

# PERBANDINGAN KINERJA METODE PRINCIPAL COMPONENT ANALYSIS (PCA) DAN NEURAL NETWORK (NN) PADA PENGENALAN WAJAH

Khoirul Anam<sup>(1)</sup>, Anang Aris Widodo<sup>(2)</sup>, Sultoni<sup>(3)</sup>  
Jurusan Teknik Informatika, FTI Universitas Merdeka Pasuruan  
Jl. Ir Juanda No.68 Pasuruan - Jawa Timur

e-mail:

<sup>(1)</sup>khoirulanam837@gmail.com, <sup>(2)</sup>anangariswidodo@gmail.com, <sup>(3)</sup>sultan.asmas@gmail.com

## ABSTRAK

*Deteksi wajah adalah salah satu langkah preprocessing yang paling penting pada sistem pengenalan wajah yang digunakan dalam identifikasi biometrik. Biometrik adalah kemampuan verifikasi dan identifikasi berdasarkan karakteristik fisik dan perilaku manusia. Sedangkan sistem pengenalan wajah adalah contoh biometrik didasarkan pada karakteristik fisik yang dapat membantu sistem komputer untuk memiliki kemampuan mirip dengan manusia.*

*Pada penelitian ini perbandingan kinerja metode PCA dan NN pada pengenalan wajah, algoritma yang digunakan dalam metode PCA adalah algoritma eigenfaces sedangkan algoritma yang digunakan dalam metode NN adalah algoritma backpropagation. Dengan perbandingan akurasi dan waktu pada kedua metode tersebut menunjukkan rata-rata 23.753% untuk akurasi dan 35.363 detik untuk waktu.*

**Kata Kunci** - Deteksi Wajah, Biometrik, Pengenalan Wajah, Principal Component Analysis, Neural Network.

## ABSTRACT

*Face detection is one of the most important preprocessing step in facial recognition systems that are used in biometric identification. Biometric verification and identification is based on the ability of physical characteristics and human behavior. While facial recognition system is the biometric sample is based on physical characteristics that can help a computer system to have the capability similar to humans.*

*In this study, comparison of the performance of PCA and NN on face recognition, the algorithm used in the method of PCA is eigenfaces algorithm while the algorithms used in the NN method is backpropagation algorithm. By comparison accuracy and time on both methods show an average of 23,753% for accuracy and 35 363 seconds to time.*

**Keywords** - Face Detection, Biometrics, Face Recognition, Principal Component Analysis, Neural Network.

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Mengenali wajah seseorang merupakan suatu hal yang mudah dilakukan oleh manusia. Seseorang akan cepat mengenal, menghafal dan membedakan wajah setiap orang yang dikenal sebelumnya walaupun dengan ekspresi yang berbeda-beda ketika ditemui, atau bahkan dalam keadaan terang maupun gelap [1].

Pengenalan wajah mempunyai dua aplikasi utama yaitu: verifikasi dan identifikasi. Verifikasi semata-mata mencocokkan data baru seseorang dengan data yang ada di database (*one to one*) dan umumnya menghasilkan dua keadaan yaitu true atau false. Sedangkan identifikasi mengenali seseorang dengan keputusan berdasarkan tingkat kedekatan atau kemiripan. Kemampuan mengukur karakteristik fisik atau perilaku yang dapat digunakan untuk memverifikasi atau mengidentifikasi seseorang disebut dengan biometrik [2].

Penelitian tentang pengenalan wajah sudah banyak dilakukan dengan berbagai macam metode, tetapi pada penelitian ini penulis ingin meneliti dengan membandingkan dua metode yang berbeda. Adapun metode pengenalan wajah yang dipakai dalam penelitian ini menggunakan metode Principal Component Analysis dan Neural Network sebagai metode pembelajaran dari sistem.

