

# Aplikasi Pengenalan Wajah Menggunakan Algoritma Backpropagation dan Voila Jones

Sayuti Rahman<sup>1\*</sup>, Emil Fitrianshah Aliff S<sup>2</sup>, Andi Marwan Elhanafi<sup>3</sup>

Address : Program Studi Teknik Informatika Universitas Harapan Medan

Email : [1masay.ram@gmail.com](mailto:1masay.ram@gmail.com), [2emilfitrianshah@gmail.com](mailto:2emilfitrianshah@gmail.com)

\* Corresponding author

## Abstrak

Saat ini, teknologi sistem keamanan semakin berkembang pesat. Para ilmuwan dibidang teknologi terus-menerus membuat ataupun menemukan cara baru untuk berupaya meningkatkan sistem keamanan pada komputer, terutama sistem keamanan dengan menggunakan metode-metode dari visi komputer seperti identifikasi ataupun berbentuk pengenalan suatu benda. Dengan perkembangan teknologi sekarang ini terutama dibidang visi komputer diharapkan mampu untuk menjawab seluruh tantangan dari sistem keamanan. Salah satu sistem keamanan dari visi komputer yang paling populer adalah sistem identifikasi atau pengenalan suatu objek atau gambar dengan menggunakan algoritma tertentu. Oleh karena itu, pada penelitian menerapkan sistem pengenalan wajah menggunakan algoritma backpropagation. Dimana, sebuah foto wajah seseorang akan dikenali berdasarkan jarak antar fitur-fitur wajah seperti: mata, hidung, dan mulut kemudian hasil dari temuan jarak fitur tersebut akan menjadi input layer pada algoritma backpropagation kemudian dengan memasukkan target output akan memunculkan identitas pada foto atau gambar tersebut. Dari hasil pengujian 10 gambar yang dilakukan, 8 gambar berhasil dikenali 2 gambar tidak berhasil, maka tingkat akurasi aplikasi tersebut adalah 80%

**Kata Kunci:** *Visi Komputer, Pengenalan wajah, Backpropagation,*

## Abstract

Currently, security system technology is growing rapidly. Scientists in the field of technology constantly create or find new ways to work to improve the security system on computers, especially security systems using the methods of computer vision such as identification or the form of recognition of an object. With the current technological developments, especially in the field of computer vision is expected to be able to answer all the challenges of the security system. One of the most popular computer security systems is the system of identification or recognition of an object or image using a particular algorithm. Therefore, the research applied face recognition system using backpropagation algorithm. Where, a photo of a person's face will be recognized based on the distance between facial features such as: eyes, nose, and mouth then the result of the feature distance finding will be the input layer of the backpropagation algorithm and then by entering the output target will bring up the identity of the photo or image. From the results of the 10 images tested, 8 images were identified by two unsuccessful images, the accuracy of the application was 80%.

**Key Word:** *Computer Vision, Face Recognition, Backpropagation*

## 1. Pendahuluan

Teknologi baru terus bermunculan dengan berbagai fitur canggih. Hal tersebut mempengaruhi teknologi pada sistem pengenalan (Biometrik). Salah satunya adalah pengenalan wajah. Teknologi pengenalan wajah sangat menarik perhatian dalam beberapa tahun terakhir karena berpotensi pada berbagai macam aplikasi, seperti untuk pengawasan dan keamanan, telekomunikasi, perpustakaan digital, dan interaksi manusia dan komputer. Hal ini karena wajah dapat memberikan informasi spesifik yang berhubungan dengan identifikasi pribadi [1]. Telah banyak penelitian dalam mengeksplorasi berbagai masalah dalam proses pengenalan wajah dan merancang pengenalan wajah manusia. Beberapa penelitian telah menggunakan jaringan saraf tiruan untuk mendapatkan variasi pendekatan seperti pencocokan garis wajah serta perbandingan ukuran jarak fitur wajah (hidung, mata, mulut) dari subjek gambar [2].

Salah satu jaringan saraf tiruan yang digunakan adalah algoritma *Backpropagation* yang dikenal sebagai bagian dari Jaringan Saraf Tiruan atau *Neural Network*. Untuk mengenali wajah seseorang dengan algoritma *Backpropagation* harus melalui beberapa tahapan, pertama adalah tahapan *preprocessing* dimana wajah harus dideteksi terlebih dahulu dengan menggunakan metode *viola jones* kemudian mengenali fitur wajah dengan metode yang sama, lalu *filter noise* dan *grayscale*. Setelah itu citra wajah harus melalui tahapan *thresholding* kemudian dilakukan perhitungan jarak antar fitur wajah dengan metode *Euclidean distance*. Jarak fitur wajah yang didapatkan akan langsung menjadi *input layer* pada algoritma *backpropagation*. Dengan menentukan target output maka dilakukan pelatihan, kemudian pengujian pada wajah. Dalam penelitian ini penulis merancang sebuah aplikasi dengan menerapkan metode *viola jones* sebagai metode dalam menemukan citra wajah dan juga menemukan fitur wajah seperti mata, hidung, dan mulut, kemudian dalam menghitung jarak menggunakan metode *euclidean distance*.

