



## **Aplikasi Metode Feature Reduction Dalam Pengenalan Wajah Manusia**

### ***Applicating Feature Reduction Method in Face Recognition***

**Fadhillah Azmi<sup>1)</sup>\*, Tommy Simatupang<sup>1)</sup>, Marselim HPE<sup>1)</sup>, Adhyaksa Putra Soaloon<sup>1)</sup> & Asido Silitonga<sup>1)</sup>**

1) Universitas Prima Indonesia, Indonesia

\*Corresponding Email: [tommymimatupang177@gmail.com](mailto:tommymimatupang177@gmail.com)

---

#### **Abstrak**

Pengenalan wajah manusia (human face recognition) memiliki banyak aplikasi seperti pada pemeriksaan keamanan, pencarian kembali gambar, dan sebagainya. Dengan metode aplikasi pengenalan wajah yang ada sekarang kebanyakan sistem pengenalan hanya menggunakan gambar wajah dengan satu macam ekspresi saja. Untuk itu dibutuhkan 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. Untuk mempercepat proses pencocokkan wajah, maka ciri wajah tersebut dapat disederhanakan lagi dengan membuang data yang kurang relevan. Metode pengenalan wajah ini dikenal dengan nama metode Feature Reduction. Tools yang dipakai untuk menganalisis dan desain adalah Unified Modeling Language. Hasil dari sebuah penelitian ini yaitu aplikasi untuk mengenali wajah manusia dengan menggunakan metode Feature Reduction.

**Kata Kunci:** *Feature Reduction, Pengenalan Wajah.*

#### **Abstract**

Face recognition has many applications such as security checks, image retrieval, and so on. With the face recognition application method that exists today most recognition systems only use face images with just one kind of expression. For that we need a face recognition application that is able to recognize faces in various forms, such as smile, laugh, anger, and other expressions. This application is not only able to recognize facial expressions and shapes but also can find someone's personal identity with a better level of accuracy. To accelerate the face matching process, the facial characteristics can be simplified again by removing the less relevant data. This face recognition method is known as the Feature Reduction method. The tools used to analysis and design are the Unified Modeling Language. The results of this study are applications to recognize human facial expressions using the Feature Reduction method.

**Keywords:** *Feature Reduction; Face Recognition.*

**How to Cite:** Simatupang, T., HPE, M., Soaloon, A. P., Silitonga, A., & Azmi, F. (2019). Aplikasi Metode Feature Reduction Dalam Pengenalan Wajah Manusia. *JITE (Journal Of Informatics Telecommunication Engineering)*. 2 (2): 52-61

---

## PENDAHULUAN

Wajah merupakan ukuran dan bentuk fisiologis yang sangat mudah dan sering digunakan untuk membedakan identitas individu yang satu dengan yang lainnya. Teknologi pengenalan wajah manusia secara digital atau lebih sering dikenal dengan *human face recognition*. Pengenalan wajah manusia (*human face recognition*) memiliki banyak aplikasi seperti pada pemeriksaan keamanan, pencarian kembali gambar, dan sebagainya. Secara umum cara kerjanya adalah dengan mengkonversikan foto, sketsa, dan gambar video menjadi sebuah angka, yang disebut dengan faceprint dan kemudian langsung membandingkannya dengan rangkaian angka lain yang mewakili wajah yang sudah dikenal sebelumnya. (Whitelaw, 2005).

Dengan metode aplikasi pengenalan wajah yang ada sekarang kebanyakan sistem pengenalan hanya menggunakan gambar wajah dengan satu macam ekspresi saja. Untuk itu dibutuhkan 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. Dalam literatur, dapat ditemukan cukup banyak metode yang bisa dipakai untuk melakukan proses pencocokkan wajah. Salah satu metode yang bisa di gunakan adalah dengan mengekstraksi ciri dari gambar wajah yang dimasukkan. Untuk mempercepat proses pencocokkan wajah, maka ciri wajah tersebut dapat disederhanakan lagi dengan membuang data yang kurang relevan. Metode pengenalan wajah ini dikenal dengan nama metode *Feature Reduction*. (Gower dan John, 2006 ).

