

PERANCANGAN DAN IMPLEMENTASI KEAMANAN PINTU BERBASIS PENGENALAN WAJAH DENGAN METODE EIGENFACE

DESIGN AND IMPLEMENTATION SECURITY DOOR WITH FACE RECOGNITION USING EIGENFACE METHOD

Derian Indra Bramantio¹, Erwin Susanto, S.T., M.T., Ph.D.², Ramdhan Nugraha, SPd., M.T.³

^{1, 2, 3}Fakultas Teknik Elektro, Universitas Telkom

¹derianib@student.telkomuniversity.ac.id, ²erwinelektro@telkomuniversity.ac.id,
³ramdhan@telkomuniversity.ac.id

Pengenalan wajah adalah hal yang mudah bagi manusia namun bagaimana pengenalan wajah dengan mesin atau komputer. Kita mengidentifikasi orang dengan melihat mata dan wajahnya. Begitu juga sama halnya dengan komputer. Hanya saja manusia mampu identifikasi orang lebih cepat dan akurat. Pada tugas akhir ini pengenalan wajah akan diimplementasikan pada pintu melalui motor servo. Diimplementasikan pada pintu di rumah karena dalam rumah terdapat barang-barang yang berharga untuk melindunginya ditambahkan sistem keamanan agar hanya pemilik rumah dan keluarganya yang dapat masuk ke dalam rumah tersebut. 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 PCA (*Principle Component Analysis*). Penggunaan citra input diharapkan bisa mejadi 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.

Kata Kunci : *face recognition, PCA, eigenface, raspberry pi, motor servo*

Face recognition is easy for humans, but what about face recognition by machine or computer. We identify people by looking at their eyes and face. So also the same as with a computer. Human is capable of identifying people more quickly and accurately. This face recognition will be implemented on the door via a servo motor. Implemented on the door in the house because inside the house there are valuable items to protect. In order to make those items safe, the owner must add security system that only owner and their families can get into the house. This face recognition method created with python programming language by using Raspberry Pi. Design of this system is in order to seek its own position of a face in an image. The algorithm is *Eigenface* or commonly called PCA (Principle Component Analysis). The Expectation of this face recognition has to form the image more flexible and has an accuracy of up to 90% so that only people who are in a database that can be entered into the house. While the servo motor is designed in order to control the position to open the key slot.

Keywords : *face recognition, PCA, eigenface, raspberry pi, motor servo*

1. Pendahuluan mengimplementasikannya pada pintu.

Pada saat ini banyak pencuri yang berhasil memasuki rumah yang kosong, terutama pada saat lebaran atau liburan panjang telah tiba. Semua keluarga mudik atau pergi ke luar kota meninggalkan rumah tersebut kosong tanpa menggunakan keamanan yang canggih. Pada normalnya pemilik rumah tersebut hanya menggunakan gembok pagar atau rantai. Dikarenakan sistem keamanan rumah kurang menggunakan alat-alat yang canggih atau hanya menggunakan alat yang pada umumnya digunakan seperti gembok, pencuri dapat gampang memasuki rumah. Oleh karena itu, penulis ingin membuat sistem keamanan yang canggih dan

Sistem keamanan yang banyak digunakan pada saat ini contohnya seperti *fingerprnt* dan *password* alarm . Sudah banyak alat-alat *fingerprnt* yang sudah dijual di toko. Harganya pun termasuk masih mahal. *Fingerprnt* itu mudah digunakan. *Fingerprnt* ini mempunyai kelemahan yaitu jika orang tersebut luka pada sidik jarinya dan jika pencuri tersebut memiliki sidik jari pemilik rumahnya. Sidik jari dapat ditemukan misalnya jika pemilik rumah tersebut minum dari minuman kaleng lalu dibuangnya ke tempat sampah, dari minuman kaleng tersebut pasti terdapat sidik jari dari pemilik rumah tersebut dan akhirnya pencurinya dapat sidik jarinya. Sedangkan untuk *password* alarm

mempunyai kelemahan jika password itu tidak pernah diganti oleh pemiliknya sehingga terdapat bekas tombol-tombol yang pernah ditekan oleh pemiliknya.