Penelitian sebelumnya sudah pernah dilakukan oleh Sigit Kusmaryanto dengan metode ekstraksi fitur dan juga Akhyar dengan menggunakan Facial Points.

## 2. Metodologi Penelitian

Metodologi penelitian yang dilakukan dalam penyelesaian penelitian ini meliputi:

1. Studi Literatur  
Pada tahap ini dilakukan dengan cara mencari informasi serta menggali pengetahuan yang berhubungan dengan penelitian ini baik melalui buku, jurnal, internet, dan juga dari sumber lainnya yang mendukung perancangan sistem berdasarkan landasan teori yang ada.
2. Pengumpulan Data  
Pada tahap ini dilakukan dengan cara mengambil foto-foto wajah seseorang baik itu dari internet atau dari sumber lainnya.
3. Desain dan Permodelan Sistem  
Pada tahap ini dibuat perancangan sistem pengenalan wajah yang akan dibangun yang terdiri dari tahapan *preprocessing*, pelatihan, dan pengujian
4. Simulasi  
Pada tahap ini sistem yang telah dirancang, ditransformasikan kedalam bentuk simulasi menggunakan Matlab 2016a, untuk mendukung analisis dari penyusunan tugas akhir berdasarkan data.
5. Pelatihan  
Pada tahap ini sistem yang telah dirancang akan dilakukan pelatihan dengan menggunakan algoritma *backpropagation* pada setiap data yang disimpan dan diinput
6. Pengujian  
Pada tahap ini dilakukan pengujian pada setiap data yang telah dilatih sebelumnya untuk melakukan pencocokan dan pengenalan pada setiap data.

### 2.1 Biometrika

Biometrika berarti mengukur karakteristik (*distinguishing traits*) pada tubuh atau perilaku seseorang yang berguna dalam melakukan pengenalan secara otomatis terhadap identitas orang tersebut, dengan membandingkannya

dengan karakteristik yang sebelumnya telah disimpan pada suatu *database*. Secara umum biometrika terbagi 2, yaitu karakteristik fisiologis atau fisik (*physiological/physical characteristic*) dan karakteristik perilaku (*behavioral characteristic*). Biometrika berdasarkan karakteristik fisiologis/fisik menggunakan bagian-bagian fisik dari tubuh seseorang, seperti DNA, telinga, wajah, geometri tangan, pembuluh tangan, sidik jari, iris, telapak tangan, retina, gigi dan bau (komposisi kimia) dari keringat tubuh. Sedangkan biometrik berdasarkan karakteristik perilaku menggunakan perilaku seseorang sebagai kode unik untuk melakukan pengenalan, seperti gaya berjalan, hentakan tombol, tanda tangan dan suara[3].

## 2.2 Wajah

Wajah atau *face* adalah bagian tubuh yang berada pada kepala bagian depan manusia yang meliputi wilayah dahi, hidung, alis, pipi, dan dagu. Setiap orang memiliki karakteristik wajah yang berbeda. Hal tersebut menjadi landasan bahwasannya wajah termasuk objek dari bidang biometrik. Biometrik adalah metode otomatisasi yang dapat mengenal seseorang berdasarkan fisik seperti mata, sidik jari, tanda tangan dan

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

suara.

Dalam aplikasi berbasis *desktop* komputer atau *device* lainnya pengenalan wajah dapat diproses dengan algoritma tertentu, agar diperoleh kualitas pengenalan wajah yang baik dibutuhkan suatu kondisi lingkungan yang mendukung untuk dilakukannya uji coba atau dipakainya sistem tersebut diantaranya dengan indeks cahaya, warna kulit, sudut-sudut pada saat pengambilan gambar wajah dan ukuran *pixel* kamera saat pengambilan gambar wajah berlangsung [4]

## 2.3 Biometrika Wajah

Sistem biometrika pengenalan wajah adalah sistem yang secara otomatis dapat mengenali seseorang dari sebuah gambar dan

video. Ada dua pedekatan identifikasi dalam pengenalan wajah. Pendekatan identifikasi bertujuan untuk mencari jawaban identitas wajah siapa orang tersebut. Sedangkan pendekatan verifikasi, bertujuan memastikan apakah benar wajah orang yang dimaksud, bukan orang lain[5].