Untuk mengatasi permasalahan diatas, penulis memilih untuk membuat aplikasi pengenalan wajah yang mampu mengenali dan mencari identitas personal seseorang menggunakan metode *Feature Reduction*.

## METODE PENELITIAN

*Feature reduction* adalah metode yang paling sering diterapkan untuk memilih sebuah *subset* dari ciri yang tersedia untuk mendeskripsikan data sebelum diterapkan sebuah algoritma pembelajaran. *Feature reduction* merupakan sebuah algoritma tangguh untuk menyederhanakan atau mempercepat proses perhitungan.

Analisis komponen dasar dapat mentransformasikan kumpulan data asli

dari variabel yang saling berhubungan menjadi sebuah kumpulan data yang lebih kecil yang merupakan kombinasi linier dari data asli. Variabel komponen dasar baru direferensikan sebagai ukuran domain.

Ekstraksi fitur (*feature extraction*) atau juga dikenal dengan sebutan *indexing*, adalah teknik pengambilan sifat-sifat khusus dari sebuah gambar. Ekstraksi fitur adalah langkah awal klasifikasi. Ekstraksi fitur citra berdasarkan tekstur pada orde pertama menggunakan metode statistik, yaitu dengan melihat distribusi derajat keabuan pada histogram citra tersebut. (Reinders dan Thiang, 2000).

Terdapat banyak teknik-teknik dalam mengekstrak gambar. Salah satunya adalah dengan menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM). Proses GLCM akan menghasilkan parameter yang digunakan dalam analisis citra. Parameter dasar itu berupa *nergy*, kontras, korelasi, *Sum of Square*, *Inverse Different Moment* (IDM), *sum average*, *sum variance*, *sum entropy*, *entropy*, *differential variance*, *differential entropy*, maksimum probabilitas, homogenitas, dissimilarity, dan lainnya. (Purnomo dan Muntasa, 2010).

Menunjukkan ukuran dispersi dari suatu citra:

$$\mu = \sum_n^N f_n p(f_n) \quad (1)$$

Energi disebut juga *Angular Second Moment* (ASM) menunjukkan ukuran sifat homogenitas pada citra digital. Rumus untuk mencari energi (ASM) adalah: (Tan, 2008)

$$ASM = \sum_{i,j=1}^{Ng} \quad (2)$$

Menunjukkan variasi elemen-elemen matriks *coocurance*. Rumus untuk menghitung *variance* (VAR) pada citra digital adalah: (Mullen, 2004).

$$VAR = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} g(i,j)(i - \mu)^2 \quad (3)$$

*Inverse Difference Moment* (IDM) atau homogenitas lokal adalah tingkat kehomogenan citra yang berderajat keabuan sejenis. Rumus IDM suatu gambar adalah:

$$IDM = \frac{\sum_{i=1}^{Ng}}{1} \quad (4)$$

Secara matematis, homogenitas pada GLCM adalah invers dari kontras pada GLCM, yaitu keseragaman intensitas keabuan pada citra. Rumus untuk mencari homogenitas (H) suatu gambar adalah:

$$H = \sum_{i,j=1}^{Ng} \frac{g(i,j)}{1 + (i-j)^2} \quad (5)$$

Proses kerja dari metode pendeteksian objek ini dapat dilihat pada contoh sederhana berikut ini:

Misalkan citra asli yang dimasukkan berupa citra berwarna, yang memiliki komposisi warna (R, G, B) sebagai berikut:

Tabel 1. Komponen Warna Citra Asli

151, 125, 167	146, 194, 205	168, 145, 179
135, 164, 197	164, 112, 131	204, 216, 231
119, 104, 131	147, 85, 64	164, 61, 14

Proses kerja dari metode pendeteksian objek menggunakan ekstraksi ciri (*feature extraction*) adalah sebagai berikut: (Wijaya, 2007).

