

## Aplikasi Pengenalan Wajah Dengan Menggunakan Metode Euclidean Distance Dan Wavelet Haar

Robby Wijaya<sup>1)</sup> Herman<sup>2)</sup>

Program Studi Teknik Informatika<sup>1,2)</sup>

Sekolah Tinggi Manajemen Informatika Dan Komputer – TIME<sup>1,2)</sup>

Jalan Merbabu No. 32 AA-BB Medan, Sumatera Utara

e-mail : robbhuang98@gmail.com<sup>1)</sup> hrman\_ang@yahoo.com<sup>2)</sup>

### Abstrak

Wajah sebagai salah satu yang dapat digunakan sebagai identifikasi seseorang. salah satunya adalah dengan menggunakan metode Transformasi Wavelet Haar dan Euclidean Distance. Alihragam Wavelet Haar berfungsi untuk mengekstraksi ciri citra masukan sedangkan Euclidean Distance berfungsi untuk proses pengenalan. Prosesnya yaitu mengubah ke citra abu-abu dan normalisasi ukuran jika diperlukan. Perhitungan dilakukan untuk mencocokkan dua buah citra yang diukur melalui nilai jarak Euclidean. Semakin kecil nilai Euclidean maka semakin mirip dua citra yang dicocokkan. Aplikasi menyediakan interface untuk melakukan penyimpanan data identitas dan gambar citra wajah. Selain itu, juga terdapat interface untuk menampilkan data identitas dan gambar yang tersimpan dalam database.

**Kata Kunci** : citra digital, metode Wavelet Haar, Euclidean Distance

### 1. Latar Belakang

Pada zaman saat ini wajah merupakan salah satu bagian dari tubuh manusia yang dapat digunakan untuk mengenali seseorang selain iris mata dan sidik jari tentunya. Dengan metode aplikasi pengenalan wajah yang ada sekarang kebanyakan sistem pengenalan hanya menggunakan gambar wajah dengan satu macam ekspresi saja.

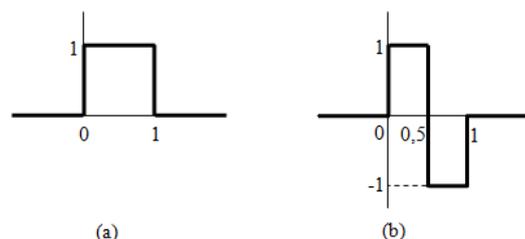
Kemampuan suatu aplikasi untuk mengenali wajah di zaman saat ini masih sangat terbatas. Mengingat pengenalan wajah saat ini dapat digunakan sebagai suatu faktor keamanan seperti pengenalan wajah pada *smartphone* ataupun lainnya.

Untuk itu di butuhkan suatu aplikasi pengenalan wajah yang mampu mengenali wajah dengan berbagai ekspresi, seperti senyum, tertawa, marah, dan ekspresi lainnya. Aplikasi ini bukan hanya mampu mengenali ekspresi wajah tapi juga dapat mencari identitas personal seseorang dengan tingkat keakuratan yang lebih baik dengan menggunakan metode *Euclidean Distance* dan *Wavelet Haar*.

### 2. Landasan Teori

#### Wavelet Haar

*Wavelet Haar* yang berfungsi untuk ekstraksi ciri atau pengambilan ciri penting dari suatu citra. *Wavelet Haar* merupakan *Wavelet* yang tertua dan sederhana. Fungsi penskalaan dan *wavelet Haar* dapat dilihat pada Gambar (a) menunjukkan gambar fungsi penskalaan Haar sedangkan Gambar (b) menunjukkan gambar fungsi *wavelet Haar*.



Gambar 1. Fungsi penskala wavelet Haar

Dalam *Wavelet Haar*, terdapat dua proses yang harus dilakukan yaitu alihragam *forward* dan alihragam *inverse*. Alihragam *forward* berguna untuk memecah atau dekomposisi citra. Sedangkan alihragam *inverse* adalah kebalikannya, yaitu membentuk kembali pecahan-pecahan citra dari proses *forward* menjadi sebuah citra seperti semula.