## 2.4 Citra

Dilihat dari segi matematis, citra adalah fungsi yang secara terus-menerus berasal dari intensitas cahaya. Sumber cahaya menyinari objek, kemudian dipantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan tersebut ditangkap oleh alat-alat yang mempunyai *optic*, seperti mata pada manusia, kamera, pemindai (*scanner*), dan lain-lain, kemudian bayangan objek dalam bentuk citra dapat terekam. Ada dua kelompok citra yaitu citra diam (*still images*) dan citra bergerak (*moving images*). Citra diam adalah citra tunggal yang tidak bergerak, Sedangkan citra bergerak merupakan citra diam yang ditampilkan secara beruntun (*sekuensial*), sehingga memberi kesan pada mata sebagai gambar yang bergerak. Citra digital adalah sebuah fungsi 2D,  $f(x,y)$ , yang merupakan fungsi intensitas cahaya, dimana nilai  $x$  dan  $y$  merupakan koordinat spasial dan nilai fungsi di setiap titik  $(x,y)$  merupakan tingkat keabuan citra pada titik tersebut[3].

## 2.5 Pengertian Citra Digital

Citra digital dinyatakan dengan sebuah matriks dimana baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matriksnya (yang disebut sebagai elemen gambar atau piksel) menyatakan tingkat keabuan pada titik tersebut. Matriks dari citra digital berukuran  $N \times M$  (tinggi  $x$  lebar), dimana:

$N$  = jumlah baris  $0 < y = N - 1$

$M$  = jumlah kolom  $0 = x = M - 1$

$L$  = derajat keabuan  $0 = f(x,y) = L - 1$

Berikut ini adalah gambaran matriks dari citra digital:

**Gambar 1.** Matriks Citra Digit

Dimana indeks baris (x) dan indeks kolom (y) menyatakan suatu koordinat titik pada citra, sedangkan f(x,y) merupakan intensitas (derajat keabuan) pada titik (x,y).

## 2.6 Jenis Citra Digital

Adapun jenis-jenis citra sebagai berikut:

### 1. Greyscale

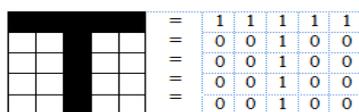
Kemungkinan warna hitam pada nilai minimal dan kemungkinan warna putih untuk nilai maksimal. Jumlah warna pada citra *Greyscale* bergantung pada jumlah bit yang telah disediakan oleh memori untuk menampung kebutuhan warna tersebut. Semakin besar jumlah bit warna yang disediakan di memori, maka semakin halus gradasi warna yang terbentuk.



**Gambar 2.** Citra *Greyscale*

### 2. Citra Biner

Dalam citra biner terdapat 2 buah warna, yaitu hitam dan putih. Untuk Warna hitam bernilai 1 dan putih bernilai 0. Agar dapat menyimpan kedua buah warna ini dibutuhkan 1 bit di memori. Contoh dari susunan piksel pada citra monokrom adalah sebagai berikut:



**Gambar 3.** Citra *Biner*

### 3. Citra Warna

Piksel yang terdapat pada citra warna mewakili kombinasi dari 3 warna dasar yaitu, merah, hijau, kuning (*red, green, blue*). . Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte* (nilai maksimum 255 warna), jadi satu piksel pada citra warna diwakili oleh 3 *byte* [3].

## 2.7 Grayscale

Untuk menghasilkan akuisisi citra warna maka, harus dilakukan proses *grayscale* yaitu citra dengan aras keabuan. Citra *grey* memiliki jumlah bit 8 dikarenakan, jumlah warnanya

adalah 286, sehingga jumlah warnanya adalah 28=256, nilainya berada pada jangkauan 0-255. Rumus dari *greyscale* adalah sebagai berikut: [5]

$$S = \frac{r+g+b}{3}$$

(2.2)

Keterangan:

S = Nilai Intensitas *grayscale*

r = Nilai Intensitas *red*

g = Nilai Intensitas *green*

b = Nilai Intensitas *blue*

## 2.8 Filter Mean

Efek smoothing yang digunakan dalam mean filter adalah spatial filtering, yaitu menggunakan piksel disekitarnya, kemudian piksel tersebut diproses dalam matriks berdimensi NxN, nilai n harus ganjil sehingga piksel tersebut dapat diletakkan ditengah matiks. Sebagai contoh matrik berdimensi 3 X 3 seperti gambar 4 di bawah ini:

1	2	3
4	T	5
6	7	8

Matriks pada gambar 2.5 terdapat nilai 1 hingga 8 yang merupakan nilai piksel. Terdapat nilai T diantara nilai 4 dan 5. Pengambilan piksel dimulai dengan mengambil piksel yang akan diproses, disimpan dalam nilai T. Kemudian diambil piksel-piksel sekitarnya sehingga matrik terisi penuh. Proses selanjutnya dijumlahkan semua nilai yang terdapat pada matrik tersebut. Hasil penjumlahan tersebut dibagi dengan jumlah titik yang terdapat pada matrik tersebut. Bilangan pembagi ini dapat diperoleh dari perkalian antara N X N. Pada gambar 2.5, maka hasil pembagiannya adalah 9. Sembilan diperoleh dari hasil kali matrik 3 X 3. Hasil pembagian tersebut akan menggantikan nilai T. Nilai T yang baru akan ditampilkan pada layar monitor untuk menggantikan nilai T yang lama. Proses diatas adalah untuk menggambar *grayscale* (hitam-putih), untuk menggambar berwarna maka masing-masing titik terlebih dahulu ditentukan nilai warna merah (R), hijau (G), dan biru (B).

