

Implementasi Pengenalan Wajah Menggunakan PCA (Principal Component Analysis)

Dian Esti Pratiwi*¹, Agus Harjoko²

¹Program Studi Elektronika dan Instrumentasi

¹Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: *dhee_antz04@yahoo.co.id, aharjoko@ugm.ac.id

Abstrak

Sistem identifikasi berkembang dengan cepat. Perkembangan tersebut mendorong kemajuan sistem keamanan berbasis biometrik. Pengenalan wajah adalah salah satu sistem identifikasi yang dikembangkan berdasarkan perbedaan ciri wajah seseorang berbasis biometrik yang memiliki keakuratan tinggi.

Eigenface merupakan salah satu metode pengenalan wajah berdasarkan Principal Component Analysis (PCA) yang mudah diimplementasikan. Eigenface dimulai dengan pemrosesan awal untuk mendapatkan hasil citra yang lebih baik. Setelah itu menghitung eigenvector dan eigenvalue dari citra wajah untuk dilakukan proses training image. Proses training wajah yaitu mencari eigenvector, eigenvalue dan average image yang diproyeksikan ke dalam subruang PCA. Proyeksi ke dalam subruang PCA digunakan untuk menyederhanakan data citra yang tersimpan. Perbandingan terkecil proyeksi PCA antara file database dan input menentukan hasil nama pengguna. Perbandingan nilai terkecil dicari menggunakan Nearest Neighbor.

Program pengenalan wajah menampilkan salah satu nama pengguna yang telah tersimpan dalam database. Pengujian menggunakan ekspresi senyum dan tanpa ekspresi pada delapan orang dan 16 wajah. Prosentase keberhasilan proses pengenalan wajah adalah 82,81%. Beberapa faktor yang mempengaruhi keberhasilan pengenalan yaitu pencahayaan pada wajah, jarak wajah dengan webcam, banyaknya gambar wajah orang yang tersimpan dan performa komputer yang digunakan.

Kata kunci— Eigenface, eigenvector, eigenvalue, average image

Abstract

Identification system grow quickly. The development encourage security system progress based biometric. Face recognition is one of the identification system is developed based on different characteristic of a person's face based biometric which has a high accuracy.

Eigenface is one method of face recognition based on PCA (Principal Component Analysis) which easy to implement. It begin with initial processing to get a better image. Then computing eigenvector and eigenvalue from face image for further training image process. Training process is finding the eigenvector, eigenvalue and average image to be projected into the PCA subspace. Projection into the PCA subspace is used to simplify the image data stored. The smallest PCA projection comparison between the database and the input file is determinants the result of username. Smallest value comparison searched using Nearest Neighbor.

Face recognition program show one of username that has been stored in a database. The test using smile expression and without expression in eight people and 16 faces. The percentage of successful face recognition process is 82,81%. Several factors that influence the success of the recognition are lighting on the face, the face distance with a webcam, sum face image of people saved and used computer performance.

Keywords— Eigenface, eigenvector, eigenvalue, average image

1. PENDAHULUAN

Sistem identifikasi merupakan hal yang penting. Salah satu sistem identifikasi yang banyak dikembangkan saat ini yaitu sistem identifikasi menggunakan informasi biologis yaitu sidik jari, wajah, retina, suara dan lain-lain. Salah satu informasi biologis yang sekarang banyak dikembangkan dan memiliki tingkat keakuratan tinggi yaitu wajah. Wajah seseorang memiliki ciri unik masing-masing yang dapat diidentifikasi. Identifikasi tersebut dapat digunakan untuk sistem pencarian seseorang di dalam sebuah gambar yang berisi wajah. Eigenface merupakan salah satu metode pengenalan wajah berdasarkan Principal Component Analysis (PCA) yang mudah diimplementasikan [1]. Lyon dan Vincent melakukan sebuah penelitian untuk membuat sebuah prototype “*Interactive Embedded Face Recognition*” [2]. Sistem ini menghadirkan pendekatan warna kulit menggunakan warna YCbCr untuk pendeteksian wajah yang lebih akurat. Pemrosesan gambar wajah ini menggunakan kombinasi operator morfologi dan *elliptical shape* dari wajah untuk proses segmentasi wajah. Secara umum, proses pendeteksian wajah ini dimulai dengan pengambilan gambar, pendeteksian kulit, pendeteksian wajah untuk selanjutnya dilakukan *face recognition*.