Oleh karena itu, penulis membuat *face recognition* untuk keamanan pintu. Sampai saat ini teknologi *face recognition* belum sempurna sehingga harus disempurnakan. Metode yang digunakan adalah *PCA (Principle Component Analysis)* atau biasa disebut dengan *Eigenface*. Sistem *face recognition* ini sangatlah unik karena muka manusia berbeda-beda. Contoh variabelnya seperti mata, hidung dan mulut. Jadi metode ini pada dasarnya adalah mereduksi citra menjadi vektor ciri. Sehingga komputasi yang dilakukan akan menjadi lebih sedikit [1]. Simulasi pengambilan gambar wajah akan dilakukan beberapa kali hingga mendapatkan hasil yang terbaik [2].

1.1. Tujuan

Tujuan dari perancangan dan implementasi ini berdasarkan latar belakang yang telah dipaparkan adalah: Membuat program untuk merancang sistem *face recognition*, merancang dan mengimplementasikan *face recognition* pada pintu, dapat mengontrol posisi servo motor agar dapat membuka slot pintu dan dapat melakukan analisis tingkat akurasi dari *face recognition* tersebut.

1.2. Rumusan Masalah

Terdapat rumusan masalah mengenai tugas akhir yang penulis susun, yaitu seperti cara membuat agar tingkat akurasi *face recognition* tinggi, cara merancang *face recognition* pada system dan cara untuk mengontrol posisi pada motor servo.

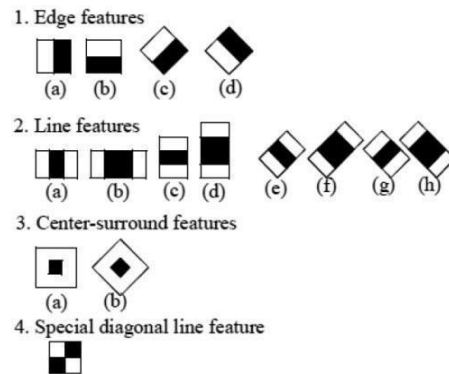
1.3. Metode Penelitian

Dalam menyelesaikan masalah dalam penelitian ini akan terdapat dua metode yang akan digunakan yaitu untuk mendeteksi muka akan digunakan *Haar Like Feature* sedangkan untuk pengenalan wajahnya akan dilakukan metode *Eigenface*.

2. Dasar Teori dan Perancangan

2.1. Haar Like Feature

Haar Like Feature [3] yang memproses citra dalam wilayah kotak-kotak, berisi beberapa *pixel* dari bagian citra. Pixel-pixel dalam satu wilayah tersebut dijumlahkan dan dilakukan proses perhitungan (pengurangan rata-rata nilai pixel di bagian kotak yang terang dan gelap) sehingga diperoleh perbedaan nilai unik disetiap wilayah kotak-kotak tersebut.



Gambar 1 Menunjukkan model fitur *Haar Like* yang dikembangkan oleh Lienhart, Kuranove dan Pisarevsky untuk mendeteksi objek wajah [4].

2.2. Eigenface

Metode *Eigenface* ditemukan oleh Matthew A Turk dan Alex P. Pentland dari MIT pada tahun 1991 [5]. Tujuan utama dari metode *Eigenface* adalah untuk mendapatkan karakteristik citra dengan tidak menggunakan karakteristik wajah tetapi dengan menggunakan rumus transformasi matematika.

2.2.1. Tahap perhitungan Eigenface[5]

Menyiapkan database dengan membuat suatu himpunan X yang terdiri dari seluruh *training image*

$$X = [x_1, x_2, \dots, x_n] \tag{1}$$

Mencari nilai *mean* gambar (μ)

