

**PENGENALAN IRIS MATA DENGAN MENGGUNAKAN  
TRANSFORMASI *HAAR WAVELET* DAN *HAMMING*  
*DISTANCE***

**SKRIPSI**

oleh:  
**NUR MAULIDIYAH**  
**0510963043 - 96**



**PROGRAM STUDI ILMU KOMPUTER  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
MALANG  
2010**

**PENGENALAN IRIS MATA DENGAN MENGGUNAKAN  
TRANSFORMASI *HAAR WAVELET* DAN *HAMMING*  
*DISTANCE***

**SKRIPSI**

Sebagai salah satu syarat untuk memperoleh  
gelar Sarjana dalam bidang Ilmu Komputer

oleh:

**NUR MAULIDIYAH**  
**0510963043 - 96**



**PROGRAM STUDI ILMU KOMPUTER  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
MALANG  
2010**

UNIVERSITAS BRAWIJAYA



**LEMBAR PENGESAHAN SKRIPSI**

**PENGENALAN IRIS MATA DENGAN MENGGUNAKAN  
TRANSFORMASI *HAAR WAVELET* DAN *HAMMING  
DISTANCE***

Oleh:  
**NUR MAULIDIYAH**  
**0510963043 - 96**

Setelah dipertahankan di depan Majelis Penguji  
pada tanggal 5 Februari 2010  
dan dinyatakan memenuhi syarat untuk memperoleh  
gelar Sarjana dalam bidang Ilmu Komputer

Pembimbing I

Pembimbing II

Drs. Marji, MT  
NIP. 196708011992031001

Candra Dewi, S.Kom, M.Sc  
NIP. 197711142003122001

Mengetahui,  
Ketua Jurusan Matematika  
Fakultas MIPA Universitas Brawijaya

Dr. Agus Suryanto, M.Sc  
NIP. 196908071994121001

UNIVERSITAS BRAWIJAYA



## LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

**Nama** : Nur Maulidiyah  
**NIM** : 0510963043-96  
**Jurusan** : Matematika  
**Penulis Tugas Akhir berjudul** : PENGENALAN IRIS MATA  
DENGAN MENGGUNAKAN TRANSFORMASI *HAAR WAVELET*  
DAN *HAMMING DISTANCE*

Dengan ini menyatakan bahwa :

1. Isi dari Tugas Akhir yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka dalam Tugas Akhir ini.
2. Apabila dikemudian hari ternyata Tugas Akhir yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 5 Februari 2010

Yang menyatakan,

(Nur Maulidiyah)

NIM. 0510963043

UNIVERSITAS BRAWIJAYA





# PENGENALAN IRIS MATA DENGAN MENGGUNAKAN TRANSFORMASI *HAAR WAVELET* DAN *HAMMING DISTANCE*

## ABSTRAK

Kebutuhan manusia akan keamanan sistem semakin meningkat dan itu bisa diperoleh dari sistem yang memiliki kunci yang berbeda untuk setiap orang. Untuk menjawab permintaan tersebut, belakangan ini berkembang teknologi yang berkaitan dengan sistem pengenalan manusia dengan salah satu anggota tubuh fisik / perilaku yang unik yang dikenal dengan biometrik. Sistem biometrik mengidentifikasi secara otomatis dari individu berdasarkan ciri atau karakteristik yang berbeda yang dimiliki setiap individu. Saat ini, pengenalan iris merupakan teknologi biometrik yang relatif baru dan diprediksi kebutuhannya semakin meningkat. Oleh karena itu, penting untuk melakukan penelitian yang bertujuan merancang sistem pengenalan iris mata, mengetahui tingkat akurasi dari sistem yang dirancang dan mengetahui faktor yang mempengaruhi tingkat akurasi.

Sistem pengenalan iris pada penelitian ini terdiri dari proses lokalisasi iris dari bagian mata lainnya, normalisasi, ekstraksi ciri menggunakan transformasi *haar wavelet* dan proses pengenalan dilakukan dengan menggunakan *hamming distance*.

Data citra mata keabuan yang digunakan dalam penelitian ini diambil dari basis data MMU (Multi Media University). Hasil penelitian menunjukkan pengujian terhadap data latih didapatkan tingkat akurasi sebesar 100% sedangkan pengujian terhadap data uji *intraclass* didapatkan tingkat akurasi terbaik adalah 75%.

Untuk pengujian pada data asing (*interclass*), *Equal Error rate* (EER) paling kecil adalah 0,05 yakni pada pengujian dengan ambang nilai *hamming distance* 0,35. Adanya bulu mata, kelopak mata, pemantulan cahaya dan posisi pengambilan citra mempengaruhi proses pengenalan iris mata.