#### Alihragam Forward

Langkah pertama dari alihragam *forward* Haar untuk delapan sinyal unsur. Disini sinyal dikalikan dengan matriks transformasi *forward*

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \leftarrow \begin{bmatrix} a_0 \\ c_0 \\ a_1 \\ c_1 \\ a_2 \\ c_2 \\ a_3 \\ c_3 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix}$$

Gambar 2. Matriks alihragam *forward Haar*

**Alihragam Inverse**

Operasi matriks untuk membalikkan langkah pertama alihragam Haar untuk delapan sinyal unsur ditunjukkan dibawah ini :

$$\begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ c_0 \\ a_1 \\ c_1 \\ a_2 \\ c_2 \\ a_3 \\ c_3 \end{bmatrix} \leftarrow \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

Gambar 3. Matriks alihragam *inverse Haar*

**Euclidean Distance**

Metode *Euclidean Distance* digunakan dalam proses pengenalan. Nilai jarak Euclidean yang mendekati nilai nol, akan menunjuk pada citra tertentu. Nilai vektor ciri citra masukan yang memiliki nilai vektor ciri yang sama dengan vektor ciri citra tertentu akan memiliki nilai jarak Euclidean yang mendekati nol. Misal nilai vektor ciri masukan citra  $A_i = (A_{i1}, A_{i2}, \dots, A_{in})$  nilai vektor ciri citra ke-  $j$  adalah  $B_j = (B_{j1}, B_{j2}, \dots, B_{jn})$  jarak Euclidean antara nilai vector ciri citra masukan dan nilai vektor ciri citra ke-  $j$  dinyatakan oleh :

$$D(A, B) = \sqrt{\sum_{i=0}^n \frac{(A_i - B_i)^2}{A_i}} \dots\dots\dots (i)$$

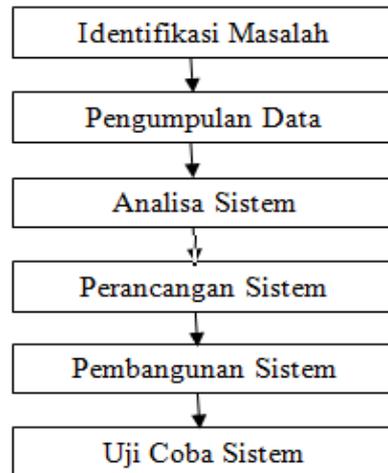
dengan:

- D(A, B) = Jarak Euclidean antara citra wajah uji dengan citra wajah basis data
- $A_i$  = Vektor ciri citra uji
- $B_i$  = Vektor ciri citra basis data
- n = Panjang vektor A dan vektor B

**3. Metode Penelitian**

**Analisis**

Dalam penelitian ini, kerangka kerja yang dipakai terdiri dari beberapa tahapan, adapun tahapan tersebut adalah:



Gambar 4. Kerangka Kerja

Dalam penelitian ini data yang dikumpulkan diambil dari beberapa sumber seperti buku atau media lainnya seperti *internet*. Untuk gambar yang digunakan diambil dari *internet* dan beberapa gambar yang *dedit* menggunakan *Adobe Photoshop*.

Perangkat lunak ini dirancang dengan menggunakan bahasa pemrograman *Microsoft Visual Basic 2010* dengan menggunakan beberapa objek dasar seperti :

1. *Label*, yang digunakan untuk menampilkan keterangan.
2. *ComboBox*, yang digunakan untuk menyediakan pilihan.
3. *Button*, yang digunakan sebagai tombol eksekusi.
4. *PictureBox*, yang digunakan untuk menampilkan gambar citra.
5. *SaveFileDialog*, yang digunakan untuk menampilkan dialog *save*.
6. *OpenFileDialog*, yang digunakan untuk menampilkan dialog *open*.
7. *TextBox*, yang digunakan untuk menampilkan hasil proses perhitungan.

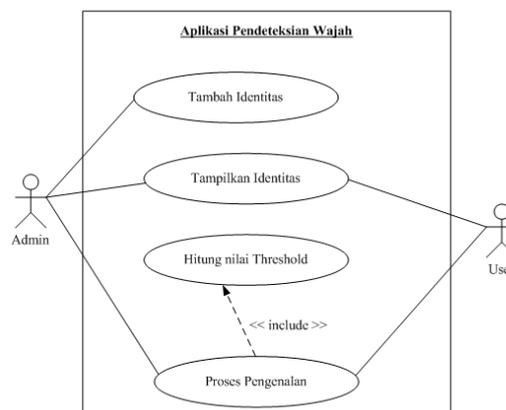
Teknik terstruktur yang digunakan untuk merancang dan mendokumentasikan perangkat lunak adalah *pseudocode*. Rancangan tampilan dari aplikasi pendeteksian wajah ini dapat dirincikan sebagai berikut:

1. *Form 'Main'*.
2. *Form 'Tambah Identitas'*.
3. *Form 'Tampilkan Identitas'*.
4. *Form 'Pengenalan'*.
5. *Form 'Login'*.
6. *Form 'Cari'*.