2. METODE PENELITIAN

2.1 Pengolahan Citra Digital

Menurut Munir menjelaskan pengolahan citra adalah pemrosesan citra, khususnya menggunakan komputer, menjadi citra yang kualitasnya lebih baik [3]. Meskipun sebuah citra kaya akan informasi, namun sering kali mengalami penurunan mutu, misalnya mengandung cacat atau derau (*noise*), warnanya terlalu kontras, kurang tajam, kabur dan sebagainya. Citra semacam ini sulit untuk diinterpretasikan karena informasi yang disampaikan oleh citra tersebut berkurang. Dengan pengolahan citra, citra yang mengalami gangguan tersebut dimanipulasi menjadi citra lain yang kualitasnya lebih baik sehingga mudah diinterpretasikan oleh manusia maupun mesin.

2.2 PCA (Principal Component Analysis)

Principal Component Analysis (PCA) adalah sebuah cara untuk mengidentifikasi pola pada data dan kemudian mengekspresikan data tersebut ke bentuk yang lain untuk menunjukkan perbedaan dan persamaan antar pola [4]. Tujuan dari PCA adalah untuk mereduksi dimensi yang besar dari ruang data (*observed variables*) menjadi dimensi yang lebih kecil dari ruang fitur (*independent variables*), yang dibutuhkan untuk mendeskripsikan data lebih sederhana. Ruang fitur adalah ciri yang digunakan sebagai kriteria dalam pengklasifikasian.

Sebuah image 2D dengan dimensi b baris dan k kolom dapat direpresentasikan kedalam bentuk image 1D dengan dimensi n ($n=b*k$). Dengan ekspresi lain dapat dituliskan sebagai \mathcal{R}^n , adalah ruang image dengan dimensi n . *Image training* yang digunakan sebanyak K sample dinyatakan dengan $\{x_1, x_2, \dots, x_K\}$ yang diambil dari sebanyak C obyek/kelas yang dinyatakan dengan $\{X_1, X_2, \dots, X_C\}$. Total *matrix scatter* S_T (atau *matrix covariance*) didefinisikan sebagai berikut:

$$S_T = \sum_{k=1}^K (x_k - \mu)(x_k - \mu)^T \quad (1)$$

dimana μ adalah rata-rata *sample image* yang diperoleh dengan merata-rata *training image* $\{x_1, x_2, \dots, x_K\}$. Dengan dekomposisi *eigen, matrix covariance* ini dapat didekomposisi menjadi:

$$S_T = \Phi \Lambda \Phi^T \quad (2)$$

dimana Φ adalah *matrix eigenvector*, dan Λ adalah is a diagonal *matrix* dari nilai *eigen*. Kemudian dipilih sejumlah m kolom *eigenvector* dari *matrix* Φ yang berasosiasi dengan sejumlah m nilai *eigen* terbesar. Pemilihan *eigenvector* ini menghasilkan *matrix* transformasi atau *matrix* proyeksi Φ_m , yang mana terdiri dari m kolom *eigenvector* terpilih yang biasa disebut juga dengan '*eigenimage*'. Berikutnya sebuah *image* x (berdimensi n) dapat diekstraksi kedalam *feature* baru y (berdimensi $m < n$) dengan memproyeksikan x searah dengan Φ_m sebagai berikut:

$$y = \Phi_m x \quad (3)$$

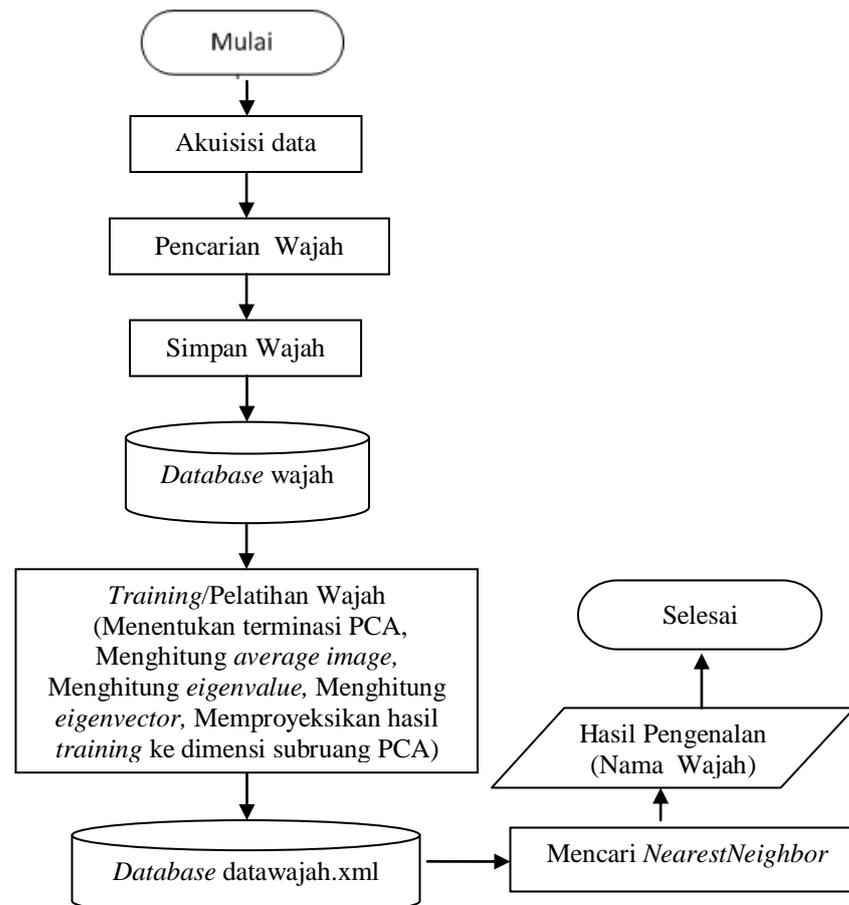
Dengan kata lain metode PCA memproyeksikan ruang asal \mathfrak{R}^n kedalam ruang baru yang berdimensi lebih rendah \mathfrak{R}^m , yang mana sebanyak mungkin kandungan informasi asal tetap dipertahankan untuk tidak terlalu banyak hilang setelah dibawa ke dimensi *feature* yang lebih kecil. Disini terlihat reduksi *feature* yang signifikan dari n buah menjadi m buah yang tentunya akan sangat meringankan komputasi dalam proses pengenalan berikutnya.

Total *matrix scatter* S_T diatas sesungguhnya adalah jumlahan dari *matrix scatter* dalam kelas (*within-class scatter matrix*) S_W dan *matrix scatter* antar kelas (*between-class scatter matrix*) S_B yaitu, $S_T = S_W + S_B$. Dengan demikian, kekurangan utama yang terlihat disini adalah bahwa dalam proses PCA ke dua *matrix scatter* ini termaksimalkan bersama-sama. Sesungguhnya yang diinginkan adalah hanya maksimalisasi S_B saja, sedangkan S_W sebaiknya diminimalkan agar anggota didalam kelas lebih terkumpul penyebarannya yang pada akhirnya dapat meningkatkan keberhasilan pengenalan. Misalkan pada variasi perubahan iluminasi maupun skala dari *image* yang terjadi pada obyek yang sama, dapat menyebabkan *matrix scatter* dalam kelas menjadi besar yang cukup menyulitkan dalam proses pengenalan. Bila ini terjadi, dengan demikian PCA akan menyertakan variasi iluminasi didalam *eigenimage*-nya, dan konsekuensinya PCA menjadi tidak tepat terhadap variasi iluminasi yang terjadi pada obyek.

Eigenface adalah salah satu algoritma pengenalan wajah yang berdasarkan pada *Principle Component Analysis* (PCA). Dalam fase pengenalan, *eigenface* mereduksi dimensi dari *input* gambar dengan memproyeksikannya ke dalam subruang yang ditemukan selama pelatihan. Memproyeksikan ke dalam subruang berarti menemukan titik terdekat dari gambar pelatihan. Subruang adalah dimensi terendah menggambarkan bahwa *eigenface* ditemukan selama fase pelatihan. Algoritma *Eigenface* secara keseluruhan cukup sederhana. *Training image* direpresentasikan dalam sebuah *vektor flat* (gabungan vektor) dan digabung bersama-sama menjadi sebuah matriks tunggal. *Eigenvector* kemudian diekstraksi dan disimpan dalam file *temporary* atau database. *Training image* kemudian diproyeksikan dalam *feature space*, bernama *face space* yang ditentukan oleh *eigenvektor*..