Kata kunci: biometrik, pengenalan iris, transformasi *haar wavelet*, *hamming distance*.



UNIVERSITAS BRAWIJAYA



# IRIS RECOGNITION USING HAAR WAVELET TRANSFORM AND HAMMING DISTANCE

## ABSTRACT

The human need for system security will continue to increase and it can be obtained from the system that has a different key for each person. To answer the request, recently developed technologies related to the human recognition system with one of the physical body / unique behavior, known as biometrics. Biometric systems are automatically identified based on the characteristics of the individual or the different characteristics of every individual. Current iris recognition technology is a relatively new biometry and predicted needs are increasing. It is therefore important to conduct research aimed at designing iris recognition system, to know the accuracy of the designed system and to know the factors that affect the level of accuracy.

Iris recognition system in this research consisted of the localization process of the iris from the other part of eye, feature extraction using wavelet haar transformation is stored as a template and recognition using the Hamming distance in the iris template.

Gray eye image data used in this research was taken from the database MMU (Multi Media University). The results showed the accuracy of the test of practice data obtained 100% accuracy rate and of intraclass data obtained the best accuracy rate was 75%. To test the foreign data (interclass), smallest Equal Error rate (EER) is 0.05 with hamming distance threshold value of 0.35. The existence of eyelashes, eyelids, reflecting light and image-making positions affect the iris recognition process.

**Keywords:** biometrics, iris recognition, haar wavelet transformation, Hamming distance.

UNIVERSITAS BRAWIJAYA



## Kata Pengantar

Puji syukur penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan segala rahmat, karunia dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir dengan judul “PENGENALAN IRIS MATA DENGAN MENGGUNAKAN TRANSFORMASI *HAAR WAVELET* DAN *HAMMING DISTANCE* “. Tugas akhir ini disusun dan diajukan sebagai syarat untuk memperoleh gelar sarjana pada program studi Ilmu Komputer, jurusan Matematika, fakultas MIPA, universitas Brawijaya.

Salawat serta salam semoga tetap tercurahkan kepada Baginda Rasulullah Muhammad SAW, makhluk paling mulia yang senantiasa memberikan cahaya petunjuk, seorang uswatun hasanah yang telah membawa agama Allah yaitu agama Islam menjadi agama yang Rahmatan Lil ‘Alamin.

Dalam penyelesaian tugas akhir ini, penulis telah mendapat begitu banyak bantuan baik moral maupun materiil dari banyak pihak. Atas bantuan yang telah diberikan, penulis ingin menyampaikan penghargaan dan ucapan terima kasih yang sedalam-dalamnya kepada:

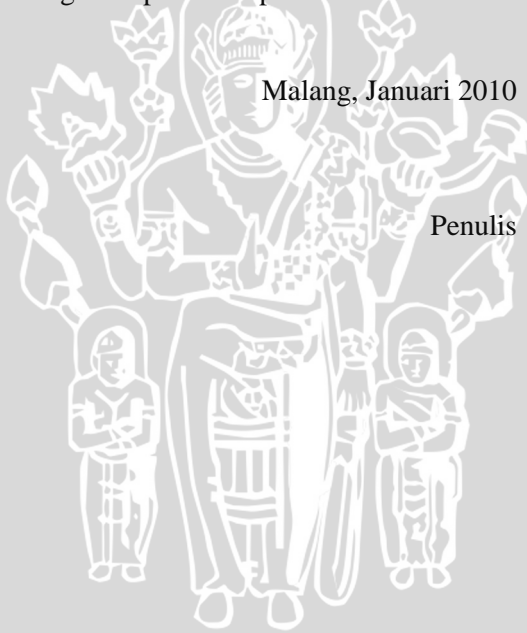
1. Drs Marji, M.T selaku pembimbing utama penulisan skripsi ini sekaligus Ketua Program Studi Ilmu Komputer Jurusan Matematika FMIPA Universitas Brawijaya. Terima kasih atas semua saran, kritik, waktu, bantuan, dorongan semangat dan bimbingannya.
2. Candra Dewi S.Kom, M.Sc selaku pembimbing pedamping dalam penulisan skripsi ini. Terima kasih semua saran, kritik, waktu, bantuan, dorongan semangat dan bimbingannya.
3. Bondan Sapto Prakoso, ST selaku Penasihat Akademik.
4. Dr. Agus Suryanto, MSc selaku Ketua Jurusan Matematika.
5. Segenap bapak dan ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada penulis.
6. Segenap staf dan karyawan di Jurusan Matematika FMIPA Universitas Brawijaya.