Alasan penulis melakukan penelitian ini karena orientasi kebutuhan terhadap sistem pengenalan wajah akan sangat besar dimasa yang akan datang, melihat dari sejarah pokok persoalan dari pengenalan wajah memiliki usia yang sama dengan visi komputer, dikarenakan arti penting praktis dari topik dan perhatian teoritis dari sejumlah ilmuwan. Dari sistem pengenalan wajah paling awal yang terkenal adalah Kohonen. Sistem ini tidak sepenuhnya berhasil, dikarenakan kebutuhan untuk pelurusan dan normalisasi yang kurang akurat. Kemudian, Kirby dan Sirovich (1989) memperkenalkan suatu manipulasi aljabar yang memudahkan untuk menghitung *eigenfaces* secara langsung. Selanjutnya, Turk dan Pentland (1991) menunjukkan kesalahan ketika mengkodekan *eigenfaces* dapat digunakan untuk mendeteksi wajah pada latar belakang yang rumit dan menentukan lokasi dengan skala yang tepat dari wajah. Mereka kemudian menggabungkan metode ini untuk mendeteksi dan menentukan lokasi dari wajah dengan metode pengenalan *eigenfaces* untuk mencapai pengenalan wajah yang akurat dan *real-time* pada keadaan lingkungan yang minimal. Sementara baru-baru ini, banyak yang mengalihkan perhatiannya pada database besar, multi pose, pengenalan wajah *real-time* dan praktis, dengan kondisi yang kurang pada citra [4]. Kemungkinan untuk masa berikutnya dengan kondisi wajah yang berbeda dan aksesoris tambahan pada wajah.

## 2. METODE PENELITIAN

### a. Analisa Komponen Utama (*Principal Component Analysis*)

*Principal Component Analysis* atau sering disebut PCA merupakan teknik linier reduksi menggunakan teori-teori sederhana dari statistik [5]. Teori statistik sederhana tersebut adalah varian, standart deviasi, zero mean,

kovarian dan persamaan karakteristik.

PCA merupakan teknik linier untuk memproyeksikan data vektor yang berdimensi tinggi ke vektor yang mempunyai dimensi lebih rendah [5]. PCA lebih banyak digunakan untuk keperluan ekstraksi fitur citra, dimana jumlah dimensi dari citra jauh lebih besar dibandingkan dengan jumlah data sampel yang digunakan. Untuk melakukan proyeksi sampel vektor dari citra pelatihan, setiap citra pelatihan disusun dalam bentuk vektor baris. Jika jumlah data pelatihan adalah sebanyak  $m$ , maka dimensi vektornya adalah  $m \times n$ . Apabila vektor citra pelatihan mempunyai dimensi  $m \times n$  tersebut diortogonalisasi dengan menentukan *eigenvector* dan *eigenvalue*, maka dimensinya akan berubah menjadi  $m \times n$ , dimana  $m \ll n$ . Pengurangan dimensi yang sangat signifikan ini akan sangat membantu untuk mempercepat waktu komputasinya dalam melakukan klasifikasi. Namun sebelum klasifikasi, perlu dilakukan penyeleksian *eigenface*.

### **b. Jaringan Syaraf Tiruan (Neural Network)**

Jaringan Syaraf secara biologis adalah merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program computer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran [3].

Setiap sel syaraf (*neuron*) akan memiliki satu inti sel, inti sel ini nanti yang akan bertugas untuk melakukan pemrosesan informasi. Informasi yang datang akan diterima oleh dendrit. Selain menerima informasi, dendrit juga menyertai aon sebagai keluaran dari suatu pemrosesan informasi. Informasi hasil olahan ini akan menjadi masukan bagi *neuron* lain yang mana antardendrit keduanya tersebut dipertemukan dengan *synapsis*. Informasi yang dikirimkan antar *neuron* ini berupa rangsangan yang dilewatkan melalui dendrit. Informasi yang datang dan diterima oleh dendrite akhir yang bersentuhan dengan dendrit dari *neuron* yang lain. Informasi ini akan diterima oleh *neuron* lain jika memenuhi batasan tertentu, yang sering dikenal dengan nama nilai ambang (*threshold*) yang dikatakan teraktivasi.

## **2.1 Rumus**

### **2.1.1 Rumus Metode PCA**

mencari nilai rata-rata  $x_1, x_2, x_3$ :

$$\begin{aligned}\mu_1 &= \frac{\text{total nilai } x_1}{\text{jumlah data}} \\ \mu_2 &= \frac{\text{total nilai } x_2}{\text{jumlah data}} \\ \mu_3 &= \frac{\text{total nilai } x_3}{\text{jumlah data}}\end{aligned}$$

(2.1.1)

