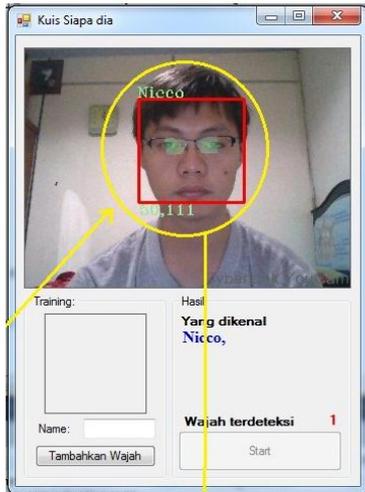


Gambar 2.6 Data *training*

Gambar 2.6 merupakan citra wajah yang dilatih dimana terdapat 5 posisi wajah yang berbeda dari setiap orang yang diambil citra wajahnya. Posisi wajah terbagi menjadi posisi depan (posisi default) dan 4 posisi lainnya (kiri, kanan, atas, & bawah) sejauh 20°-30° dari posisi depan.

## 2.13 Deteksi & Pengenalan Wajah

Pada saat sistem dijalankan, kamera akan segera memulai proses pengolahan citra untuk mendeteksi dan mengenali wajah secara bersamaan dari citra yang di-*capture* oleh *webcam* (lihat Gambar 2.7). Dua proses tersebut dijalankan bersamaan dikarenakan sistem dibuat untuk bekerja secara *real time* agar menghemat waktu akses masuk ke ruangan.

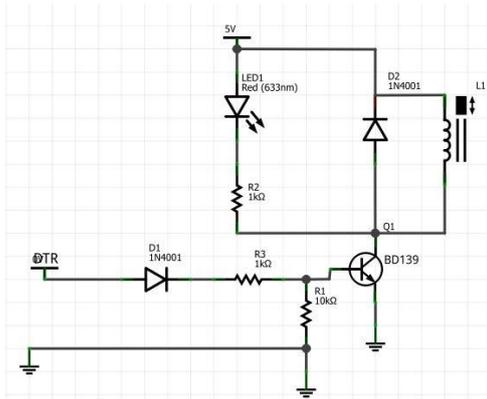


Gambar 2.7 Kotak merah merupakan tanda bahwa ada wajah terdeteksi dan tulisan 'Nicco' berarti sistem mengenali citra tersebut

### 2.14 Pengaktifan Solenoid

Sistem telah diprogram untuk memberikan hak akses masuk pada *user* tertentu (pemilik ruangan) sebelumnya. Logikanya adalah ketika sistem telah mengenali wajah dari *user*, maka program akan mengirim sinyal ke *driver solenoid* untuk mengaktifkan *solenoid* pada pintu agar pintu ruangan menjadi terbuka. Sebaliknya jika wajah yang dikenali bukan *user*, maka pintu ruangan tetap dalam keadaan terkunci.

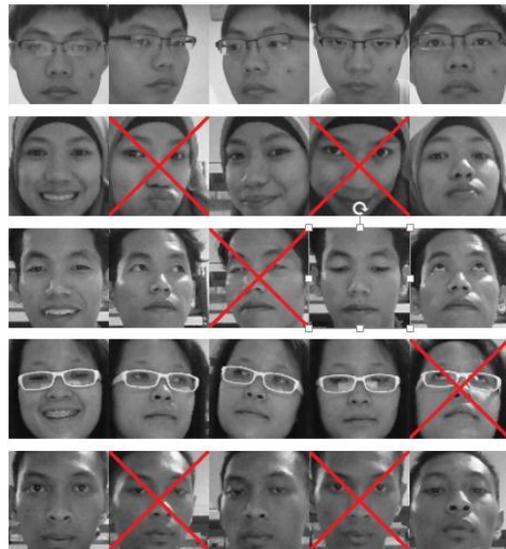
Rangkaian *driver solenoid* ini digunakan sebagai rangkaian penghubung tegangan (saklar) dari *power supply* 12 VDC ke *solenoid (electric strike)* yang dikontrol oleh komputer (lihat Gambar 2.8).