7. Orang tua, mertua, adik, kakak, nenek dan saudara penulis yang tak pernah berhenti memberikan doa dan dukungannya kepada penulis.
8. Suami tercinta, M.Firdaus yang selalu memberi semangat, membantu, dan mendampingi penulis.
9. Sahabat-sahabat di Program Studi Ilmu Komputer Universitas Brawijaya angkatan 2002-2008 yang telah banyak memberikan bantuannya demi kelancaran pelaksanaan penyusunan tugas akhir ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini banyak kekurangan, untuk itu saran dan kritik yang membangun demi kesempurnaan penulisan selanjutnya sangat penulis harapkan. Semoga skripsi ini dapat bermanfaat untuk semua pihak.

Malang, Januari 2010

Penulis



# DAFTAR ISI

Halaman

Halaman Judul .....	i
Halaman Pengesahan .....	iii
Halaman Pernyataan.....	v
Abstrak .....	vii
Abstract .....	ix
Kata Pengantar .....	xi
Daftar Isi .....	xiii
Daftar Gambar .....	xvii
Daftar Tabel .....	xix

## **BAB I PENDAHULUAN** .....

1

1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian.....	3
1.6 Metodologi Penelitian.....	3
1.7 Sistematika Penulisan.....	4

## **BAB II TINJAUAN PUSTAKA** .....

7

2.1 Identifikasi Manusia(Biometrik) .....	7
2.2 Iris .....	7
2.3 Pengenalan Iris .....	8
2.3.1 Lokalisasi Iris .....	9
2.3.1.1 Region Growing .....	10
2.3.1.2 Erosi .....	10
2.3.1.3 Dilasi .....	10
2.3.2 Normalisasi .....	11
2.3.2.1 Perhitungan Parameter Koordinat Polar.....	12
2.3.2.2 Pembentukan Citra Polar.....	12
2.3.3 Ekstraksi Fitur .....	13
2.3.3.1 Transformasi <i>Haar Wavelet</i> .....	14
2.3.3.2 Dekomposisi Koefisien <i>Haar Wavelet</i> .....	14
2.3.3.3 Transformasi <i>Haar Wavelet pada Citra</i> .....	16
2.3.4 Hamming Distance .....	17



2.4 Metode Pengujian .....	18
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>19</b>
3.1 Langkah-Langkah Penelitian .....	19
3.1.1 Studi Literatur .....	20
3.1.2 Pengumpulan Data Lapangan.....	20
3.1.3 Implementasi Perangkat Lunak.....	20
3.1.3.1 Lokalisasi Iris .....	20
3.1.3.1.1 Pencarian Pupil .....	21
3.1.3.1.2 Pencarian Iris .....	26
3.1.3.1.3 Rekonstruksi Citra.....	28
3.1.3.2 Normalisasi .....	29
3.1.3.3 Ekstraksi Ciri dengan Transformasi Haar Wavelet .....	30
3.1.3.4 Pencocokan dengan <i>Hamming Distance</i> .....	32
3.1.4 Pengujian Perangkat Lunak .....	35
3.1.4.1 Pengujian dengan Nilai <i>Hamming Distance</i> Terkecil .....	35
3.1.4.2 Pengujian dengan Ambang <i>Hamming Distance</i> .....	36
3.2 Contoh Perhitungan Manual .....	37
3.2.1 Perhitungan Transformasi <i>Haar Wavelet</i> .....	37
3.2.2 Perhitungan nilai <i>Hamming Distance</i> .....	41
<b>BAB IV IMPLEMENTASI DAN PEMBAHASAN .....</b>	<b>43</b>
4.1 Lingkungan Implementasi.....	43
4.2 Implementasi Basis Data.....	43
4.3 Implementasi Program .....	44
4.3.1 Lokalisasi Iris .....	44
4.3.1.1 Pencarian Pupil.....	44
4.3.1.2 Pencarian Iris.....	48
4.3.1.3 Rekonstruksi Look Up .....	52
4.3.2 Normalisasi .....	54
4.3.3 Transformasi <i>Haar Wavelet</i> .....	55
4.3.4 Pencocokan Iris .....	61
4.4 Implementasi Antar Muka .....	63
4.5 Implementasi Uji Coba .....	65
4.5.1 Hasil Pengujian FIR .....	65
4.5.1.1 Hasil Pengujian FIR terhadap 5 Data Latih .....	65
4.5.1.2 Hasil Pengujian FIR terhadap 10 Data Latih .....	66
4.5.1.3 Hasil Pengujian FIR terhadap 15 Data Latih .....	68
4.5.1.4 Hasil Pengujian FIR terhadap 20 Data Latih .....	71