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i \tag{2}$$

Hitung matriks covarians (S). Dimana $(x_i - \mu)$ adalah selisih antara training image dengan *mean* dari wajah

$$S = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T \tag{3}$$

Hitung *eigenvalues* (λ) dan *eigenvektor* (w) dari S

$$S w = \lambda w \tag{4}$$

Mengurutkan *eigenvektor* secara menurun dengan *eigenvalue* . *K principal components* adalah *eigenvektor* yang sesuai dengan *eigenvalue* nilai k terbesar. Memproyeksikan semua sampel pelatihan ke subruang PCA

$$y = W^T (x - \mu) \tag{5}$$

Dimana $W = (v_1, v_2, \dots, v_k)$. Memproyeksikan citra uji ke dalam subruang PCA

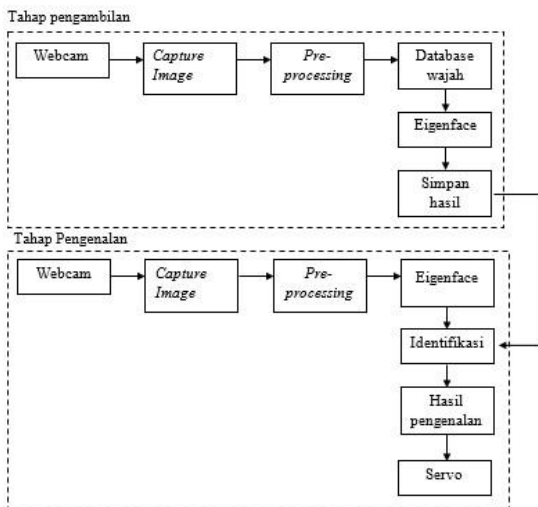
$$x = Wy + \mu \tag{6}$$

Menemukan nilai terdekat antara gambar pelatihan yang diproyeksikan dan gambar uji yang diproyeksikan.

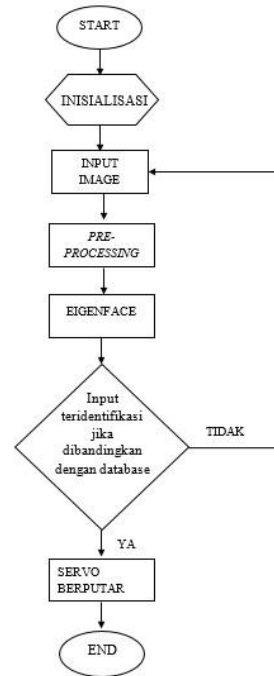
2.3. Vektor

Melalui persamaan (4), vektor akan digunakan karena jika menggunakan matriks perhitungannya akan sangat besar dan lambat. Menggunakan vektor ukuran gambar relatif lebih kecil jika dibandingkan dengan menggunakan bitmap. Dengan menggunakan vektor menggunakan yang sangat berkurang, dari urutan jumlah piksel dalam gambar dengan urutan jumlah gambar dalam training set. Dalam prakteknya, training set gambar wajah akan relatif kecil dan perhitungan menjadi sangat mudah dikelola.

2.4. Cara kerja Sistem



Gambar 2 Diagram blok sistem



Gambar 3 Diagram alir dari sistem

Gambar 2 dan gambar 3 adalah diagram blok dan diagram alir dari sistem pengenalan wajah yang akan dibuat. Untuk gambar 2, gambar diagram blok disini memiliki 2 tahap pengambilan dan tahap pengenalan. Di tahap pengambilan ini foto yang diambil akan dijadikan database lalu akan disimpan hasilnya. Setiap foto berukuran (250x250) pixel. Lalu ada *pre-processing*, itu adalah tahap *edit* foto agar foto siap dipergunakan/disimpan. Dalam *pre-processing* itu sendiri ada empat tahap antara lain *cropping*, *resize*, *grayscale*, *histogram equalization*. Setelah melalui proses itu lalu foto akan melalui tahap *Eigenface* yang perhitungannya sudah dibahas sebelum ini. Lalu menuju tahap pengenalan disinilah tempat kedua matrix dari foto dibandingkan. *Threshold* itu adalah batas tingkat kemiripan dan di dalam kasus ini kita yang menentukan *threshold* berdasarkan percobaan yang sudah dilakukan sebelumnya. Jika foto tersebut melampaui batas *threshold* yang ditentukan sebelumnya maka foto tidak akan dikenali.