2.3 Rancangan Sistem

Sistem kerja program ini menggunakan sebuah *webcam* PC untuk mengambil gambar wajah seseorang kemudian dilakukan proses *face recognition*. Blok diagram program secara keseluruhan dari sistem menggunakan *face recognition* ini dapat dilihat pada Gambar 1. Proses pada sistem ini dimulai dengan pengaturan mode yang akan digunakan. Pada sistem ini memiliki dua mode, yaitu mode pelatihan dan mode penggunaan. Mode pelatihan adalah sebuah mode dimana pengguna dapat memasukkan data-data wajah yang akan digunakan untuk proses pengenalan wajah.



Gambar 1 Diagram Blok Sistem

Mode pelatihan diawali dengan pengambilan wajah menggunakan *webcam*. Pada wajah pengguna dilakukan pencarian wajah untuk menentukan area wajah. Pencarian area wajah dilakukan menggunakan *Haar Cascade Frontal Face*. Area wajah yang sudah ditemukan selanjutnya dilakukan proses *cropping* dan disimpan ke dalam *database* wajah dalam bentuk *.jpg. Ukuran wajah yang tersimpan dalam *database* sama yaitu 600x600 piksel. Hasil wajah yang disimpan tersebut selanjutnya dilatih untuk dilakukan proses pengenalan wajah dan disimpan di dalam *database* hasil pelatihan wajah. Mode penggunaan adalah mode untuk melakukan proses pengenalan dari sebuah wajah. Pada mode ini terdapat tombol kenali yang akan mengeluarkan output berupa nama pengguna yang tertera di dalam *database* wajah. Prinsip pengenalan wajah dengan *eigenface* yaitu menghitung perbandingan nilai rata-rata wajah dari beberapa *training image* wajah. *Eigenface* pertama yaitu perbedaan nilai rata-rata wajah yang paling dominan, *eigenface* kedua yaitu perbedaan nilai rata-rata wajah yang paling dominan kedua, dan begitu seterusnya. Hasil dari *eigenvector*, *eigenvalue* dan nilai rata-rata wajah ini disimpan dalam *datawajah.xml*.

Pada program ini, menggunakan beberapa *file* yang saling berhubungan antara satu dengan lainnya yang disebut *database*. *File* yang diperlukan untuk menyusun *database* ini antara lain:

1. *daftarnama.txt* yang digunakan untuk menyimpan daftar nama orang.
2. *daftarwajah.txt* yang digunakan untuk menyimpan daftar wajah orang.

Data *file* yang tersimpan terdiri dari beberapa kolom, antara lain:

- a. Kolom 1 : kolom berisi nomor wajah.
- b. Kolom 2 : kolom berisi *path* folder dan *file* gambar yang digunakan untuk menyimpan wajah.
- c. Kolom 3 : kolom yang berisi nama orang yang memiliki wajah.

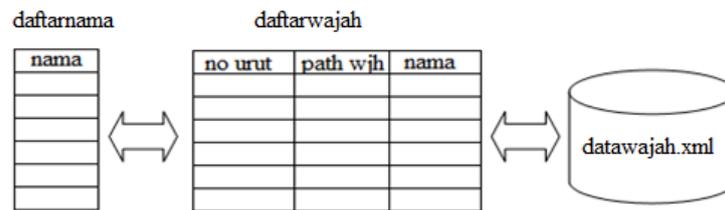
Masing-masing kolom wajah ini dipisahkan dengan menggunakan spasi, jadi nama orang yang diketikkan saat proses pelatihan tidak boleh menggunakan spasi supaya penggunaan kolom sesuai pada pengaturan diatas.

Contohnya : - 1 wajah/dian1.jpg dian

Penjelasannya :

- Kolom 1 = 1
 - Kolom 2 = wajah/dian1.jpg
 - Kolom 3 = dian
3. Folder *eigenface*/wajah yang digunakan unruk menyimpan *file* wajah orang.
 4. *datawajah.xml* yang digunakan untuk menyimpan formula wajah orang. *File* ini didapatkan dari hasil *generate* fungsi menggunakan *openCV*.