4.5.2 Hasil Pengujian FOR .....	74
4.5.2.1 Hasil Pengujian FOR terhadap 5 Data Uji .....	74
4.5.2.2 Hasil Pengujian FOR terhadap 10 Data Uji .....	75
4.5.2.3 Hasil Pengujian FOR terhadap 15 Data Uji .....	77
4.5.2.4 Hasil Pengujian FOR terhadap 20 Data Uji .....	80
4.5.2.5 Hasil Pengujian FOR terhadap 105 Data Uji .....	83
4.5.3 Hasil Pengujian dengan Nilai Batas <i>Hamming Distance</i> .....	89
4.5.3.1 Hasil Pengujian Terhadap Data Uji Asing ( <i>interclass</i> ) .....	89
4.5.3.1.1 Hasil Pengujian dengan Nilai Batas 0,26 .....	89
4.5.3.1.2 Hasil Pengujian dengan Nilai Batas 0,35 .....	92
4.5.3.1.3 Hasil Pengujian dengan Nilai Batas 0,41 .....	94
4.5.3.2 Hasil Pengujian Terhadap Data Latih.....	96
4.5.3.3 Hasil Pengujian Terhadap Data <i>Intraclass</i> .....	97
4.5.3.3.1 Hasil Pengujian dengan Nilai Batas 0,26.....	99
4.5.3.3.2 Hasil Pengujian dengan Nilai Batas 0,35.....	100
4.5.3.3.3 Hasil Pengujian dengan Nilai Batas 0,41.....	100
4.6 Analisis hasil.....	102
4.6.1 Analisis Hasil Pengujian dengan Nilai <i>Hamming Distance</i> terkecil.....	100
4.6.1.1 Analisis Hasil Pengujian terhadap Data Latih.....	100
4.6.1.2 Analisis Hasil Pengujian terhadap Data Uji .....	102
4.6.2 Analisis Hasil Pengujian dengan Ambang nilai Hamming Distance.....	105
 <b>BAB V KESIMPULAN DAN SARAN</b> .....	 107
5.1 Kesimpulan.....	107
5.2 Saran.....	108
 <b>DAFTAR PUSTAKA</b> .....	 109
<b>LAMPIRAN</b> .....	111

UNIVERSITAS BRAWIJAYA



## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Penampakan Iris pada Mata.....	7
Gambar 2.2 Anatomi Iris Mata.....	8
Gambar 2.3 Citra Mata Setelah Proses Binerisasi .....	9
Gambar 2.4 Iris Setelah Dilakukan Dilasi.....	11
Gambar 2.5 Iris Setelah Dilokalisasi Sempurna.....	11
Gambar 2. 6 Daughman Rubber Shet.....	12
Gambar 2. 7 Transformasi <i>Haar Wavelet</i> .....	15
Gambar 2. 8 Proses Transformasi <i>Haar Wavelet</i> .....	16
Gambar 2. 9 Citra Iris Setelah Diekstraksi .....	16
Gambar 3.1 <i>Flowchart</i> Langkah-Langkah Penelitian .....	19
Gambar 3.2 Proses Pengenalan Citra Iris.....	20
Gambar 3.3 <i>Flowchart</i> Tahapan Lokalisasi Iris .....	21
Gambar 3.4 <i>Flowchart</i> Tahapan Pencarian Pupil .....	22
Gambar 3.5 <i>Flowchart</i> Proses Binerisasi.....	23
Gambar 3.6 <i>Flowchart</i> Proses Erosi pada Pencarian Pupil ...	24
Gambar 3.7 <i>Flowchart</i> Proses Perhitungan Jari-Jari.....	25
Gambar 3.8 <i>Flowchart</i> Tahapan Pencarian Iris.....	26
Gambar 3.9 <i>Flowchart</i> Proses Erosi Dilasi .....	28
Gambar 3.10 <i>Flowchart</i> Rekonstruksi Look Up .....	29
Gambar 3.11 Ilustrasi Pembacaan Data .....	30
Gambar 3.12 <i>Flowchart</i> Transformasi <i>Haar Wavelet</i> pada Citra .....	31
Gambar 3.13 Contoh Template Iris .....	32
Gambar 3.14 Ilustrasi proses <i>shifting</i> (penggeseran) <i>hamming</i> <i>distance</i> .....	33
Gambar 3.15 <i>Flowchart</i> Pencocokan Iris .....	34
Gambar 3.16 Matriks 8X8.....	37
Gambar 3.17 Matriks Hasil Dekomposisi Baris Level 1.....	38
Gambar 3.18 Matriks Hasil Dekomposisi Baris Level 2.....	38
Gambar 3.19 Matriks Hasil Dekomposisi Baris Level 3.....	39
Gambar 3.20 Matriks Hasil Dekomposisi Kolom Level 1 .....	39
Gambar 3.21 Matriks Hasil Dekomposisi Kolom Level 2 .....	40
Gambar 3.22 Matriks Hasil Dekomposisi Kolom Level 3 .....	40