Gambar 2.8 Desain Driver Solenoid

## 3. HASIL PENELITIAN

Proses pengujian yang dilakukan pada sistem dilakukan dengan menguji masing-masing posisi sesuai dengan posisi pengambilan data training serta bersamaan dengan proses pengaktifan *solenoid*.



Gambar 3.1. Data uji pengenalan wajah

Hasil pengujian oleh sistem menunjukkan bahwa tidak semua posisi wajah dikenali oleh sistem seperti yang ditunjukkan pada gambar 3.1 (tanda silang berarti tidak dikenali sistem). Wajah diuji dengan urutan posisi depan, kiri, kanan, bawah, dan atas. Wajah yang selalu dikenali dengan baik adalah wajah dengan posisi depan.

Data training secara lengkap dapat dilihat pada Tabel 3.1 dan data uji pada Tabel 3.2.

Tabel 3.1 Data Training 30 orang

Data Training					
Orang ke-n	Jarak dari kamera sebesar 30 cm				
	Depan	20° - 30° dari posisi Depan			
		Kanan	Kiri	Atas	Bawah
1	v	v	v	v	v
2	v	v	v	v	v
3	v	v	v	v	v
4	v	v	v	v	v
5	v	v	v	v	v
6	v	v	v	v	v
7	v	v	v	v	v
8	v	v	v	v	v
9	v	v	v	v	v
10	v	v	v	v	v
11	v	v	v	v	v
12	v	v	v	v	v
13	v	v	v	v	v
14	v	v	v	v	v
15	v	v	v	v	v

16	v	v	v	v	v
17	v	v	v	v	v
18	v	v	v	v	v
19	v	v	v	v	v
20	v	v	v	v	v
21	v	v	v	v	v
22	v	v	v	v	v
23	v	v	v	v	v
24	v	v	v	v	v
25	v	v	v	v	v
26	v	v	v	v	v
27	v	v	v	v	v
28	v	v	v	v	v
29	v	v	v	v	v
30	v	v	v	v	v

Tabel 3.2 Data Uji 30 orang

Data Uji					
Orang ke-n	Jarak dari kamera sebesar 30 cm				
	Depan	20° - 30° dari posisi Depan			
		Kanan	Kiri	Atas	Bawah
1	ok	ok	ok	ok	ok
2	ok	no	ok	ok	no
3	ok	ok	ok	ok	ok
4	ok	ok	ok	ok	ok
5	ok	ok	ok	ok	ok
6	ok	ok	ok	no	ok
7	ok	ok	no	ok	ok
8	ok	ok	ok	ok	ok
9	ok	ok	ok	ok	ok
10	ok	ok	ok	ok	ok
11	ok	ok	ok	ok	ok
12	ok	ok	no	ok	no
13	ok	ok	no	ok	ok
14	ok	ok	no	ok	ok
15	ok	ok	ok	ok	ok
16	ok	ok	ok	ok	ok
17	ok	no	no	no	no
18	ok	ok	ok	ok	ok
19	ok	ok	no	ok	ok
20	ok	ok	no	ok	ok
21	ok	ok	ok	ok	ok
22	ok	no	ok	ok	ok
23	ok	no	no	ok	ok
24	ok	ok	no	ok	ok
25	ok	ok	no	ok	no
26	ok	ok	ok	ok	ok

27	ok	ok	ok	ok	ok
28	ok	ok	no	ok	ok
29	ok	ok	no	no	ok
30	ok	ok	no	ok	no

Pada data uji (Tabel 3.2), diketahui bahwa citra yang dikenal sebanyak 125 citra dari 150 citra yang ada. Tingkat pengenalan yang diperoleh sebesar 83,33% seperti yang ditunjukkan pada persamaan 3.1.