Gambaran umum *database* yang digunakan dapat dilihat pada Gambar 2.

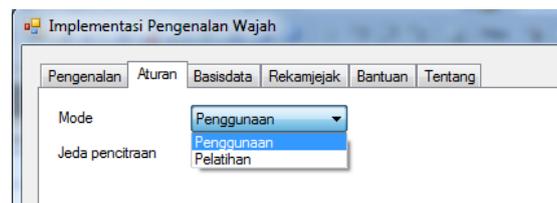


Gambar 2 Diagram blok *Database*

Penentuan nama pengguna yang sesuai pada proses pengenalan dilakukan dengan mencari perbedaan nilai *test* wajah dan nilai *train* wajah. Perbedaan nilai ini dimisalkan d. d yang terkecil dimisalkan *d_kecil*. *d_kecil* inilah yang dinamakan nilai *NearestNeighbor* yang nantinya digunakan untuk mencari namaPerson di dalam array *person*. Hasil dari pencarian dalam *array person* ini yang nantinya ditampilkan sebagai *output* nama pengguna.

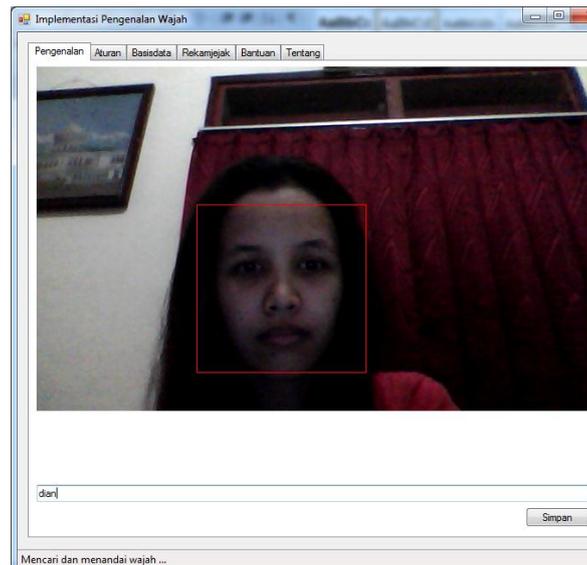
2.4 Implementasi

Ketika *tab* aturan dipilih maka program akan menampilkan layar pengaturan. Pada *tab* ini, pengguna dapat memilih mode pelatihan atau mode pengenalan. Tampilan pemilihan mode ditunjukkan pada Gambar 3.



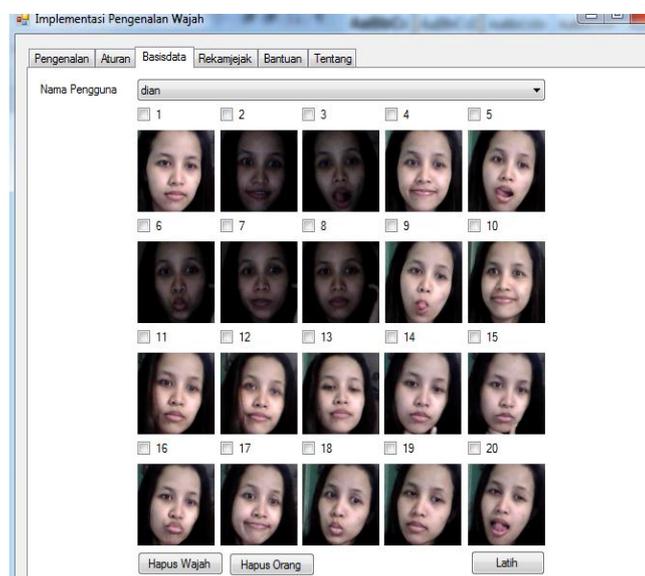
Gambar 3 Tampilan *Tab* Pemilihan Mode

Apabila pengguna belum pernah memasukkan berkas wajah, mode yang harus dipilih yaitu mode pelatihan. Setelah menentukan mode pelatihan, lalu membuka tab pengenalan. Tab pengenalan akan berubah seperti pada Gambar 4. Pada tab pelatihan pengguna harus mengisi nama ke kolom pengisian teks dengan menggunakan nama terdiri dari satu kata. Setelah itu mengambil gambar wajah dengan menekan tombol simpan.