1. Proses konversi citra ke bentuk *grayscale*.

Piksel (1, 1):

$$(151 + 125 + 167) / 3 = 147.67 (\approx 148)$$

Piksel (1, 2):

$$(146 + 194 + 205) / 3 = 181.67 (\approx 182)$$

Piksel (1, 3):

$$(168 + 145 + 179) / 3 = 164$$

Piksel (2, 1):

$$(135 + 164 + 197) / 3 = 165.33 (\approx 165)$$

Piksel (2, 2):

$$(164 + 112 + 131) / 3 = 135.67 (\approx 136)$$

Piksel (2, 3):

$$(204 + 216 + 231) / 3 = 217$$

Piksel (3, 1):

$$(119 + 104 + 131) / 3 = 118$$

Piksel (3, 2):

$$(147 + 85 + 64) / 3 = 98.67 (\approx 99)$$

Piksel (3, 3):

$$(164 + 61 + 14) / 3 = 79.67 (\approx 80)$$

Tabel 2. Komponen Warna Citra Grayscale

148	182	164
165	136	217
118	99	80

2. Konversikan citra ke bentuk gambar hitam putih. Misalkan nilai *threshold* = 150, maka proses kerjanya adalah sebagai berikut:

Piksel (1, 1):  $148 < 150 \rightarrow BW = 0$

Piksel (1, 2):  $182 > 150 \rightarrow BW = 1$

Piksel (1, 3):  $164 > 150 \rightarrow BW = 1$

Piksel (2, 1):  $165 > 150 \rightarrow BW = 1$

Piksel (2, 2):  $136 < 150 \rightarrow BW = 0$

Piksel (2, 3):  $217 > 150 \rightarrow BW = 1$

Piksel (3, 1):  $118 < 150 \rightarrow BW = 0$

Piksel (3, 2):  $99 < 150 \rightarrow BW = 0$

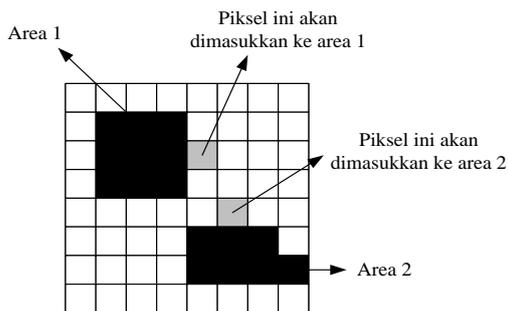
Piksel (3, 3):  $80 < 150 \rightarrow BW = 0$

Tabel 3. Komponen Biner Citra

0	1	1
1	0	1
0	0	0

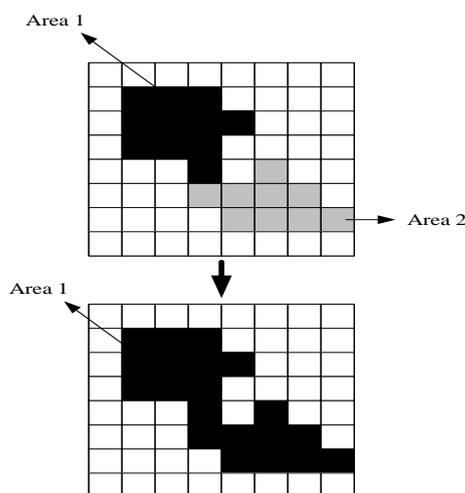
3. Proses segmentasi objek

Proses pengelompokan area yang terdeteksi. Sebuah piksel (titik) akan dimasukkan ke dalam sebuah area apabila piksel tersebut merupakan piksel tetangga dari salah satu piksel yang terdapat pada area yang terdeteksi, seperti terlihat pada ilustrasi berikut:



Gambar 1. Proses Pengelompokan Area

4. Apabila terdapat dua buah area yang memiliki piksel yang saling bertetangga, maka kedua area tersebut akan digabungkan menjadi satu.