Gambar 4.1 Struktur Basis Data .....	44
Gambar 4.2 Gambar Citra .....	53
Gambar 4.3 Hasil Normalisasi Citra Iris .....	55
Gambar 4.4 Hasil Crop Citra Normalisasi .....	55
Gambar 4.5 <i>Template</i> Iris .....	56
Gambar 4.6 <i>Template</i> iris hasil dekomposisi baris level 4.....	58
Gambar 4.7 <i>Template</i> iris hasil dekomposisi kolom level 4.....	60
Gambar 4.8 <i>Template</i> iris 1 dimensi.....	61
Gambar 4.9 Form Utama dan Data Latih .....	64
Gambar 4.10 Diagram batang persebaran nilai <i>hamming distance</i> pada semua data.....	87
Gambar 4.11 Diagram batang persebaran nilai <i>hamming distance</i> pada data yang dikenali benar.....	87
Gambar 4.12 Diagram batang persebaran nilai <i>hamming distance</i> pada data yang dikenali salah.....	88
Gambar 4.13 Grafik Perbandingan Nilai FIR dengan Jumlah Data Latih.....	101
Gambar 4.14 Grafik Perbandingan Nilai FOR dengan Jumlah Data Uji.....	103



## DAFTAR TABEL

	Halaman
Tabel 3.1 Tabel Data.....	34
Tabel 3.2 Rancangan Tabel Pengujian Citra Mata.....	35
Tabel 3.3 Rancangan Tabel Pengujian Citra Mata 2.....	36
Tabel 4.1 Tabel Hasil Pengujian FIR terhadap 5 Data Latih.....	65
Tabel 4.2 Tabel Hasil Pengujian FIR terhadap 10 Data Latih.....	66
Tabel 4.3 Tabel Hasil Pengujian FIR terhadap 15 Data Latih.....	68
Tabel 4.4 Tabel Hasil Pengujian FIR terhadap 20 Data Latih.....	70
Tabel 4.5 Tabel Hasil Pengujian FIR terhadap 5 Data Latih.....	74
Tabel 4.6 Tabel Hasil Pengujian FIR terhadap 10 Data Latih.....	75
Tabel 4.7 Tabel Hasil Pengujian FIR terhadap 15 Data Latih.....	77
Tabel 4.8 Tabel Hasil Pengujian FIR terhadap 20 Data Latih.....	80
Tabel 4.9 Tabel Hasil Pengujian FIR terhadap 105 Data Latih.....	83
Tabel 4.10 Tabel perbandingan nilai <i>hamming distance</i> .....	88
Tabel 4.11 Tabel Hasil Pengujian FAR terhadap 75 Data Uji Asing.....	89
Tabel 4.12 Tabel Hasil Pengujian FAR terhadap 75 Data Uji Asing2.....	92
Tabel 4.13 Tabel Hasil Pengujian FAR terhadap 75 Data Uji Asing3.....	94
Tabel 4.14 Tabel Hasil Pengujian FRR terhadap 75 data uji yang berasal dari individu yang sama dengan data latih.....	97
Tabel 4.15 Tabel Perbandingan dengan Penelitian Lain.....	101
Tabel 4.16 Tabel Citra Mata.....	104

UNIVERSITAS BRAWIJAYA





# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Kebutuhan manusia akan keamanan sistem semakin hari semakin meningkat. Dan itu bisa diperoleh dari sistem yang memiliki kunci unik untuk setiap orang. Untuk menjawab permintaan tersebut, belakangan ini berkembang teknologi yang berkaitan dengan sistem pengenalan manusia dengan salah satu anggota tubuh fisik / perilaku yang unik yang dikenal dengan biometrik (Agus, 2004).