Nilai data dikurangi rata-rata masing-masing parameter :

$$(x_1 - \mu_1), (x_2 - \mu_2), (x_3 - \mu_3) \quad (2.1.2)$$

Perkalian kombinasi dari hasil nilai data dikurangi parameter masing-masing :

$$(x_1 - \mu_1)(x_1 - \mu_1), (x_1 - \mu_1)(x_2 - \mu_2), (x_1 - \mu_1)(x_3 - \mu_3), (x_2 - \mu_2)(x_2 - \mu_2), (x_2 - \mu_2)(x_3 - \mu_3), (x_3 - \mu_3)(x_3 - \mu_3) \quad (2.1.3)$$

Nilai matrik kovarian masing-masing :

$$\begin{aligned} c(x_1, x_1) &= \frac{\text{total perkalian kombinasi } x_1, x_1}{\text{jumlah data} - 1} \\ c(x_1, x_2) &= \frac{\text{total perkalian kombinasi } x_1, x_2}{\text{jumlah data} - 1} \\ c(x_1, x_3) &= \frac{\text{total perkalian kombinasi } x_1, x_3}{\text{jumlah data} - 1} \\ c(x_2, x_2) &= \frac{\text{total perkalian kombinasi } x_2, x_2}{\text{jumlah data} - 1} \\ c(x_2, x_3) &= \frac{\text{total perkalian kombinasi } x_2, x_3}{\text{jumlah data} - 1} \\ c(x_3, x_3) &= \frac{\text{total perkalian kombinasi } x_3, x_3}{\text{jumlah data} - 1} \end{aligned} \quad (2.1.4)$$

Maka matrik kovarian dapat disusun :

$$C = \begin{bmatrix} c(x_1, x_1) & c(x_1, x_2) & c(x_1, x_3) \\ c(x_1, x_2) & c(x_2, x_2) & c(x_2, x_3) \\ c(x_1, x_3) & c(x_2, x_3) & c(x_3, x_3) \end{bmatrix} \quad (2.1.5)$$

Untuk mencari nilai eigen( $\lambda$ ) akan digunakan polinomial karakteristik dan persamaan karakteristik :

$$f(\lambda) \det(C - \lambda I) \quad (2.1.6)$$

Kemudian nilai eigen dapat dihitung lewat persamaan karakteristik :

$$f(\lambda) = 0 \quad (2.1.7)$$

Dan melakukan substitusi nilai Eigen :

$$(C - \lambda_1 I)x = 0 \quad (2.1.8)$$

Vektor eigen untuk masing - masing nilai eigen kemudian dapat ditentukan dengan melakukan operasi baris elementer atau teknik eliminasi sistem persamaan linear :

$$(\lambda I - C)v = 0$$

(2.1.9)

### 2.1.2. Metode NN

Masukkan maksimal epoch, target *error*, dan *learningrate* untuk langkah awal inialisasi bobot.

Kemudian akan diteruskan dengan umpan maju (*feedforward*) :

$$z_{in_j} = b1_j + \sum_{i=1}^n x_i v_{ij}$$

(2.1.10)

Menggunakan fungsi aktivasi untuk menghitung sinyal outputnya:

$$z_j = f(z_{in_j})$$

(2.1.11)

Dan tiap-tiap unit output ( $y_k, k = 1, 2, 3, \dots, m$ ) menjumlahkan sinyal-sinyal output terbobot:

$$y_{in_k} = b2_k + \sum_{i=1}^p z_i w_{jk}$$

(2.1.12)

Menggunakan fungsi aktivasi untuk menghitung sinyal outputnya:

$$y_k = f(y_{in_k})$$

(2.1.13)

Selanjutnya hasil dari perhitungan *feedforward* akan dihitung error (*backpropagation*):

$$\begin{aligned} \delta 2_k &= (t_k - y_k) f'(y_{in_k}) \\ \varphi 2_{jk} &= \delta_k z_j \\ \beta 2_k &= \delta_k \end{aligned}$$

(2.1.14)

Kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memprediksi nilai  $w_{jk}$ ):

$$\Delta w_{jk} = \alpha \varphi_{jk}$$

(2.1.15)

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $\beta 2_k$ ):

$$\Delta \beta 2_k = \alpha \beta_k$$

(2.1.16)

Selanjutnya tiap-tiap unit tersembunyi ( $z_j, j = 1, 2, 3, \dots, p$ ) menjumlahkan delta inputnya:

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.1.17)$$

Kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi error:

$$\begin{aligned} \delta 1_j &= \delta_{in_j} f'(z_{in_j}) \\ \varphi 1_{ij} &= \delta_j x_j \\ \beta 1_j &= \delta 1_j \end{aligned} \quad (2.1.18)$$

Kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{ij}$ ):

$$\Delta v_{ij} = \alpha \varphi 1_{ij} \quad (2.1.19)$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $\beta 1_j$ ):

$$\Delta \beta 1_j = \alpha \varphi 1 \quad (2.1.20)$$

Selanjutnya tiap-tiap unit output ( $y_k, k = 1, 2, 3, \dots, m$ ) memperbaiki bias dan bobotnya ( $j = 0, 1, 2, \dots, p$ ):

$$\begin{aligned} w_{jk}(\text{baru}) &= w_{jk}(\text{lama}) + \Delta w_{jk} \\ \beta 2_k(\text{baru}) &= \beta 2_k(\text{lama}) + \Delta \beta 2_k \end{aligned} \quad (2.1.21)$$

Tiap-tiap unit tersembunyi ( $z_j, j = 1, 2, 3, \dots, p$ ) memperbaiki bias dan bobotnya ( $i = 0, 1, 2, \dots, n$ ):

$$\begin{aligned} v_{ij}(\text{baru}) &= v_{ij}(\text{lama}) + \Delta v_{ij} \\ \beta 1_j(\text{baru}) &= \beta 1_j(\text{lama}) + \Delta \beta 1_j \end{aligned} \quad (2.1.22)$$

Kemudian perubahan bobot dari hasil perhitungan *feedforward* dan *backpropagation* akan dihitung MSE.

Inisialisasi bobot awal secara random dengan nilai awal -0.5 sampai 0.5 (atau 1 sampai 1, atau interval yang lainnya). Jika hasil perhitungan MSE kurang dari target error, maka akan simpan bobot.

### 3. HASIL IMPLEMENTASI DAN PEMBAHASAN

#### 3.1 Kebutuhan Sistem

Dalam implementasi aplikasi pada Pengenalan Wajah ini dibangun dengan beberapa perangkat keras dan perangkat lunak. Berikut adalah spesifikasi perangkat keras dan perangkat lunak yang direkomendasikan.

##### 3.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang memadai sangat dibutuhkan untuk mendukung kinerja sistem, dimulai pembangunan sampai dengan masa penggunaan sistem adalah sebagai berikut :

- ✓ Processor Intel atau AMD x86 yang mendukung SSE2.
- ✓ Memory (RAM) 1 GB atau lebih.
- ✓ Minimal spasi Hard Disk (kosong) 10 GB untuk MATLAB untuk instalasi.

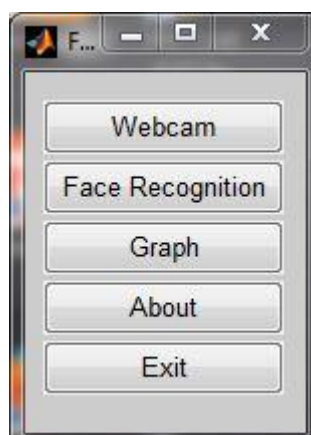
##### 3.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak memiliki peranan penting dalam mengembangkan sistem terutama untuk implementasi kode program. Perangkat lunak yang digunakan untuk membangun aplikasi pada Pengenalan Wajah ini adalah sebagai berikut :

- ✓ Sistem Operasi Windows XP/7/8/8.1/10
- ✓ MATLAB R2014a atau yang lebih baru.