Gambar 2. Proses Penggabungan Dua Area yang Memiliki Piksel yang Bertetangga

Ambil nilai kolom minimum dan maksimum serta baris minimum dan

maksimum dari area tersebut. Tampilkan kotak berdasarkan nilai kolom dan baris tersebut, dengan rumusan berikut:

Kotak.*Left* = Kolom Minimum

Kotak.*Top* = Baris Minimum

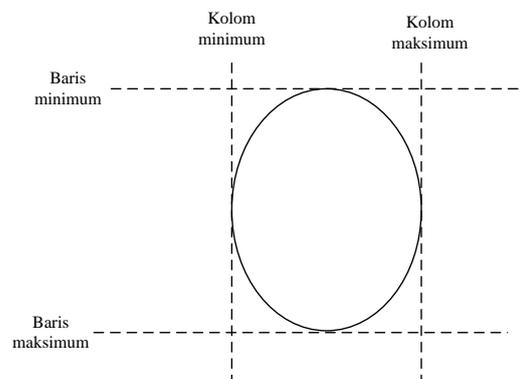
Kotak.*Width* = Kolom Maksimum-  
Kolom Minimum

Kotak.*Height* = Baris Maksimum-  
Baris Minimum

5. Proses pendeteksian area wajah.

Untuk mendeteksi area wajah, perangkat lunak akan melakukan pendeteksian area tersebut dengan mengambil beberapa nilai berikut:

- Posisi kolom minimum.
- Posisi kolom maksimum.
- Posisi baris minimum.
- Posisi baris maksimum.



Gambar 3. Proses Pendeteksian Posisi Kotak

6. Proses Ekstraksi Ciri dengan Metode GLCM

*Mean:*

$$\mu = (0 + 1 + 1 + 1 + 0 + 1 + 0 + 0 + 0) / 9$$

$$= 4 / 9 = 0.444$$

*ASM:*

$$\text{ASM} = 02 + 12 + 12 + 12 + 02 + 12 + 02 + 02 + 02$$

$$\text{ASM} = 0 + 1 + 1 + 1 + 0 + 1 + 0 + 0 + 0 = 9$$

$$\begin{aligned} \text{Var} = & 0 * (1 - 0.444)^2 + 1 * (2 - 0.444)^2 \\ & + 1 * (3 - 0.444)^2 + 1 * (4 - 0.444)^2 \\ & + 0 * (5 - 0.444)^2 + 1 * (6 - 0.444)^2 \\ & + 0 * (7 - 0.444)^2 + 0 * (8 - 0.444)^2 \\ & + 0 * (9 - 0.444)^2 \end{aligned}$$

$$\begin{aligned} \text{Var} = & 0 + 2.421136 + 6.533136 + 12.645136 \\ & + 0 + 30.869136 + 0 + 0 + 0 \end{aligned}$$

$$\text{Var} = 52.4685$$

$$\begin{aligned} \text{IDM} = & 0/(1+(1-1)^2) + 1/(1+(1-2)^2) \\ & + 1/(1+(1-3)^2) + 1/(1+(2-1)^2) \\ & + 0/(1+(2-2)^2) + 1/(1+(2-3)^2) + \\ & 0/(1+(3-1)^2) + 0/(1+(3-2)^2) + \\ & 0/(1+(3-3)^2) \end{aligned}$$

$$\text{IDM} = 0 + 0.5 + 0.2 + 0.5 + 0 + 0.5 + 0 + 0 + 0 = 1.7$$

$$\begin{aligned} \text{H} = & 0/(1+(1-1)^2) + 1/(1+(1-2)^2) \\ & + 1/(1+(1-3)^2) + 1/(1+(2-1)^2) \\ & + 0/(1+(2-2)^2) + 1/(1+(2-3)^2) + \\ & 0/(1+(3-1)^2) + 0/(1+(3-2)^2) + \\ & 0/(1+(3-3)^2) \end{aligned}$$

$$\text{H} = 0 + 0.5 + 0.2 + 0.5 + 0 + 0.5 + 0 + 0 + 0 = 1.7$$

$$\begin{aligned} \text{D} = & 0 * |0 - 0| + 1 * |0 - 1| + 1 * |0 - 2| + \\ & 1 * |1 - 0| + 0 * |1 - 1| + 1 * |1 - 2| + \\ & 0 * |2 - 0| + 0 * |2 - 1| + 0 * |2 - 2| \end{aligned}$$