Salah satu bagian tubuh pada manusia yang unik adalah iris mata. Iris mata orang yang satu dengan yang lain tidak ada yang sama bahkan anak kembar identik sekalipun. Selain itu, iris mata memiliki beberapa kelebihan dibanding sidik jari maupun wajah yang sering digunakan untuk biometrik. Kelebihan iris yakni iris berada di dalam mata dan dilindungi oleh kelopak mata sehingga keunikannya bertahan lama, sejak umur 1 tahun iris mata seseorang tidak pernah berubah. Hal ini sangat berbeda dengan sidik jari dan wajah yang berkemungkinan mengalami cacat, kecelakaan, maupun dipalsukan (Chen, 2005).

Pengenalan iris menjadi bagian dari biometrik yang sangat penting untuk dijadikan penelitian untuk mencari metode apa yang paling tepat digunakan. Mengingat kebutuhan akan sistem identifikasi personal menggunakan iris di beberapa tahun mendatang akan menguat.

Dalam proses pengenalan iris ada beberapa tahap yang dilakukan. Secara garis besar ada 3 tahap utama yakni lokalisasi iris, ekstraksi ciri dan pencocokkan. Penelitian tentang identifikasi personal berdasarkan iris mata ini pernah dilakukan oleh daughman dengan metode ekstraksi gabor wavelet, wilde dengan piramida replace, dan Bole dengan zero crossing transformasi *wavelet* (Agus, 2004).

Salah satu metode untuk melakukan ekstraksi ciri adalah dengan transformasi *haar wavelet* yang memiliki



keuntungan lebih dibandingkan transformasi fourier (Masek, 2003). Transformasi *haar wavelet* mulai diperkenalkan pada tahun 1980 oleh Morlet dan Grossman sebagai fungsi matematis untuk merepresentasikan data atau fungsi sebagai alternatif transformasi-transformasi matematika yang lahir sebelumnya untuk menangani masalah resolusi. Pada tahun 1998, salah satu jenis wavelet yakni *zero crossing* transformasi dicoba untuk pengenalan iris pada tahap ekstraksi ciri oleh Bole (Agus, 2004).

Jenis wavelet yang lain adalah *Haar Wavelet* Transform. *Haar Wavelet* digunakan karena memiliki konsep yang sederhana, prosesnya cepat, efisiensi memori, dan dapat dibalikkan dengan tepat tanpa *edge effects* yang merupakan masalah bagi transformasi *wavelet* yang lainnya (Kaplan, 2004)

Menurut Masek, 2003 menyatakan salah satu metode untuk pencocokan pola iris mata adalah *hamming distance*. *Hamming distance* dicetuskan oleh Daughman dan terbukti efektif dalam proses pencocokan iris mata.

Sehingga menjadi sangat menarik mengetahui tingkat akurasi pengenalan iris dengan menggabungkan metode transformasi *haar wavelet* untuk proses ekstraksi ciri dan *hamming distance* untuk pencocokan.

Berdasarkan latar belakang tersebut, penulis mengangkat tema **"Pengenalan Iris Mata dengan Menggunakan Transformasi *Haar Wavelet* dan *Hamming Distance*"** pada skripsi ini.

## 1.2 Rumusan Masalah

Berdasarkan uraian latar belakang masalah di atas, maka dalam skripsi ini dapat dirumuskan permasalahan yang akan dibahas sebagai berikut :

1. Bagaimana meimplementasikan sistem untuk pengenalan iris mata dengan metode transformasi *haar wavelet* dan *hamming distance*.
2. Berapa tingkat akurasi pengenalan iris dengan metode transformasi *haar wavelet* dan *hamming distance*.

### **1.3 Batasan Masalah**

Dari permasalahan di atas, berikut ini diberikan batasan masalah untuk menghindari melebarnya masalah yang akan diselesaikan:

1. Gambar yang diuji berasal dari database MMU (Multi Media University) yang berwarna *grayscale*.
2. Data latih terdiri dari 80 citra mata yang terdiri dari 20 individu dan masing-masing memiliki 4 citra mata yang serta tiap citra berukuran 320X240 piksel.
3. Program digunakan untuk pengguna umum (bukan untuk pengguna lensa kontak dan pengguna khusus lainnya).
4. Pengujian untuk pengenalan iris dilakukan dengan FIR, FOR, FAR dan FRR.

### **1.4 Tujuan Penelitian**

Tujuan penelitian yang ingin dicapai adalah:

1. Mengimplementasikan sistem pengenalan iris mata dengan menggunakan transformasi *haar wavelet* dan *hamming distance*.
2. Mengetahui tingkat akurasi pengenalan iris mata dengan menggunakan transformasi *haar wavelet* dan *hamming distance*.
3. Mengetahui faktor yang mempengaruhi tingkat akurasi.