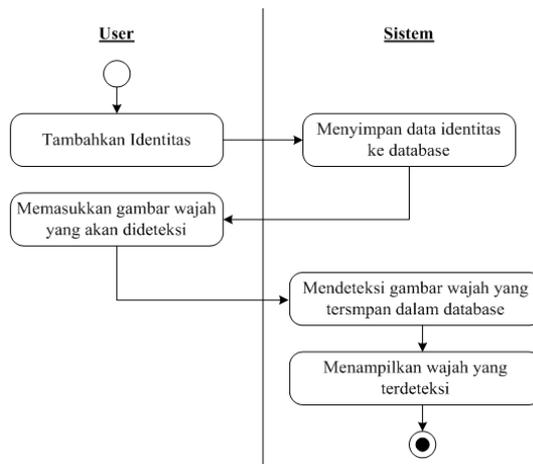
Perangkat lunak yang telah dibangun harus dilakukan uji coba terlebih dahulu sebelum digunakan. Proses dilakukan dengan *testing and debugging* terhadap *coding* yang dirancang tersebut. Jika terjadi *error* maka akan dilakukan perbaikan.

### Pemodelan Sistem

Sistem yang dibuat akan dimodelkan dengan menggunakan *use case diagram* yang dapat digambarkan sebagai berikut:



Gambar 5. Use Case Diagram

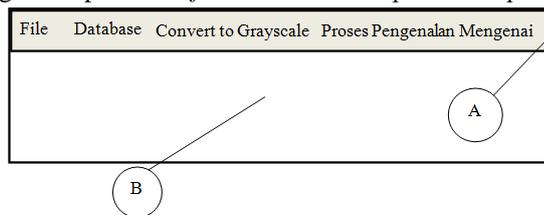


Gambar 6. Activity Diagram

Proses kerja dari sistem yang dibuat dapat dirincikan sebagai berikut:

**Form ‘Main’**

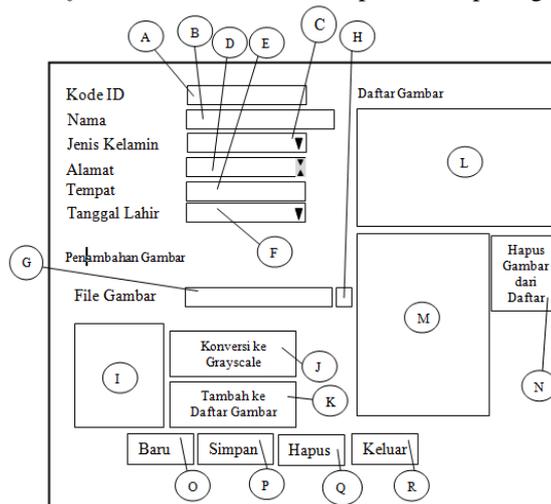
Form ini merupakan form inti dari perangkat lunak yang berfungsi untuk menghubungkan form-form yang ada pada perangkat lunak. Rancangan tampilan dari form ‘Main’ ini dapat dilihat pada gambar dibawah



Gambar 7. Rancangan Form ‘Main’

**Form ‘Tambah Identitas’**

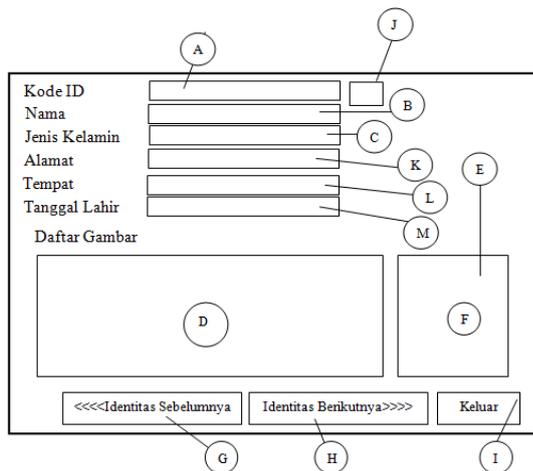
Form ini berfungsi untuk menambahkan identitas baru yang akan disimpan ke dalam database yang digunakan oleh sistem. Rancangan tampilan dari form ‘Tambah Identitas’ dapat dilihat pada gambar dibawah