Gambar 4 Tampilan tab pengenalan mode pelatihan

Untuk melihat hasil wajah yang telah tersimpan dengan menekan tab basisdata. Pada tab ini pengguna harus memilih satu nama pengguna yang telah dimasukkan sebelumnya. Apabila salah satu nama pengguna sudah dipilih, maka akan muncul gambar wajah pengguna yang telah tersimpan. Jumlah maksimal gambar wajah satu pengguna adalah 20 gambar wajah. Tampilan tab basisdata dari gambar pengguna ditunjukkan Gambar 5. Pada tampilan Gambar 5 pengguna dapat menyeleksi gambar wajah yang akan dihapus dengan memberikan tanda pada *checkbox*. Pengguna dapat menghapus satu atau beberapa gambar wajah pengguna dari nama pengguna dengan menekan tombol hapus wajah. Selain itu juga dapat menghapus nama pengguna yang telah tersimpan dengan menekan tombol hapus orang.

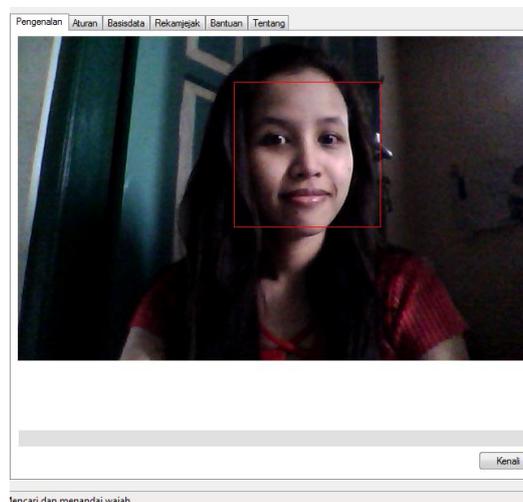


Gambar 5 Tampilan tab basisdata wajah pengguna

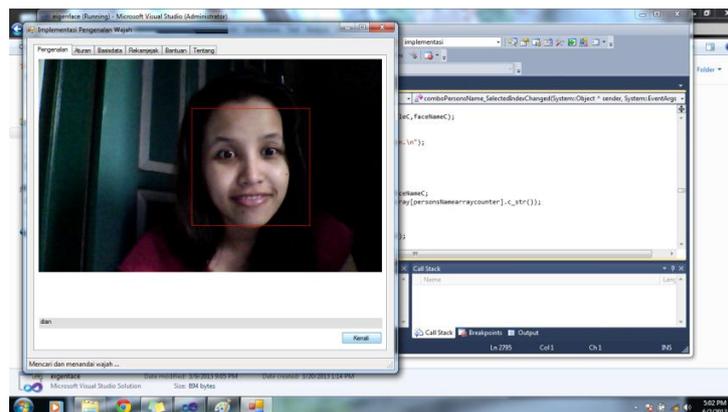
Gambar wajah yang diseleksi harus dilatih dengan menekan tombol latih agar didapatkan informasi wajah yang dibutuhkan untuk proses pengenalan. Setelah semua wajah yang berada pada basisdata dilatih, pengguna dapat memulai mode pengenalan dengan mengganti mode menjadi mode pengenalan. Apabila mode pengenalan dipilih, tab pengenalan berubah seperti yang ditunjukkan Gambar 6. Proses pengenalan diproses dengan menekan tombol kenali.

Setelah tombol kenali ditekan, output nama pengguna hasil pengenalan wajah ditunjukkan pada kolom edit teks yang sebelumnya kosong.

Tahapan pengujian pengenalan menggunakan *eigenface* ditunjukkan pada Gambar 6 dan Gambar 7.