Masing-masing nilai RGB dijumlahkan. Hasil penjumlahan nilai RGB dibagi dengan jumlah titik yang diproses. Hasil dari pembagian digunakan untuk menentukan warna baru yang akan di letakkan pada titik T [6].

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.3)$$

Keterangan:

X = Nilai rata-rata (Mean)

n = Jumlah data

i x = Nilai ke -i

i = Nilai Awal

### 2.9 Thresholding

Untuk dapat memisahkan obyek dengan latar belakang maka digunakan metode *Image thresholding*. Metode ini adalah teknik yang lebih efektif dalam segmentasi citra dan juga sederhana dan lebih mudah. Teknik ini disebut juga binerisasi citra karena mengubah citra berwarna menjadi hitam dan putih. Pada beberapa aplikasi pengolahan citra, terlebih dahulu dilakukan *threshold* terhadap citra *gray level* untuk dapat menjadi citra *biner* (citra yang memiliki nilai level keabuan 0 atau 255). Misalkan suatu tingkat keabuan *image f* dapat mempunyai K kemungkinan *gray level* 0,1,2, ...,K-1. Ditentukan sebuah nilai *threshold, T* yang berada pada *range* tingkat keabuan:  $T \in \{0,1,2,\dots,K-1\}$ . Proses *thresholding* adalah proses perbandingan sederhana dimana setiap nilai piksel di *f* dibandingkan dengan nilai *T*. Rumus citra biner adalah sebagai berikut: [7]

$$g(n) = \begin{cases} 0 & \text{if } f(n) \geq T \\ 1 & \text{if } f(n) < T \end{cases} \quad (2.4)$$

Keterangan:

g(n) = citra *biner* dari citra *grayscale*

f(n) = nilai *threshold*

T = *Threshold*

### 2.10 Euclidean Distance

Menghitung jarak dua buah objek dengan klasifikasi tetangga terdekatnya disebut metode *euclidean distance*. Jarak *Euclidean (Euclidean distance)* yang merupakan selisih antara 2 buah vektor yang akan dibandingkan untuk

pengenalan sebuah objek yang akan diuji. Rumus penghitungan jarak ditulis sebagai berikut [8]:

$$de = \sum_{k=1}^m \sqrt{\sum_{k=1}^m (d_{i,k} - k_j)^2} \quad (2.5)$$

Keterangan:

de: jarak *euclidean*

fdi: data buah training

kj: data buah uji

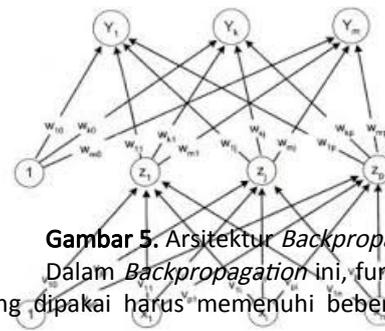
m: jumlah data pelatihan

### 2.11 Backpropagation

*Backpropagation*

mempunyai

keseimbangan antara jaringan untuk mengenali pola serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa tapi tidak sama dengan pola yang dipakai selama pelatihan. Dalam *backpropagation* memiliki beberapa unit *neuron* yang ada dalam satu atau lebih layar tersembunyi. Arsitektur *backpropagation* dengan n buah masukan (ditambah satu bias), dan sebuah layar tersembunyi yang terdiri dari p unit (ditambah satu bias), dan juga m buah unit keluaran. Arsitektur jaringan *Backpropagation* seperti tampak pada Gambar 2.6 dimana  $V_{ji}$  adalah bobot garis dari unit masukan  $X_i$  ke unit layar tersembunyi  $Z_j$  ( $V_{j0}$  adalah bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi  $Z_j$ ). Sedangkan  $W_{kj}$  adalah bobot dari unit layar tersembunyi  $Z_j$  ke unit keluaran  $Y_k$  ( $W_{k0}$  adalah bobot dari bias di layar tersembunyi ke unit keluaran ( $Y_k$ )).