### **1.5 Manfaat Penelitian**

Manfaat yang bisa didapatkan melalui penelitian ini adalah:

1. Mengenali nama individu dari input gambar mata.
2. Mendapatkan nilai *hamming distance* dari input gambar mata seseorang yang diuji.

### **1.6 Metodologi Penelitian**

Untuk mencapai tujuan yang dirumuskan sebelumnya, maka metodologi yang digunakan dalam penulisan tugas akhir ini adalah:

#### 1. Studi Literatur

Mempelajari teori-teori yang berhubungan dengan konsep pengenalan iris , metode transformasi *haar wavelet* dan *hamming distance* dari berbagai referensi.

#### 2. Pengumpulan Data Lapangan

Mengumpulkan data-data yang diperlukan untuk penelitian.

#### 3. Perancangan dan implementasi sistem

Membuat rancangan model perangkat lunak dengan analisis terstruktur dan mengimplementasikan hasil rancangan tersebut yaitu membuat piranti lunak pengenalan iris mata.

#### 4. Uji coba dan analisis hasil implementasi

Menguji perangkat lunak dengan berbagai image mata. Kemudian menganalisis hasil dari implementasi tersebut sesuai dengan tujuan yang dirumuskan sebelumnya, untuk kemudian dievaluasi dan disempurnakan

### 1.7 Sistematika Penulisan

Skripsi ini disusun berdasarkan sistematika penulisan sebagai berikut:

#### 1. BAB I PENDAHULUAN

Berisi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

#### 2. BAB II TINJAUAN PUSTAKA

Menguraikan teori-teori yang berhubungan dengan biometrik, pengenalan iris mata dan tranformasi wavelet.

#### 3. BAB III METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan mengenai metode-metode yang digunakan dalam pengenalan iris mata.

#### 4. BAB IV HASIL DAN PEMBAHASAN

Bab ini menerangkan proses implementasi dari rancangan penelitian yang dijelaskan pada bab III. Implementasi yang dijelaskan terdiri dari implementasi basis data dan implementasi program. Selain itu, bab ini juga menjelaskan penerapan aplikasi, analisis hasil percobaan

mengenai tingkat keakuratan yang dibandingkan dengan jumlah data yang berhasil dideteksi dan dikenali dengan jumlah data yang gagal dideteksi dan dikenali.

5. **BAB V KESIMPULAN DAN SARAN**

Berisi kesimpulan dari seluruh rangkaian penelitian serta saran kemungkinan pengembangannya.

UNIVERSITAS BRAWIJAYA



UNIVERSITAS BRAWIJAYA





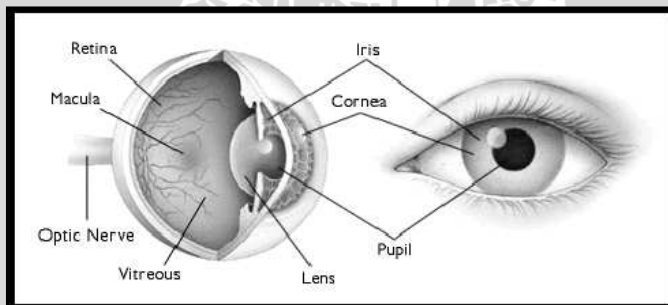
## BAB II TINJAUAN PUSTAKA

### 2.1 Identifikasi Manusia (Biometrik)

Metode lama (tradisional) yang dipakai untuk mengenali identitas manusia berdasarkan pada benda-benda fisik seperti kartu tanda pengenal, kunci, dan lain-lain. Atau juga dari hal yang diketahui manusia seperti pin, password, dan lain sebagainya. Metode ini memiliki kelemahan contohnya kartu tanda pengenal bisa sewaktu-waktu hilang. Begitu pula dengan password bisa lupa. Pada beberapa waktu terakhir, berkembang dengan pesat metode pengenalan personal berdasarkan salah satu anggota tubuh manusia yang unik dan berdasarkan perilaku manusia. Metode pengenalan ini disebut dengan biometrik. Salah satu anggota tubuh yang bersifat unik dan bias dijadikan sebagai pengenal adalah iris atau selaput pelangi pada manusia (Agus,2004).

### 2.2 Iris

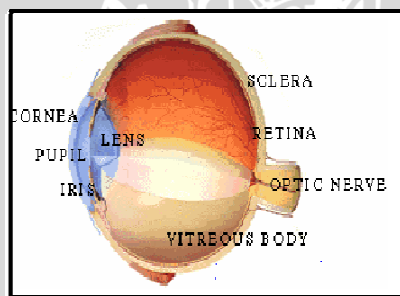
Iris merupakan bagian yang berwarna yang tampak pada bola mata. Bagian iris terlihat sebagai lingkaran mata yang melingkupi bagian hitam pupil dengan warna-warna tertentu.