Gambar 6 Tampilan sebelum proses penenalan



Gambar 7 Tampilan proses pengenalan yang benar

3. HASIL DAN PEMBAHASAN

Hasil pengujian terhadap beberapa sampel. Hasil pengujian berupa akurasi program pengenalan wajah menggunakan metode *eigenface* terhadap sampel yang diujikan dan tingkat keakurasian pengenalan serta waktu proses pengujian. Akurasi metode *eigenface* untuk pengenalan wajah merupakan perbandingan jumlah sampel yang berhasil dikenali dengan benar terhadap semua jumlah sampel yang diproses. Hasil pengujian metode *eigenface* untuk pengenalan wajah ditunjukkan Tabel 1 dan Tabel 2.

Berdasarkan Tabel 1 dan Tabel 2 didapatkan 53 kali hasil benar dan 11 kali hasil salah. Prosentase tingkat keberhasilan dari program pengenalan menggunakan *eigenface* yaitu:

$$\begin{aligned} \% &= \frac{\text{hasil pengujian benar}}{\text{total pengujian}} \times 100\% \\ &= \frac{53}{64} \times 100\% = 82,81\% \end{aligned} \quad (4)$$

Tabel 1 Hasil Pengujian Sampel Tanpa Ekspresi

No	Sampel Orang	1	Waktu (menit)	2	Waktu (menit)	3	Waktu (menit)	4	Waktu (menit)
1.	Dian	√	1	√	1	√	1	√	2
2.	Fifi	√	2	√	1	√	2	√	2
3.	Gita	x	3	√	2	√	3	√	2
4.	Jefry	√	2	√	2	x	3	x	2
5.	Lina	√	1	√	2	√	1	√	2
6.	Mb ika	√	1	√	1	√	1	√	1
7.	Memmy	x	2	√	2	x	2	√	1
8.	Nindy	x	1	x	2	x	2	√	2

Tabel 2 Hasil Pengujian Sampel Ekspresi Senyum

No	Sampel Orang	1	Waktu (menit)	2	Waktu (menit)	3	Waktu (menit)	4	Waktu (menit)
1.	Dian	√	1	√	1	√	1	√	2
2.	Fifi	√	1	√	2	√	2	√	1
3.	Gita	√	3	√	2	√	2	√	3
4.	Jefry	√	2	√	2	√	2	√	2
5.	Lina	√	2	x	1	x	1	x	2
6.	Mb ika	√	2	√	1	√	2	√	2
7.	Memmy	√	2	√	1	√	2	√	2
8.	Nindy	√	1	√	1	√	2	√	2

Berdasarkan hasil pengujian terhadap 16 sampel yang ditunjukkan pada Tabel 1 dan Tabel 2. Sampel terdiri dari 8 orang dengan variasi dua yaitu tanpa ekspresi dan ekspresi senyum. Masing-masing sampel orang memiliki 20 wajah di dalam *database* wajah. Pengujian dilakukan sebanyak 8 kali untuk tiap sampel orang yang terdiri dari 4 kali pengujian dengan pose tanpa ekspresi dan 4 kali pengujian dengan pose ekspresi senyum. Hasil pengujian menunjukkan prosentase tingkat keberhasilan yaitu 82,81%.

Pengujian menggunakan dua variasi tersebut menghasilkan data yang berbeda. Berdasarkan Tabel 1 dan Tabel 2 dan gambar foto yang tersimpan pada *database* didapatkan korelasi antara jumlah foto tiap pose dan keberhasilan pengenalan pada tiap pose. Data banyaknya gambar menggunakan ekspresi senyum dan tanpa ekspresi pada sampel orang yang mengalami kesalahan identifikasi dapat dilihat pada Tabel 3. Korelasi antara Tabel 1, Tabel 2 dan Tabel 3 adalah semakin banyak wajah dengan ekspresi senyum pada database akan memperbanyak kebenaran pengujian pada ekspresi senyum, begitu pula sebaliknya.

Tabel 3 Data Gambar Wajah Sampel Orang yang Salah Identifikasi

Sampel	Jumlah Wajah di <i>database</i>		Identifikasi Salah	
	Tanpa Ekspresi	Ekspresi Senyum	Tanpa Ekspresi	Ekspresi Senyum
Dian	8	12	0	0
Fifi	10	10	0	0
Gita	5	15	1	0
Jefry	8	12	2	0
Lina	4	16	0	3
Memmy	16	4	2	0
Nindya	4	16	3	0

Beberapa variasi keadaan pada saat melakukan pengujian menyebabkan hasil pengujian yang berbeda. Berdasarkan pengujian, ada faktor lain yang mempengaruhi hasil pengenalan wajah, yaitu kondisi lingkungan dan kondisi pemroses program. Kondisi lingkungan yang mempengaruhi yaitu pencahayaan dan jarak wajah dengan *webcam*. Berdasarkan pengujian yang dilakukan pada sampel bernama Jefry, pose tanpa ekspresi yang terakhir dilakukan pada saat pencahayaan berkurang menyebabkan adanya salah identifikasi sebanyak dua kali.