Gambar 8. Rancangan Form ‘Tambah Identitas’

**Form ‘Tampilkan Identitas’**

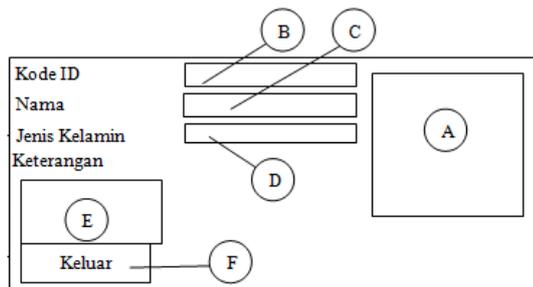
Form ini berfungsi untuk menampilkan semua identitas yang tersimpan dalam database. Rancangan tampilan dari form ‘Tampilkan Identitas’ dapat dilihat pada gambar dibawah



**Gambar 9.** Rancangan Form 'Tampilkan Identitas'

**Form 'Pengenalan'**

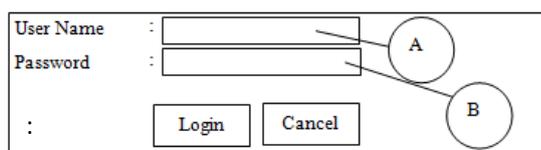
Form 'Pengenalan' berfungsi untuk menampilkan hasil pendeteksian wajah yang telah dilakukan. Rancangan tampilan dari form 'Pengenalan' dapat dilihat pada Gambar dibawah



**Gambar 10.** Rancangan Form 'Pengenalan'

**Form 'Login'**

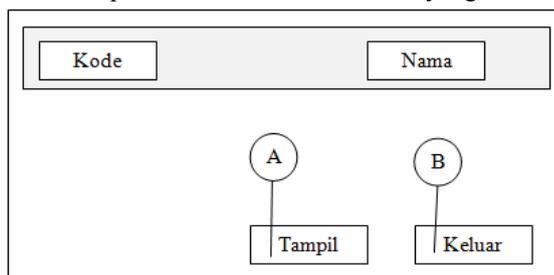
Form 'Login' berfungsi untuk melakukan login data user yang akan menggunakan sistem yang hanya dapat digunakan oleh user admin. Rancangan tampilan dari form 'Login' dapat dilihat pada Gambar dibawah :



**Gambar 11.** Rancangan Form 'Login'

**Form 'Cari'**

Form 'Cari' berfungsi untuk menampilkan semua daftar identitas yang terdaftar pada database.



**Gambar 12.** Rancangan Form 'Cari'

#### 4. Hasil

Untuk menggunakan perangkat lunak ini, jalankan file "Deteksi Wajah.EXE", maka akan ditampilkan tampilan login dari program seperti terlihat pada gambar berikut:



**Gambar 13.** Tampilan Login

Proses pengujian akan dilakukan dengan menggunakan dua buah identitas yang tersimpan dalam *database* dengan masing-masing memiliki 30 buah citra *database*, seperti terlihat pada tabel berikut:

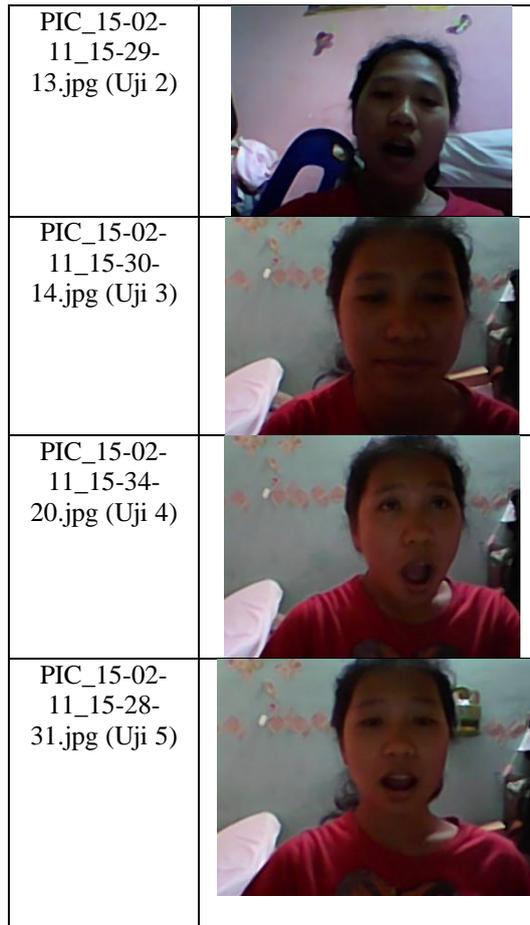
**Tabel 1.** Citra Database untuk ID-008