2.5. Implementasi

Disini alat akan diimplementasikan pada motor servo. Jadi pertama-tama saat ada orang ingin membuka pintu. Orang tersebut harus menghadap webcam agar foto dari wajahnya bisa diambil. Setelah itu sistem akan mulai bekerja untuk mulai mengolah foto itu lalu dikalkulasikan dengan menggunakan algoritma *Eigenface*. Setelah itu akan muncul tulisan bawah foto itu dapat atau tidak dikenali. Jika dapat dikenali maka sistem akan mengirim sinyal berupa *pulse* yang akan diterima oleh motor servo dan akhirnya motor servo akan

berputar untuk membuka pintu. Setelah 10 detik, kunci akan otomatis terkunci kembali.

3. Pengujian dan Pembahasan

Tabel 1(a) Pengujian pada gambar yang sama

No	Gambar yang diuji	Hasil <i>Predict</i>
1	Database I – Gambar 1	0.0
2	Database I – Gambar 2	2572.35
3	Database I – Gambar 3	3155.36
4	Database I – Gambar 4	3651.69
5	Database I – Gambar 5	4508.42
6	Database II – Gambar 1	0.0
7	Database II – Gambar 2	4168.79
8	Database II – Gambar 3	5808.67
9	Database II – Gambar 4	4925.29
10	Database II – Gambar 5	3573.61
11	Database III – Gambar 1	0.0
12	Database III – Gambar 2	3031.55
13	Database III – Gambar 3	2147.16
14	Database III – Gambar 4	3491.81
15	Database III – Gambar 5	3232.12
16	Database IV – Gambar 1	0.0
17	Database IV – Gambar 2	4627.72
18	Database IV – Gambar 3	3315.73
19	Database IV – Gambar 4	1449.35
20	Database IV – Gambar 5	3056.15

Tabel 1(b) Pengujian pada gambar yang mirip

No	Gambar yang diuji	Hasil <i>Predict</i>
1	Database I – Gambar 1	4737.38
2	Database I – Gambar 2	6351.70
3	Database I – Gambar 3	5873.87
4	Database I – Gambar 4	3334.53
5	Database I – Gambar 5	4428.39
6	Database II – Gambar 1	4469.16
7	Database II – Gambar 2	5189.13
8	Database II – Gambar 3	4875.85
9	Database II – Gambar 4	6771.11
10	Database II – Gambar 5	3435.05
11	Database III – Gambar 1	5541.30
12	Database III – Gambar 2	3301.48
13	Database III – Gambar 3	3524.34
14	Database III – Gambar 4	2694.16
15	Database III – Gambar 5	4899.71
16	Database IV – Gambar 1	3531.90
17	Database IV – Gambar 2	4359.15
18	Database IV – Gambar 3	4822.65
19	Database IV – Gambar 4	4094.43
20	Database IV – Gambar 5	4219.37

Pada tabel 1(a) dan 1(b) adalah tabel hasil pengujian. Tabel 1(a) adalah pengujian terhadap foto yang sama dari database yang disediakan. Tabel 1(b) adalah pengujian terhadap foto yang mirip yang artinya foto ini diambil langsung melalui webcam hanya dengan pencahayaan yang sama karena cahaya salah satu faktor penting untuk menambah akurasi. Hasil angka ini yang nanti akan dipergunakan untuk *threshold*. Angka *predict* ini didapatkan dari hasil perhitungan algoritma

eigenface.

3.1. Penentuan *threshold*

Hasil dari pengujian terhadap gambar yang sama *predict*-nya di berada angka maksimal. Sedangkan pengujian terhadap gambar yang mirip berada di angka maksimal. Pengujian terhadap gambar yang berbeda *predict*-nya di angka. Setelah mendapatkan hasil prediksi dari kedua tabel 1(a) dan 1(b) diatas maka dapat disimpulkan untuk *threshold*-nya harus dengan kriteria persis diatas angka *predict* terhadap gambar yang sama dan gambar yang mirip.. Jadi dapat diambil kesimpulan, *threshold* yang akan digunakan di angka 6000.00

3.2. Tingkat akurasi

Dalam pengenalan wajah akurasi yang didapatkan sangatlah susah jika harus 100% karena bnyak faktor-faktor yang dapat mengurangi akurasi. Contohnya seperti cahaya, tempat pengambilan foto atau mungkin bisa wajahnya. Jadi disini akan dihitung seberapa besar akurasi yang didapatkan dari tugas akhir ini.