**Gambar 5.** Arsitektur *Backpropagation*  
 Dalam *Backpropagation* ini, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat

yaitu: kontinu, terdeferensial dengan mudah dan merupakan fungsi yang tidak turun. Fungsi yang dipakai adalah *sigmoid* biner, *tansig* dan *purelin*, dengan persamaan sebagai berikut: [9]

Fungsi *sigmoid*:

$$y = f(x) = \frac{1}{1+e^{-\theta x}} \quad (2.6)$$

$$y' = f'(x) = \theta f(x)[1 - f(x)]$$

Fungsi *tansig*:

$$y = f(x) = \frac{1-e^{-2x}}{1+e^{-2x}} \quad (2.8)$$

$$y' = f'(x) = [1 + f(x)][1 - f(x)] \quad (2.9)$$

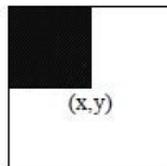
### 2.12 Viola Jones

Ada 4 kunci utama dalam Metode *Viola-Jones* yaitu *Haar Like Feature*, *Integral Image*, *Adaboost learning* dan *Cascade classifier*. Pertama yaitu *Haar Like Feature* adalah hasil pengurangan dari jumlah piksel dalam daerah persegi panjang. Contoh *Haar Like Feature* disajikan dalam Gambar 6.



Gambar 6. *Haar Like Feature*

Kemudian pada tahap *Integral Image*, yaitu mengubah nilai piksel menjadi representasi citra baru, seperti Gambar 7.



Gambar 7. *Integral Image*

Berdasarkan Gambar 2.8, persamaan dari integral image adalah:

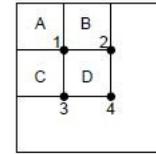
$$ii(x,y) = \sum_{x' \leq x, y' \leq y} v(x',y') \quad (2.10)$$

Keterangan

$ii(x,y)$  = Citra integral pada lokasi  $x,y$

$(x',y')$  = nilai piksel pada citra asli

Perhitungan nilai dari suatu fitur dapat dilakukan secara cepat dengan menghitung nilai citra integral pada empat buah titik sebagaimana disajikan dalam Gambar 8.



Gambar 8. Perhitungan nilai fitur

Jika titik 1 adalah A, titik 2 adalah A+B, titik 3 adalah A+C, dan di titik 4 adalah A+B+C+D, maka jumlah piksel di daerah D dapat diketahui dengan cara  $4 + 1 - (2 + 3)$ . Algoritma *Adaboost learning* berfungsi untuk meningkatkan kinerja klasifikasi pembelajaran sederhana dalam menggabungkan classifier-classifier lemah menjadi satu classifier kuat. Classifier lemah adalah suatu jawaban benar dengan tingkat kebenaran yang kurang akurat

$$hj(x) = \begin{cases} 1, & \text{jika } P_j F_j(x) < P_j \theta_j(X) \\ 0, & \text{lainnya} \end{cases} \quad (2.11)$$

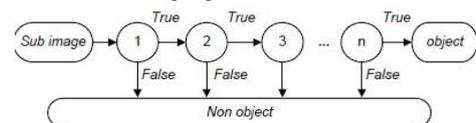
Keterangan:

$hj(x)$  adalah klasifikasi lemah,  $P_j$  adalah *parity* ke  $j$ ,  $\theta_j$  adalah *threshold* ke  $j$  dan  $x$  adalah dimensi *sub image* misalnya  $24 \times 24$ .

Langkah-langkah untuk mendapatkan sebuah *classifier* kuat dinyatakan dalam suatu algoritma sebagai berikut:

1. Diberikan contoh gambar  $(x_1, y_1), \dots, (x_n, y_n)$  dimana  $y_i = 0$  untuk contoh positif dan  $y_i = 1$  untuk contoh negatif
2. Inisialisasi bobot  $y_i$ ;  $1 = \frac{1}{2m}, \frac{1}{2l}$  m dan l adalah jumlah negatif dan positif.

*Cascade classifier* adalah sebuah metode untuk mengkombinasikan *classifier* yang kompleks dalam sebuah struktur bertingkat yang dapat meningkatkan kecepatan pendeteksian obyek dengan memfokuskan pada daerah citra yang berpeluang saja. Struktur *cascade classifier* disajikan Gambar 9. [10]



Gambar 9. *Cascade Clasifier*

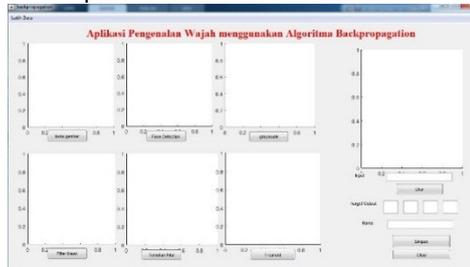
### 3. Hasil dan Pembahasan

Pada bagian ini, diuraikan hasil perancangan aplikasi dan pembahasan dari metode yang telah diuraikan diatas

### 3.1. Hasil

Hasil dari perancangan aplikasi pengenalan wajah dengan algoritma backpropagation adalah sebagai berikut:

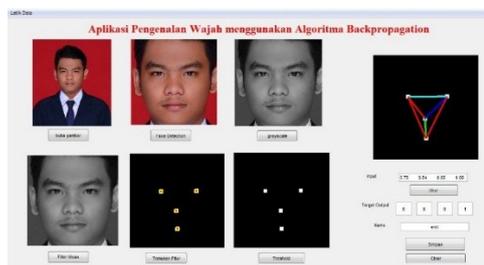
#### A. Tampilan Utama



**Gambar 10.** Tampilan Utama

Pada Gambar 10 menampilkan halaman utama dari sistem yang telah dibuat, halaman ini muncul ketika pertama kali sistem dijalankan. Pada halaman ini menampilkan proses dari sistem dimulai hingga melakukan penyimpanan. Sebelum pengguna melakukan pengujian maka, pengguna sistem harus terlebih dahulu menyimpan data-data gambar/foto pada halaman ini.