<b>Nama Citra</b>	<b>Gambar Citra</b>
PIC_15-02-11_15-27-59-70.jpg	
PIC_15-02-11_15-28-11-71.jpg	
PIC_15-02-11_15-28-22-72.jpg	
PIC_15-02-11_15-28-31-73.jpg	

Sementara itu, citra yang digunakan untuk melakukan pengujian dapat dirincikan sebagai berikut:

**Tabel 2.** Citra Pengujian

<b>Nama File Citra</b>	<b>Gambar Citra</b>
PIC_15-02-11_15-28-11.jpg (Uji 1)	



Hasil pengujian yang diperoleh dapat dirincikan sebagai berikut:

**Tabel 3.** Hasil Pengujian

No	Citra Uji	Citra Basis Data ke	Citra Terdekat	Jarak Euclidean	Keterangan
1	Uji1.jpg	ID-008	PIC_15-02-11_15-28-11-71.jpg	0.007127	Benar
2	Uji2.jpg	ID-008	PIC_15-02-11_15-28-31-73.jpg	0.48330219	Benar
3	Uji3.jpg	Citra tidak dikenali	-	0.53906368	Salah
4	Uji4.jpg	Citra tidak dikenali	-	0.480165956	Salah
5	Uji5.jpg	ID-008	PIC_15-02-11_15-28-31.jpg	0.007689	Benar

Berdasarkan hasil pengujian yang dilakukan, dapat diperoleh beberapa informasi berikut:

1. Dimensi citra input harus berukuran 240 x 240 piksel dan harus berupa citra *grayscale*.
2. Jumlah sampel yang digunakan harus banyak. Hal ini diperlukan agar dapat meningkatkan akurasi hasil pengenalan. Namun, jumlah sampel yang banyak akan mengakibatkan proses pengenalan wajah menjadi lama.
3. Berdasarkan hasil pengujian, diperoleh informasi bahwa akurasi dari metode yang digunakan adalah  $\frac{3}{5} * 100 \% = 60 \%$ .
4. Metode yang digunakan tidak dapat mengenali gambar wajah yang dimasukkan apabila ukuran kepala pada gambar tersebut berbeda dengan ukuran kepala pada gambar di *database*.
5. Untuk meningkatkan akurasi dari metode yang digunakan, maka dapat dikombinasikan dengan algoritma pendeteksian wajah sehingga proses pengenalan wajah hanya terhadap daerah wajah saja.

## 5. Kesimpulan

Dalam penelitian ini, penulis menarik kesimpulan :

1. Dimensi citra input harus berukuran 240 x 240 piksel dan harus berupa citra *grayscale*.
2. Jumlah sampel yang digunakan harus banyak. Hal ini diperlukan agar dapat meningkatkan akurasi hasil pengenalan. Namun, jumlah sampel yang banyak akan mengakibatkan proses pengenalan wajah menjadi lama.
3. Metode yang digunakan tidak dapat mengenali gambar wajah yang dimasukkan apabila ukuran kepala pada gambar tersebut berbeda dengan ukuran kepala pada gambar di *database*.

Adapun saran untuk penelitian berikutnya adalah :

1. Untuk meningkatkan akurasi dari metode yang digunakan, maka dapat dikombinasikan dengan algoritma pendeteksian wajah sehingga proses pengenalan wajah hanya terhadap daerah wajah saja.
2. Mengoptimalkan waktu eksekusi dengan memperbaiki algoritma atau dengan menggunakan bahasa pemrograman lainnya.

## Daftar Pustaka

- [1] Clifford Gower, John. *Euclidean Distance Matrix*, 2006.
- [2] Lim, R. Reinders, M.J.T. and Thiang., *Pengenalan Citra Wajah dengan Pemrosesan Awal Transformasi Wavelet*, Jurnal, vol. 02, 2000.
- [3] Purnomo, M. H. dan A. Muntasa, *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*, Graha Ilmu, Yogyakarta, 2010.
- [4] Ramadijanti, Nana dan Achmad Basuki, *"Fitur Bentuk pada Citra"*, 2008.
- [5] Wijaya, C. M. dan A. Prijono, *Pengolahan Citra Digital*, Informatika Bandung, Bandung, 2007.