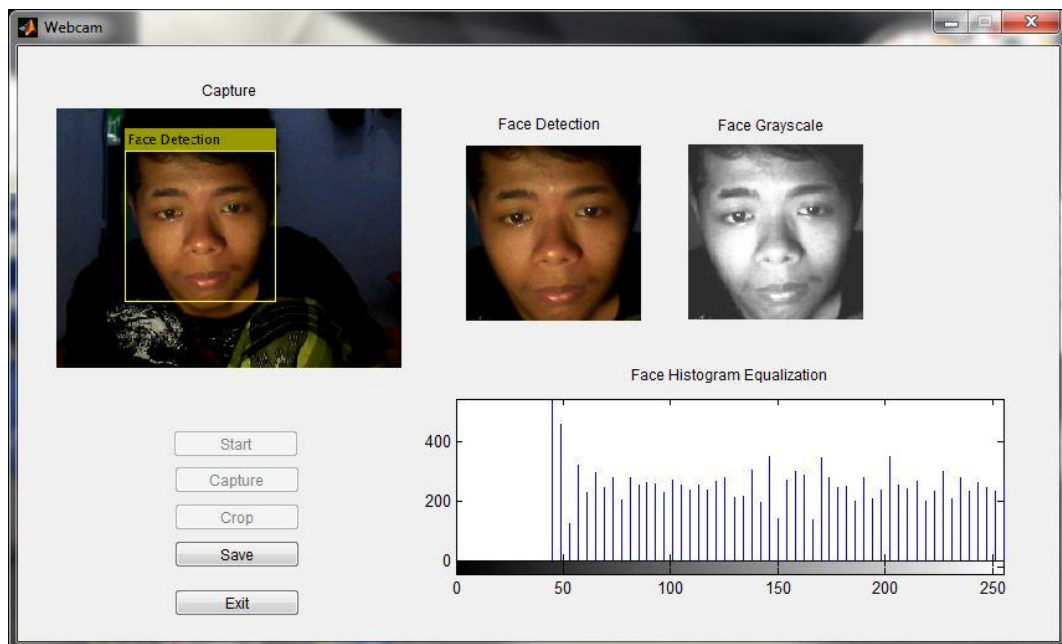
#### 3.2 Design dan Tahap Implementasi

- a. Pertama kali ketika program dijalankan akan menampilkan form menu.



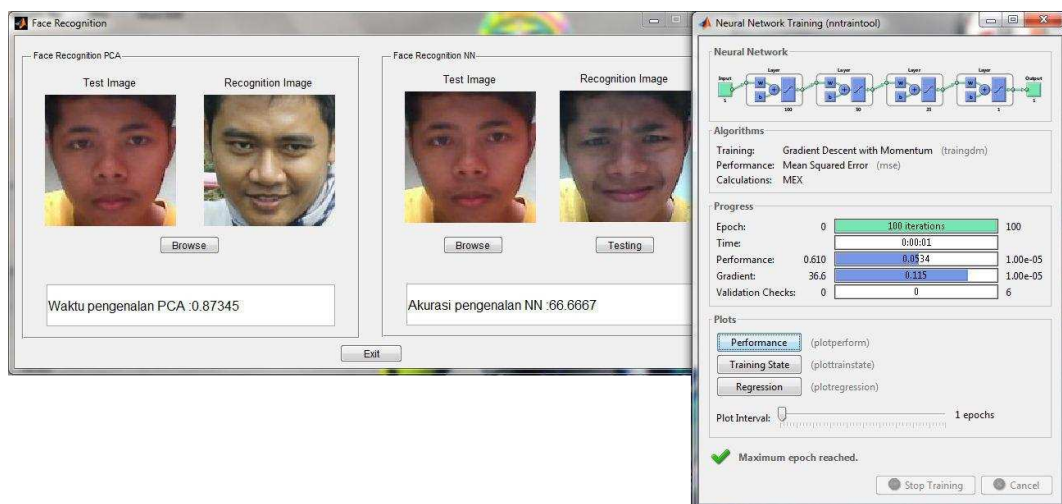
Gambar 1 Tampilan form menu.

- b. Ketika tombol “Webcam” ditekan akan menampilkan form webcam yang digunakan untuk mengambil data citra wajah secara *realtime*.



Gambar 2 Tampilan form webcam saat pengambilan citra *realtime*.