Akurasi ini akan dibagi dua. Pertama akurasi jika pengujiannya menggunakan database yang ada. Kedua jika pengujiannya langsung menggunakan webcam. Untuk akurasi yang didapatkan dari pengujian menggunakan database yang sudah ada diperoleh hasil sebagai berikut. Dari pengujian yang dilakukan sebanyak 50 kali. Wajah yang berhasil dikenal sebanyak 50 dan tidak dikenal nol. Oleh karena itu untuk akurasi pengujian terhadap database yang ada sebesar 100%.

Dengan menggunakan *threshold* sebesar 6000.00. Untuk akurasi yang kedua yaitu jika pengujiannya melalui webcam dari 50 kali percobaan yang dilakukan didapatkan hasil sebagai berikut. Dari 50 kali dapat dikenali sebanyak 46 buah gambar sedangkan empat gambar sisanya tidak dapat dikenali. Sehingga didapatkan perhitungan akurasi sebesar 92%.

4. Kesimpulan dan Saran

4.1. Kesimpulan

Setelah melakukan perencanaan pemrograman perangkat lunak sistem pengenalan wajah (*face recognition*) serta melalui pengujian dengan sudah menggunakan slot kunci maka dapat diambil kesimpulan antara lain:

Sistem pengenalan wajah (*face recognition*) dengan menggunakan metode *Eigenface* dapat mengenali orang sesuai dengan label (nama) yang diberikan pada database dan tidak dapat mengenali orang yang tidak terdapat di database dan pada penelitian ini, dengan menggunakan *database* hasil dari foto yang dibuat sendiri ukuran tiap foto yang digunakan adalah (250x250) pixel dengan sudah

hasil *pre-processing* maka diperoleh persentase akurasi sebesar 92% dari 50 kali percobaan.

Setelah berhasil dalam mengenali wajah, sistem akan mengirimkan sinyal berupa *pulse* kepada servo agar servo dapat berputar untuk membuka kunci

4.2. Saran

Setelah mengambil kesimpulan dari setiap analisis yang dilakukan, adapun saran-saran agar sistem pengenalan wajah (*face recognition*) ini dikembangkan lebih lanjut dan lebih disempurnakan antara lain:

1. Pengenalan wajah ini layak dikembangkan lebih lanjut oleh peneliti-peneliti lainnya dengan menggabungkan beberapa metode yang ada untuk dapat menghasilkan akurasi yang lebih baik dan respon lebih cepat.
2. Pencahayaan pada saat pengambilan database dan saat diuji hendaknya sama agar mendapatkan akurasi yang lebih baik.
3. Untuk yang melanjutkan bisa dikembangkan dengan cara pengenalan wajah bisa dimana dan kapan saja.

Daftar Pustaka :

- [1] Sigit W, Bima SBD, Sandra AP. 2011. *Sistem Pengenalan Wajah Pada Mesin Absensi Mahasiswa Menggunakan Metode PCA Dan DTW*. Electronic Engineering Polytechnic Institute of Surabaya (EEPIS), Indonesia
- [2] Dewi AR., Indri Septadepi. 2010. *Face Recognition System Using Eigenface Method based on Facial Component Region. Department of Informatics, Gunadarma University*
- [3] Freund Y. & Schapire R., "A Short Introduction to Boosting", *Journal of Japanese Society for Artificial Intelligence*, 14(5)- 780, September, 1999.
- [4] Lienhart, R & Maydt, J. "An extended Set of Haar-like Features for Rapid Object Detection", *IEEE ICIP 2002*, Vol 1, pp. 900-903, Sep. 2002.
- [5] Matthew A. Turk and Alex P. Pentland. (2005). *Face Recognition Using Eigenfaces. IEEE*, 1991.