#### B. Tampilan Akhir



**Gambar 11.** Tampilan Akhir

Gambar 11 adalah tampilan Akhir ataupun tampilan pengujian. Sekilas tampak sama dengan tampilan utama, yang membedakan adalah pada tampilan ini terdapat tombol "Uji". Pada tampilan ini seluruh data yang telah dilatih dan disimpan pada tampilan utama akan diuji pada tampilan ini. Tampilan ini akan

menampilkan apakah data yang telah disimpan dan dilatih cocok atau tidak.

#### C. Face Detection



**Tabel 1.** Hasil Face Detection

Ini adalah tahapan setelah pemilihan gambar. Pada tahapan ini sistem akan memisahkan gambar wajah dengan bukan wajah, berguna untuk memudahkan dalam tahapan selanjutnya.

#### D. Grayscale



**Tabel 2.** Hasil Grayscale

Setelah gambar melalui tahapan *face detection*, maka akan masuk pada tahap *grayscale* yaitu, dimana gambaryang semula berwarna akan menjadi keabuan (*gray*)

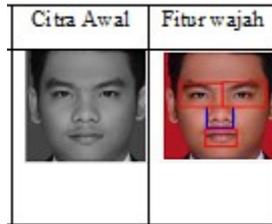
#### E. Filter Mean



**Tabel 3.** Filter *mean*

Filter mean ini berfungsi untuk menghilangkan noise pada gambar

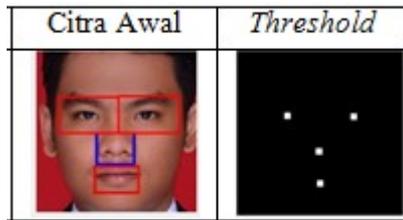
#### F. Temukan Fitur



Tabel 4. Fitur Wajah

Pada tahap ini sistem akan dapat mengenali fitur wajah seperti mata, hidung, mulut.

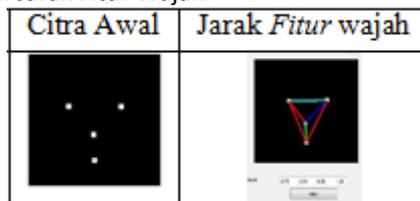
G. *Threshold*



Tabel 5. Threshold

Gambar yang telah ditemukan fiturnya maka akan diubah menjadi citra biner. Hasil dari threshold ini adalah titik-titik fitur wajah yang telah ditemukan fiturnya.

H. Jarak Fitur Wajah



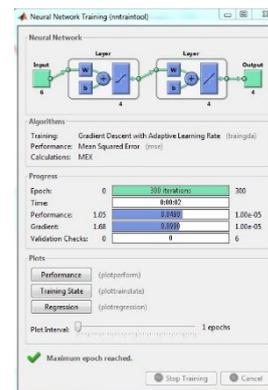
Tabel 6. Jarak fitur

Setelah gambar *threshold* didapatkan maka proses selanjutnya adalah melakukan perhitungan jarak fitur. Jarak fitur diukur berdasarkan jarak fitur yang sebelumnya. Pengukuran jarak dengan menggunakan metode *Euclidean Distance*.

I. Pelatihan Data

Setelah seluruh proses preprocessing telah selesai dan data telah disimpan,

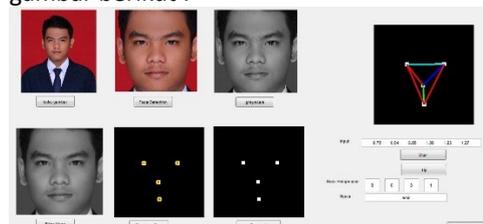
maka dilakukan pelatihan data. Tujuan pelatihan adalah sebagai proses pembelajaran pada algoritma backpropagation dalam mengenali wajah dan akan masuk dalam database backpropagation. Dalam pelatihan data, mungkin tidak dapat hanya melakukan sekali pelatihan, mungkin bias saja berkali-kali hingga data yang dimasukkan cocok. Proses pelatihan data ditunjukkan pada gambar.