Faktor jarak wajah dengan *webcam* merupakan salah satu penyebab kesalahan pengenalan selain pencahayaan. Berdasarkan pengujian yang dilakukan pada sampel bernama Memmy, pada saat pengambilan gambar wajah terlalu dekat hasil identifikasi menjadi salah. Jarak minimal wajah terhadap *webcam* yaitu 20cm.

Kondisi pemroses program merupakan hal yang tidak dapat dihindarkan dalam program pengenalan wajah. Performa komputer yang digunakan dapat mempengaruhi lama proses pengenalan wajah. Berdasarkan pengujian dan literatur yang ada, semakin tinggi spesifikasi komputer yang digunakan akan mempercepat pembacaan data pada proses pengenalan wajah. Apabila jumlah nama dan wajah dalam *database* terlalu banyak, maka *file* *datawajah.xml* yang dihasilkan semakin besar. Berdasarkan pengujian yang dilakukan menggunakan 10 orang menghasilkan proses pengenalan yang lebih lama daripada menggunakan jumlah orang yang lebih sedikit. *File* *datawajah.xml* untuk delapan orang dan 20 wajah tiap orang menghasilkan ukuran 1,07GB. Maka dari itu proses pengenalan wajah dapat mencapai waktu kurang lebih dua menit untuk sekali proses pengenalan. Waktu pengenalan menjadi bertambah seiring bertambahnya sampel yang menjadikan jumlah wajah dalam *database* menjadi lebih besar. Hal tersebut terbukti ketika melakukan proses pengenalan pada saat *database* terisi 10 orang, lama satu kali proses pengenalan mencapai lebih dari tiga menit.

4. KESIMPULAN

Pada uji pengenalan yang dilakukan, hasil pengujian menunjukkan dari 16 sampel yang diujikan sebanyak 64 kali pengenalan menghasilkan 53 pengenalan benar dan 11 pengenalan salah. Berdasarkan hasil pengujian, data wajah sejumlah 160 wajah dari delapan orang memerlukan waktu kurang lebih dua menit pada setiap satu kali proses pengenalan. Prosentase keberhasilan pengenalan wajah pada sampel yang dilakukan adalah 82,81%. Faktor-faktor yang mempengaruhi kebenaran pengenalan antara lain yaitu kesesuaian ekspresi antara data di basis data dan data yang dikenali, faktor pencahayaan dimanan apabila pencahayaan semakin terang maka pengenalan semakin baik dan jarak terdekat wajah dengan *webcam* adalah 25cm.

5. SARAN

Penelitian ini masih jauh dari sempurna, maka saran untuk pengembangan lebih lanjut antara lain:

1. Penelitian selanjutnya menambahkan pemrosesan awal untuk mempertahankan informasi citra pada kondisi gelap.
2. Penelitian selanjutnya dapat memperkecil ukuran data pada *database* sehingga proses pembacaan data lebih cepat.
3. Penelitian selanjutnya menambahkan citra wajah pelatihan untuk menambah keakuratan proses pengenalan.

DAFTAR PUSTAKA

- [1] Shervin, 2010, *Introduction to Face Detection and Face Recognition*, <http://www.shervinemami.info/faceRecognition.html> diakses tanggal 23 Mei 2012 jam 9.29
- [2] Lyon, Douglass., Vincent. N. 2009. *Interactive Embedded Face Recognition*. Object Technology 8:23-25
- [3] Munir. R. 2004. *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*. Informatika. Bandung
- [4] Lim, Resmana., Raymond., Kartika. G. 2002. *Face Recognition Menggunakan Metode. Linear Discriminant Analysis (LDA)*. Proceeding Komputer dan Sistem Intelijen. Jakarta 21-22 Agustus 2002