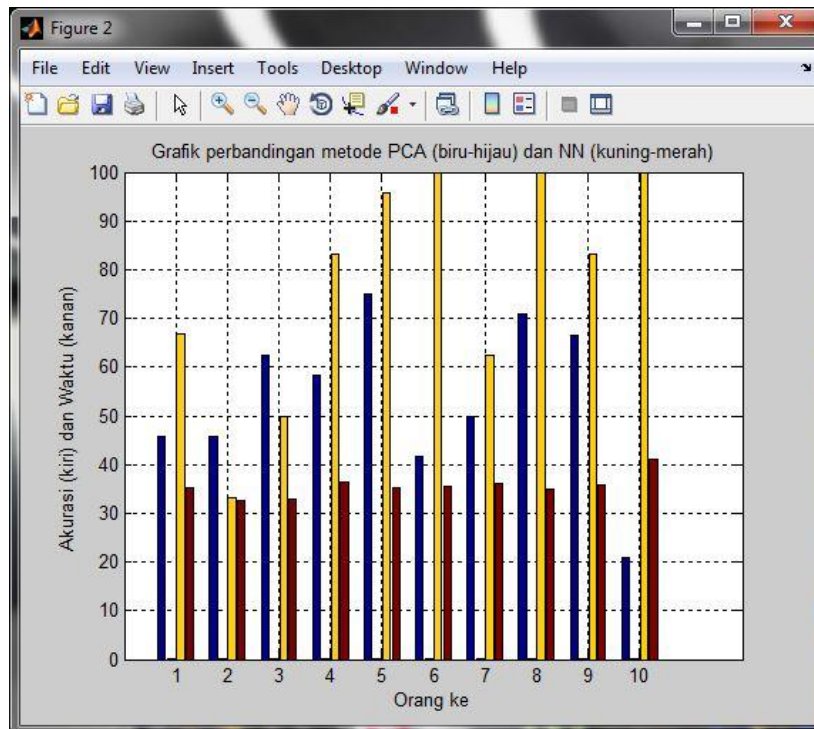
- c. Selanjutnya jika tombol “Face Recognition” ditekan akan menampilkan form face recognition yang digunakan untuk melakukan pengenalan wajah pada 2 metode sekaligus, yaitu metode PCA dan NN.



Gambar 3 Tampilan form face recognition saat pengenalan wajah dengan 2 metode.



- d. Selanjutnya jika tombol “Graph” akan menampilkan grafik untuk perbandingan antara metode PCA dan NN.



Gambar 4 Tampilan grafik perbandingan antara metode PCA dan NN.

- e. Jika tombol “About” ditekan akan menampilkan si pembuat program.



Gambar 5 Tampilan form about.

### 3.3 Data Citra

Tabel 1 Sampel data citra *realtime*.

No	Orang ke	Nama File	Citra
1	1	1.jpg	
2	2	11.jpg	
3	3	21.jpg	
4	4	31.jpg	
5	5	41.jpg	
6	6	51.jpg	
7	7	61.jpg	
8	8	71.jpg	
9	9	81.jpg	
10	10	91.jpg	

### 3.4 Pembahasan

Dalam tahap ujicoba ini pengenalan wajah dilakukan dengan menggunakan pose 2,4,6,8,9,10 sebagai data *training* dan pose 1,3,5,7 sebagai data *testing*. Dengan 10 orang dan tiap orang mempunyai 1 pose citra yang berbeda, jadi total seluruhnya 100 citra. Dan khusus untuk metode NN dengan ketetapan maksimal *epoch* 1000, target *error* 0,00001, *learning rate* 0,01 dan 4 layer dengan layer pertama 500 layer kedua 50 layer ketiga 25 layer keempat 1. Berikut adalah hasil ujicobanya :

#### a. Pengenalan Wajah metode PCA

Tabel 3.2 Hasil ujicoba pengenalan wajah metode PCA.

No	Orang ke	Hasil Ujicoba	
		Akurasi	Waktu
1	1	45.83%	0.228346 detik
2	2	45.83%	0.244686 detik
3	3	62.5%	0.206594 detik
4	4	58.33%	0.206855 detik
5	5	75%	0.21044 detik
6	6	41.66%	0.198112 detik
7	7	50%	0.212915 detik
8	8	70.83%	0.200453 detik
9	9	66.66%	0.200737 detik
10	10	20.83%	0.210425 detik
	Rata-rata	53,75%	0.211956 detik

#### b. Pengenalan Wajah metode NN

Tabel 3.3 Hasil ujicoba pengenalan wajah metode NN.

No	Orang ke	Hasil Ujicoba	
		Akurasi	Waktu
1	1	66.6667%	35.25 detik
2	2	33.3333%	32.5 detik
3	3	50%	33 detik
4	4	83.333325%	36.5 detik
5	5	95.833325%	35.25 detik
6	6	100%	35.5 detik
7	7	62.500025%	36 detik
8	8	100%	35 detik
9	9	83.333325%	35.75 detik
10	10	100%	41 detik
	Rata-rata	77.50%	35.575 detik