Gambar 12. Pelatihan Data

J. Proses Pengujian

Ini adalah tahapan terakhir pada pengenalan wajah, dimana setelah data tersebut dilakukan pelatihan, maka akan dilakukan pengujian. Tujuannya untuk mengetahui apakah data gambar yang telah disimpan dan dilatih sama dengan data gambar yang keluar pada saat pengujian. seperti ditunjukkan pada gambar berikut :



Gambar 13. Pengujian.

3.2. Pembahasan

Pada bagian ini dilakukan agar dapat mengetahui kemampuan aplikasi atau sistem

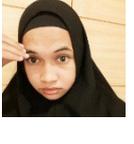
dalam mengenali gambar wajah, dan seberapa akurat aplikasi dalam melakukan pereprocessing. Untuk menguji sistem maka, seluruh data yang disimpan akan masuk ke data latih, kemudian untuk pengujian akan masuk kedalam data uji. pengujian ini dilakukan menggunakan wajah yang tampak dari depan. Berikut adalah tabel data latih

Tabel 7. Data Latih

Nama	Gambar	Keterangan
Emil		<b>Jarak fitur</b> [0.75 0.54 0.85 1.06 1.23 1.27] <b>Target Output</b> [ 0 0 0 1 ]
Sayuti		<b>Jarak fitur</b> [0.67 0.41 0.74 0.87 1.05 1.07] <b>Target Output</b> [ 0 0 1 1 ]
Mama		<b>Jarak fitur</b> [1.2 0.52 1.21 1.18 0.86 0.74] <b>Target Output</b> [ 1 0 0 1 ]
Intan		<b>Jarak fitur</b> [0.57 0.2 0.69 1.25 0.77 0.49] <b>Target Output</b> [ 1 1 0 0 ]
Reza		<b>Jarak fitur</b> [0.54 0.41 0.19 0.73 0.14 0.6] <b>Target Output</b> [ 0 1 1 0 ]
Kak Melly		<b>Jarak fitur</b> [0.68 1.77 2.19 1.65 1.25 0.41] <b>Target Output</b> [ 1 1 1 0 ]
Nanda		<b>Jarak fitur</b> [2.71 0.74 2.42 2.24 2.11 1.76] <b>Target Output</b> [ 1 1 1 1 ]
Han Hyo Joo		<b>Jarak fitur</b> [0.55 0.32 0.58 0.73 0.85 0.8] <b>Target Output</b>

	[ 0 0 1 0 ]
--	-------------

Tabel 8. Data Uji

Nama	Gambar	Keterangan
Emil		<b>Jarak fitur</b> [0.75 0.54 0.85 1.06 1.23 1.27] <b>Target Output</b> [ 0 0 0 1 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI
Sayuti		<b>Jarak fitur</b> [0.67 0.41 0.74 0.87 1.05 1.07] <b>Target Output</b> [ 0 0 1 1 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI
Mama		<b>Jarak fitur</b> [1.2 0.52 1.21 1.18 0.86 0.74] <b>Target Output</b> [ 1 0 0 1 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI
Intan		<b>Jarak fitur</b> [0.57 0.2 0.69 1.25 0.77 0.49] <b>Target Output</b> [ 1 1 0 0 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI
Reza		<b>Jarak fitur</b> [0.54 0.41 0.19 0.73 0.14 0.6] <b>Target Output</b> [ 0 1 1 0 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI
Kak Melly		<b>Jarak fitur</b> [0.68 1.77 2.19 1.65 1.25 0.41] <b>Target Output</b> [ 1 1 1 0 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI

<p><b>Nanda</b></p> 	<p><b>Jarak fitur</b> [2.71 0.74 2.42 2.24 2.11 1.76] <b>Target Output</b> [ 1 1 1 1 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI</p>
<p><b>Han Hyo Joo</b></p> 	<p><b>Jarak fitur</b> [0.55 0.32 0.58 0.73 0.85 0.8] <b>Target Output</b> [ 0 0 1 0 ] <b>Hasil Pengenalan</b> COCOK / BERHASIL DIKENALI</p>
<p><b>Tidak diketah ui</b></p> 	<p><b>Jarak fitur</b> [0.54 0.42 1.2 0.76 0.13 0.83] <b>Target Output</b> [ 0 0 1 1 ] <b>Hasil Pengenalan</b> TIDAK COCOK</p>
<p><b>Tidak diketah ui</b></p> 	<p><b>Jarak fitur</b> [0.16 0.6 0.86 1 0.47 1.26] <b>Target Output</b> [ 0 0 1 1 ] <b>Hasil Pengenalan</b> TIDAK COCOK</p>