Tingkat pengenalan :

$$= \frac{125}{150} \times 100\% = 83,33\% \dots \dots \dots (3.1)$$

Tabel 3.3 Data Pengujian Solenoid

Data Pengujian Solenoid		
No.	Nama	Solenoid (Electric Strike)
1	Nicco	on
2	Dwi Rizki	off
3	Oktharina	off
4	Abdul Rois	off
5	Eko Prasetio	off
6	Raden Zulfikri Pamungkas	off
7	Noverina Br sihombing	off
8	Nengsi K	off
9	Rizky Febrianto	off
10	Edy Syahputra	off
11	Hari Wijaya	off
12	Ermanda Saputra	off
13	Bossy Stephanno Manurung	off
14	Raffi Maizar	off
15	Alpia Sandi	off
16	Sayed Musthafa Isa A	off
17	Arif Wahyu Budiarto	off
18	Muhammad Irpandi	off
19	Padillah	off
20	Teddy Sanjani S	off
21	Muhammad Farihan	off
22	Emilio Santos Abdullah	off
23	Dodi Radot Lumbanturuan	off
24	Anugerah Wibisana	off
25	Puspa Dewi	off
26	M. Adrian Ariowibowo	off
27	Moch Riau Maulana	off
28	Reggi Febriandy	off
29	Safrizal	off
30	M.P. Aidina	off

Tabel 3.3 menunjukkan pengujian *Solenoid* dilakukan terhadap 30 orang. Terlihat pada tabel bahwa *Solenoid* hanya akan aktif (kondisi 'on') jika hanya mengenali *user* tertentu, dalam hal ini *user* yang dimaksud adalah 'Nicco' yang sebelumnya sudah didefinisikan sebagai *user* di dalam program.

#### 4. KESIMPULAN

Berdasarkan uji coba dan analisa yang telah dilakukan, dapat diambil beberapa kesimpulan bahwa:

1. Hasil pengujian sistem pengenalan wajah menunjukkan bahwa sistem berjalan dengan baik dengan tingkat pengenalan di atas 80%. Citra wajah yang dikenali dengan baik adalah citra wajah pada posisi depan. Tingkat pengenalan posisi wajah depan sebesar 100%.
2. Metode *PCA* membutuhkan spesifikasi komputer yang tinggi untuk menjalankan pengolahan citra secara *real time*. Dalam sistem *real time*, semakin banyak citra yang diolah maka semakin berat untuk ditangani komputer dengan spesifikasi rendah.
3. Data *Solenoid* menunjukkan sistem dapat digunakan untuk tujuan keamanan agar dapat mencegah tindak kriminal seminimal mungkin.

#### PUSTAKA

1. [library.binus.ac.id/eColls/eThesisdoc/Bab2/TSA-2012-0089%202.pdf](http://library.binus.ac.id/eColls/eThesisdoc/Bab2/TSA-2012-0089%202.pdf), diakses April 2013.
2. Bamukrah, Jihan Faruq. 2010. *Pengertian Pengolahan Citra (Image Processing)*. Universitas Gunadarma.
3. Fatta, Hanif Al. 2007. *Konversi Format Citra RGB ke Format Grayscale Menggunakan Visual Basic*. Yogyakarta: STMIK AMIKOM Yogyakarta.
4. Puspitasari, Diah Eka. 2010. *Pengenalan Wajah Menggunakan Metode Principal Component Analysis (PCA) Untuk Aplikasi Sistem Keamanan Rumah*. Bandung: Universitas Diponegoro.
5. Vidyaningrum, Esty. Prihandoko. 2009. *Human Face Detection Using Eigenface Method for Various Pose of Human Face*. Universitas Gunadarma.

6. Indra. 2012. *Sistem Pengenalan Wajah dengan Metode Eigenface Untuk Absensi Pada PT Florida Lestari*. Jakarta: Universitas Budi Luhur.
7. Viola, Paul and Michael J. Jones, "Robust Real-Time Face Detection", *International Journal of Computer Vision*, vol. 57(2), pp. 137-154, July 2003.