Berdasarkan hasil ujicoba sebagaimana telah diuraikan di atas, maka dapat kita analisis sebagai berikut; dari segi akurasi metode NN lebih baik

dalam melakukan pengenalan yang mencapai 77,50% apabila dibandingkan dengan metode PCA yang hanya memiliki akurasi 53,75%, ini dikarenakan metode NN melakukan pembelajaran secara kontinyu baik maju maupun mundur, dimana prinsip kerja dari NN adalah meniru prinsip kerja dari otak manusia, sehingga semakin banyak melakukan pembelajaran, maka akan cepat mengenali. Namun apabila dilihat dari waktu komputasi, metode NN masih cukup lambat apabila dibandingkan dengan metode PCA (lihat tabel 4.2 dan 4.3 di atas) ini dikarenakan metode NN akan melakukan pembelajaran dari awal setiap ada inputan atau data baru, sedangkan PCA tidak melakukan pembelajaran dari awal setiap ada data baru, hanya data atau inputan baru saja yang dilakukan perhitungan oleh metode PCA sehingga akan mempercepat waktu komputasi.

Untuk mengurangi atau meminimalisir kelemahan kelemahan yang dimiliki oleh masing – masing metode, maka dapat dilakukan penelitian lanjutan salah satunya dengan menggabungkan metode PCA dengan NN sehingga akan tercipta algoritma baru, yaitu dengan memanfaatkan kelebihan dari masing – masing metode. Dimana metode PCA memiliki kelebihan dalam hal waktu komputasi dan kurang dalam hal akurasi, sedangkan metode NN memiliki akurasi yang lebih baik, namun lambat dalam waktu komputasi. Dengan menggabungkan kelebihan – kelebihan tersebut, maka akan tercipta algoritma baru yang dapat meningkatkan akurasi dan waktu komputasi.

Penelitian lanjutan juga bisa dilakukan untuk tujuan lain, misalnya untuk masuk ke ruang rahasia, untuk absensi kuliah maupun absensi mengajar dosen. Karena penelitian pengenalan wajah bisa dimanfaatkan dibanyak bidang misalnya robotic, autentikasi, kedokteran dll.

#### 4. KESIMPULAN

Pada penelitian ini perbandingan kinerja metode PCA dan NN pada pengenalan wajah, algoritma yang digunakan dalam metode PCA adalah *algoritmaeigenfaces* sedangkan algoritma yang digunakan dalam metode NN adalah *algoritma backpropagation*. Dengan perbandingan akurasi dan waktu menunjukkan rata-rata 23.75% untuk akurasi dan 35.3 detik untuk waktu.

Berdasarkan tujuan penelitian dari hasil analisis pada kedua metode tersebut, maka dapat diambil kesimpulan sebagai berikut :

##### 1) Kelebihan :

###### a. Metode PCA :

- Dapat digunakan untuk segala kondisi data / penelitian, terutama pada penelitian *pattern recognition*.
- Walaupun metode ini memiliki tingkat kesulitan yang tinggi akan tetapi kesimpulan yang diberikan cukup akurat.
- Dapat mereduksi atau mengurangi informasi data yang besar dari sebuah citra tanpa menghilangkan informasi yang ada pada sebuah citra tersebut.

###### b. Metode NN :

- Dapat melakukan tugas-tugas yang program linear tidak bisa (non-linear).

- Mempunyai kemampuan untuk melakukan sesuatu kegiatan yang didasarkan atas data yang diberikan pada saat pembelajaran.
- Dapat membuat organisasi sendiri atau me-representasikan informasi yang didapat pada saat pembelajaran.
- Dapat menghasilkan perhitungan parallel.

## 2) Kekurangan :

### a. Metode PCA :

- Sebuah metode yang linear, sehingga akan menemui kesulitan jika berhadapan dengan data-data non-linear.
- Masih membutuhkan proses merubah citra 2 dimensi ke citra 1 dimensi untuk mendekati kemiripan, tidak langsung seperti metode NN.

### b. Metode NN :

- Kurang mampu untuk melakukan operasi-operasi numerik dengan perbedaan nilai yang tinggi.
- Kurang mampu melakukan operasi algoritma aritmatik, operator logika dan simbolis.
- Lamanya proses *training* yang mungkin terjadi dalam waktu yang sangat lama.