2. Pada tahapan *preprocessing* harus melalui inti proses seperti menemukan citra dan fitur wajah dengan metode *viola-jones*, serta untuk menemukan jarak antar fitur wajah menggunakan metode *Euclidean distance*.
3. Setelah dilakukan pengujian sebanyak 10 citra wajah, menghasilkan 8 citra wajah dapat diketahui identitasnya 5 citra wajah tidak dapat dikenali, maka akurasi dari aplikasi ini sebesar 80%.
4. Dari proses pelatihan dan pengujian wajah dapat dikenali apabila citra atau gambar wajah tersebut telah tersimpan didalam *database*. Jika tidak terdapat didalam *database* kemudian dilakukan pengujian maka akan menampilkan data wajah yang mendekati.
5. Pada aplikasi ini masih terdapat kelemahan yaitu, untuk menemukan citra bentuk wajah serta untuk menemukan fitur belum terlalu akurat, kemudian masih manual dalam melakukan pengenalan yaitu menggunakan foto belum *real-time* dengan menggunakan kamera lalu belum juga dapat terkoneksi internet sebagai pengenalan jarak jauh untuk *security sistem* serta belum terintegrasi dengan aplikasi *mobile* seperti *android* dan *apple*.

#### 4. Penutup

Pada bagian ini diuraikan kesimpulan dan saran dari perancangan aplikasi pengenalan wajah dengan menggunakan algoritma *backpropagation*.

##### 4.1. Kesimpulan

Berdasarkan penelitian pengembangan aplikasi pengenalan wajah menggunakan algoritma *backpropagation* maka dapat diambil kesimpulan yaitu:

1. Berdasarkan dari hasil pengujian bahwa untuk dapat mengenali wajah seseorang dengan algoritma *backpropagation* harus melalui beberapa tahapan, yaitu tahapan *preprocessing*, tahapan pelatihan dan tahapan pengujian

##### 4.2. Saran

Beberapa saran dari hasil penelitian yang telah dilakukan:

1. Diharapkan dapat menemukan metode yang jauh lebih akurat untuk menemukan citra wajah dan fitur wajah.
2. Diharapkan dapat menemukan metode untuk dapat mengenali wajah dari segala posisi tidak hanya tampak dari depan.
3. Dapat dilakukan secara *real-time* menggunakan kamera dan terintegrasi dengan internet.
4. Dapat menjadi *security system* sebagai pengenalan jarak jauh.
5. Diharapkan dapat diakses secara *mobile* baik dalam aplikasi *android*, maupun *apple*

#### REFERENSI

- [1] P. C. Riau, "Faktor-Faktor yang Mempengaruhi Sistem Pengenalan Wajah Menggunakan Metode Eigenface pada Perangkat Mobile Berbasis Android," *J. Politek. Caltex Riau*, vol. 2, no. 2, pp. 127–136, 2016.
- [2] R. N and G. T, "FACE RECOGNITION SYSTEM USING BACK PROPAGATION Address for Correspondence," no. 1, 2012.
- [3] A. B. S and H. Maulana, "Pengenalan Citra Wajah Sebagai Identifier Menggunakan Metode Principal Component Analysis ( PCA )," vol. 9, no. 2, pp. 166–175, 2016.
- [4] S. Habiburrahman, F. Hakim, I. Cholissodin, and A. W. Widodo, "Seleksi Fitur Dengan Particle Swarm Optimization Untuk Pengenalan Pola Wajah Menggunakan Naive Bayes ( Studi Kasus Pada Mahasiswa Universitas Brawijaya Fakultas Ilmu Komputer Gedung A )," vol. 1, no. 10, pp. 1045–1057, 2017.
- [5] D. E. Kurniawan, K. Adi, and F. Rohim, "Sistem Identifikasi Biometrika Wajah Menggunakan Metode Gabor KPCA dan Mahalanobis Distance," *J. Sist. Inf. Bisnis*, vol. 1, pp. 6–10, 2012.
- [6] B. Yuwono, "Image Smoothing Menggunakan Mean Filtering, Median Filtering, Modus Filtering dan Gaussian Filtering," *Telemat. UPN Veteran*, vol. 7, no. 1, 2010.
- [7] E. Permata, "Identifikasi Obyek Benda Tajam Menggunakan Pengolahan Citra Digital Pada Citra X-Ray," vol. 1, no. 1, pp. 1–14, 2016.
- [8] B. Aditya, A. Hidayatno, and A. A. Zahra, "Sistem Pengenalan Buah Menggunakan Metode Discrete Cosine Transform dan Euclidean Distance," *Transient*, vol. 3, pp. 134–138, 2014.
- [9] S. Kusmaryanto, "Jaringan Saraf Tiruan Backpropagation untuk Pengenalan Wajah Metode Ekstraksi Fitur Berbasis Histogram," *J. EECCIS Vol. 8, No. 2, Desember 2014*, vol. 8, no. 2, pp. 193–198, 2014.
- [10] A. Hendrotriatmoko, S. Hadi, and H. S. Dachlan, "Penggunaan Metode Viola-Jones dan Algoritma Eigen dalam Sistem Kehadiran Pegawai," vol. 8, no. 1, pp. 41–46, 2014.