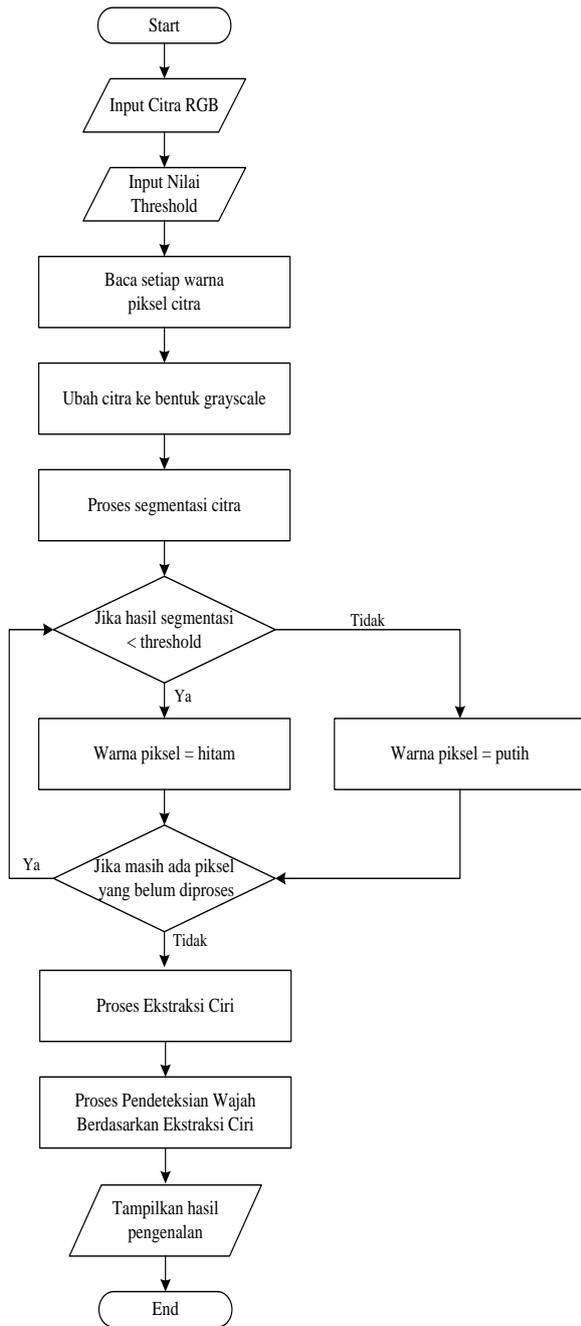
$$\text{D} = 0 + 1 + 2 + 1 + 0 + 1 + 0 + 0 + 0 = 5$$

Jadi, total nilai ekstraksi ciri dari citra input adalah :

$$\begin{aligned} & = 0.444 + 9 + 52.4685 + 1.7 + 1.7 + 5 \\ & = 70.3125 \end{aligned}$$

Setelah memperoleh nilai ciri (*feature*) dari wajah, maka akan dibandingkan dengan nilai ciri wajah yang tersimpan dalam *database*. Selisih nilai terkecil yang diperoleh merupakan solusi dari proses pengenalan wajah.

Proses kerja dari sistem yang dibuat dapat dirincikan sebagai berikut:



Gambar 4. Analisis Proses Pengenalan Wajah dengan Metode *Feature Reduction*

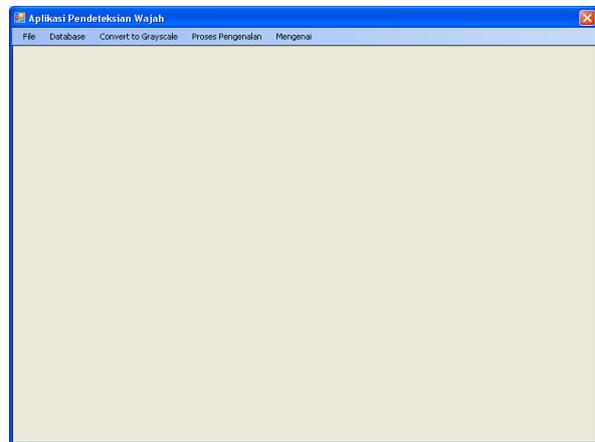
## HASIL DAN PEMBAHASAN

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



Gambar 7. Tampilan *Login*

Apabila data yang dimasukkan merupakan data 'Admin' maka sistem langsung menampilkan *form* Main seperti gambar berikut:



Gambar 8. Tampilan Utama untuk 'Admin'

Pada tampilan utama ini ada beberapa menu yang berfungsi untuk mengakses *form-form* yang terdapat dalam sistem. Berikut perincian dari menu yang terdapat dalam sistem:

1. Menu File, terdiri dari:

*Submenu New*, yang berfungsi untuk mengosongkan citra yang ditampilkan. *Submenu Open Image*, yang fungsinya menampilkan kotak dialog *open* untuk memilih *file* citra yang ingin

dicocokkan. *Submenu Save Image*, yang berfungsi untuk menyimpan citra yang sedang ditampilkan. *Submenu Exit*, yang berfungsi untuk menutup aplikasi. Tampilan menu File ada di gambar berikut:



Gambar 9. Tampilan Menu 'File'

2. Menu Database, terdiri dari:

Sub menu Tampil, yang berfungsi untuk menampilkan *form* Tampil. *Form* ini digunakan untuk menampilkan gambar identitas yang terdaftar pada *database*.

Sub menu Tambah, yang fungsinya menampilkan *form* Tambah. *Form* tersebut digunakan untuk menambahkan gambar identitas baru ke *database*.

Tampilan menu Database dapat dilihat di gambar berikut:



Gambar 10. Tampilan Menu 'Database'

3. Menu *Convert to Grayscale*, yang berfungsi untuk mengkonversi citra yang akan dicocokkan dari format RGB ke format *grayscale*.

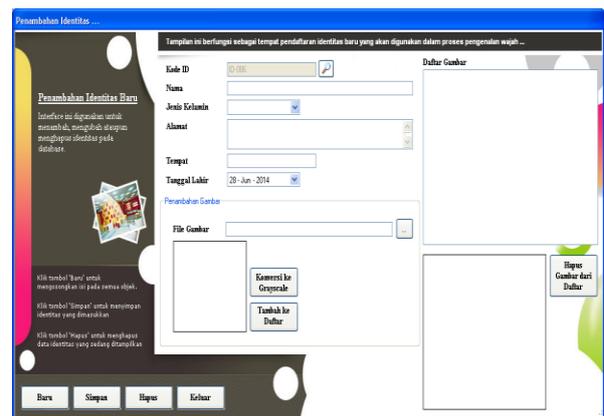
4. Menu Proses Pengenalan, yang fungsinya menampilkan *form* Pengenalan. *Form* tersebut dipergunakan untuk menampilkan hasil pengenalan.

5. Menu Mengenai, yang fungsinya untuk menampilkan data pribadi dari pembuat perangkat lunak.

Khusus untuk pemakai tipe 'Admin', dapat mengakses menu *Database*. Menu ini menyediakan dua buah tampilan, yaitu:

1. *Form* 'Tambah'

*Form* 'Tambah' dipakai untuk menambahkan, merubah ataupun menghapus data identitas yang tersimpan pada *database*. *Form* ini dapat ditampilkan dengan mengakses menu 'Database' >> 'Tambah' lalu sistem akan langsung menampilkan *form* berikut:



Gambar 11. Tampilan *Form* 'Tambah'

2. *Form* 'Tampil'

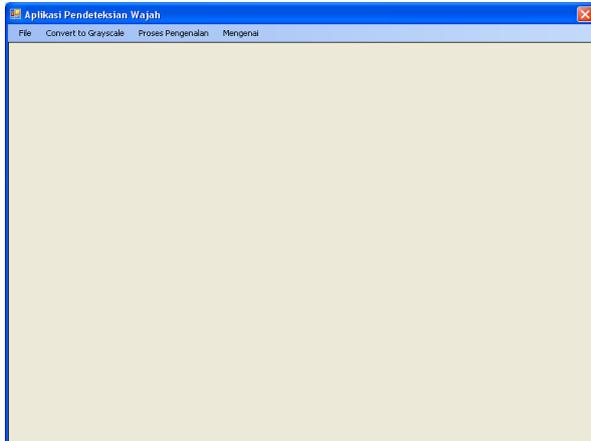
Fungsi *form* ini untuk menampilkan semua daftar identitas yang terdaftar

pada *database*. Tampilan *form* Tampil pada gambar berikut:



Gambar 12. Tampilan *Form* Tampil

Apabila data yang dimasukkan merupakan data '*User*' maka sistem akan langsung menampilkan *form* *Main* seperti gambar berikut:



Gambar 13. Tampilan Utama untuk '*User*'

Hasil pengujian yang diperoleh dapat dirincikan sebagai berikut:

Tabel 4. Hasil Pengujian

No	Citra Uji	Citra Basis Data ke	Citra Terdekat	Jrk Euclidean	Ket
----	-----------	---------------------	----------------	---------------	-----

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
6	Uji1.jpg	ID-006	013.jpg	0.446116	Benar
7	Uji2.jpg	ID-007	P025.jpg	0.379077	Salah
8	Uji3.jpg	ID-006	P025.jpg	0.3472258	Salah
9	Uji4.jpg	ID-006	028.jpg	0.2818669	Benar
10	Uji5.jpg	ID-006	P025.jpg	0.245804	Salah
11	Uji6.jpg	ID-006	028.jpg	0.279546	Benar
12	Uji7.jpg	ID-006	030.jpg	0.192178	Benar
13	1.jpg	ID-007	P019.jpg	0.2189163	Benar
14	2.jpg	-	-	0.41917014	Salah
15	3.jpg	ID-007	P016.jpg	0.3542404	Benar
16	4.jpg	-	-	0.40144089	Salah
17	5.jpg	ID-007	P015.jpg	0.23773748	Benar
18	6.jpg	ID-007	P021.jpg	0.22398212	Benar
19	7.jpg	-	-	0.447087926	Salah

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

Dimensi citra input harus berukuran 240 x 240 dan harus berupa citra *grayscale*.

Jumlah sampel yang digunakan harus banyak. Hal ini diperlukan agar bisa meningkatkan akurasi hasil pengenalan. Namun, jumlah sampel yang banyak akan mengakibatkan proses pengenalan wajah menjadi lama.

Berdasarkan hasil pengujian, diperoleh informasi bahwa akurasi dari metode yang digunakan adalah  $11/19 * 100\% = 57.895\%$ .

Metode yang digunakan tidak dapat mengenali gambar wajah yang dimasukkan apabila ukuran kepala pada gambar tersebut berbeda dengan ukuran kepala pada gambar di *database*.

Untuk meningkatkan akurasi dari metode yang digunakan, maka dapat dikombinasikan dengan algoritma pendeteksian wajah sehingga proses pengenalan wajah hanya terhadap daerah wajah saja.

## SIMPULAN

Setelah menyelesaikan pembuatan perangkat lunak ini, penulis dapat menarik beberapa kesimpulan sebagai berikut:

1. Dimensi citra input harus berukuran 240 x 240 dan harus berupa citra *grayscale*.

2. Jumlah sampel yang digunakan harus banyak. Hal ini diperlukan agar langsung bisa meningkatkan akurasi hasil 5pengenalan. 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*.

## DAFTAR PUSTAKA

- Clifford Gower, John. Euclidean Distance Matrix, 2006.
- Lim, R. Reinders, M.J.T. and Thiang., Pengenalan Citra Wajah dengan Pemrosesan Awal Transformasi Wavelet, Jurnal, vol. 02, 2000.
- Purnomo, M. H. dan A. Muntasa, Konsep Pengolahan Citra Digital dan Ekstrasi Fitur, Graha Ilmu, Yogyakarta, 2010.
- Mullen, T., & Collier, N. (2004). "Sentiment analysis using support vector machines with diverse information sources". In Proceedings EMNLP'04, pp. 412-418.
- Tan, S. B., & Zhang, J. (2008). An Empirical study of opinion analysis for Chinese documents. Expert Systems with Application, 34(4), 2622-2629.
- Whitelaw C., Garg N., & Argamon, S. (2005). "Using appraisal groups for sentiment analysis". In Proceedings of the 14th ACM CIKM, pp. 625-631.
- Wijaya, C. M. dan A. Prijono, Pengolahan Citra Digital, Informatika Bandung, Bandung, 2007.