Karena aplikasi yang digunakan pada penelitian ini hanyalah simulasi, maka terdapat kekurangan dari aplikasi diantaranya :

- a. Keterbatasan kapasitas webcam yang dipakai oleh penulis adalah 0.3 MP dengan resolusi 640x480 piksel, sehingga sering terjadi kesalahan dalam pengenalan citra yang dimaksud dengan citra yang sebelumnya.
- b. Perubahan latar (*background*) yang terlalu banyak mencolok dan pencahayaan yang kurang akan mengurangi keakuratan pengenalan wajah.
- c. Sedikitnya data latih akan mempengaruhi akurasi dan komputasi pada pengenalan wajah, terutama pada metode Jaringan Syaraf Tiruan (NN).
- d. Data *training* untuk metode NN masih menggunakan inputan excel manual dari hasil perkalian luas matrik citra.
- e. Nilai akurasi metode PCA masih menggunakan perhitungan manual dengan rumus :  $\text{jumlah citra yang dikenali} \times 100 / \text{total citra yang diuji}$ .

## 5. SARAN

Setelah dilakukan analisis dan ujicoba, penelitian ini masih memiliki kekurangan yang dapat diperbaiki untuk penelitian selanjutnya. Maka dari itu penulis mengajukan saran, sebagai berikut :

- a. Perlu dilakukan pembelajaran dan pelatihan lebih banyak lagi, terutaman untuk metode NN agar pola Jaringan Syaraf Tiruan yang dikenali lebih mendekati target.
- b. Untuk pengembangan selanjutnya perlu diperbanyak data citra, karena pada penelitian ini hanya memiliki beberapa hasil citra *realtime*.
- c. Penelitian selanjutnya diharapkan adanya pengembangan, salah satunya dengan mengaplikasikan dengan menggunakan metode yang lainnya.

- d. Untuk metode PCA dapat membandingkan antara proses pengenalan berdasarkan jarak (*euclidean distance*) dengan menggunakan berdasarkan warna kulit, mana yang lebih baik dari segi akurasi dan waktu.
- e. Data *training* khusus untuk metode NN masih menggunakan inputan excel manual dari hasil luas matrik citra, maka dari itu untuk pengembangan selanjutnya diharapkan agar bisa otomatis ketika simpan citra *training* akan langsung membentuk file khusus untuk *training* metode NN.
- f. Untuk penelitian selanjutnya diharapkan ada kategori pencahayaan redup, sedang dan cerah pada citra dan mungkin ditambahkan dengan tampilan yang menggunakan aksesoris tambahan seperti kacamata, topi dan sebagainya.

## DAFTAR PUSTAKA

- [1]Ahmad, U. (2005). *Pengolahan Citra Digital & Teknik Pemrogramannya*. Yogyakarta: Graha Ilmu.
- [2]Anil K. Jain, A. R. (2004). An Introduction to Biometric Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14, 4-20.
- [3]Fauset, L. (1994). *Fundamental of Neural Networks (Architectures, Algorithms, and Application)*. New Jersey: Prentice-Hall.
- [4]Goering, R. (2004, October 4). Matlab edges closer to electronic design automation world. *EE Times* .
- [5]Gonzalez, R. C. (2002). *Digital Image Processing*. Prentice Hall, Internasional Edtion Second Edition.
- [6]Hjelmas, E. L. (2001). Face Detection: A Survey. *Computer Vision and Image Understanding*, 83, 236-274.
- [7]Moghaddam, B. (n.d.). *Face Recognition*. Retrieved from <http://www-white.media.mit.edu/vismod/demos/facerec/index.html>
- [8]Osuna, E. F. (1997). Training Support Vector Machines: An Application to Face Detection. *Proc. IEEE Conf. Computer Vision and Pattern Recognition* .
- [9]Pärt-Enander, E. d. (1999). *The Matlab 5 Handbook*. Addison-Wesley.  
Proakis, J. d. (1996). *Digital Signal Processing: Principles, Algorithms and Applications*. Macmillan.
- [10]Rowley, H. B. (1998). Neural Network-Based Face Detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20 (1).
- [11]Sung, K. (1996). *Learning and Example Selection for Object and Pattern Detection*. Massachusetts Institute of Technology AI Lab.
- [12]Sung, K. P. (1994). *Example-Based Learning for View-Based Human Face Detection*. Massachusetts Institute of Technology AI Lab.
- [13]Turk, M. a. (1991). Face Recognition Using Eigenface. *Proc. IEEE Conf. of Computer Vision and Pattern Recognition*, 13, 586-591.
- [14]Yang, M. K. (2002). Detecting Faces in Images: A Survey. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24 (1).