**Gambar 2.1** Penampakan Iris Pada Bagian Mata  
Sumber : Agus, 2004

Secara anatomi, iris merupakan sebuah organ internal yang dilindungi yang terletak di belakang kornea dan aqueous humour, serta berada di depan lensa mata. Iris merupakan satu-satunya organ internal tubuh yang dapat terlihat dari luar. Iris dapat terlihat cukup jelas pada jarak 1 meter.

Bagian depan dari iris berbentuk tidak teratur, cenderung kasar serta memiliki alur yang tidak rata. Bagian ini dibentuk oleh lapisan yang terdiri dari sel pigmen dan fibroblast. Bagian bawah dari lapisan ini adalah jaringan ikat yang berkadar darah rendah (*poorly vascularized*) dengan beberapa serat, fibroblast dan melanocyte. Bagian selanjutnya merupakan bagian yang kaya akan supply darah tertutup oleh jaringan ikat yang longgar (Junqueira, 1988).



**Gambar 2.2** Anatomi Iris  
Sumber: Junqueira, 1988

Adapun karakteristik iris adalah :

1. Mempunyai bentuk geometri polar, merupakan sistem koordinat yang alami.
2. Mempunyai tingkat ketidakteraturan yang tinggi, dan mempunyai entropi 3,2 bit per milimeter persegi jaringan iris (Fahmi, 2007) .

### **2.3 Pengenalan Iris**

Metode pengenalan iris dapat dibagi menjadi beberapa tahap yakni lokalisasi iris dari area mata,



normalisasi, ekstraksi citra, pencocokan pola mata (Ganorkar, 2007).

### 2.3.1 Lokalisasi Iris

Sebelum melakukan pengenalan iris, maka iris harus dilokalisasi terlebih dahulu. Lokalisasi iris bisa disebut juga proses segmentasi. Lokasi iris ditentukan melalui lokasi komponennya atau dengan batas luar dan dalam iris.

Citra mata mengandung objek-objek selain iris yang dalam proses analisis biometrik tidak dibutuhkan, bahkan terkadang mengganggu. Objek tersebut misalnya kelopak mata, alis mata dan bagian lainnya. Untuk itu diperlukan sebuah teknik pemisahan objek iris dari citra mata keseluruhan. (Mei, 2006)

Metode yang dikembangkan untuk pemisahan iris mata ini dilakukan dengan 2 tahap yakni binerisasi dan rekonstruksi. Binerisasi merupakan tahap untuk diubah dari citra warna menjadi citra *grayscale* lalu menjadi citra biner dengan *threshol* nilai derajat keabuan 128. Hasil dari proses ini adalah bentuk iris mata yang berwarna hitam (pixel 0) dan selainnya berwarna putih (255) (Achmad, 2005).

Proses ini bisa disebut juga tahap segmentasi yang memisahkan iris yang diinginkan dengan bagian lain mata yang tidak dibutuhkan secara global. (Fahmi, 2007). Sedangkan rekonstruksi lookup merupakan proses untuk mengambil hanya bagian iris dengan melakukan rekonstruksi ulang citra mata asli.



**Gambar 2.3** Citra Mata Setelah Proses Binerisasi  
Sumber : fahmi, 2007

### **2.3.1.1 Region Growing**

*Region growing* merupakan salah satu metode sederhana segmentasi citra yang berdasarkan region. Menurut pengertiannya, *region growing* adalah suatu pendekatan untuk menentukan pixel yang bertetangga dari suatu seed serta menentukan apakah suatu pixel ditambahkan ke dalam *seed* tersebut atau tidak. Prinsip metode ini mulanya ialah penentuan sekumpulan *seed point*. Kemudian diinisialisasikan suatu region awal dari *seed* tersebut. Region ini akan terus berkembang dari seed point menjadi point-point yang saling berdekatan tergantung pada kriteria yang dibuat. Kriteria yang dibuat biasanya ditentukan berdasarkan *graylevel*, intensitas, atau warnanya. Dalam proses pengenalan iris, *region growing* digunakan untuk membedakan daerah iris dengan daerah lainnya.

Pada proses pengenalan iris, metode *region growing* digunakan untuk mengelompokkan bagian iris pada region tersendiri yang terpisah dari bagian lain seperti bulu mata dan kelopak mata.

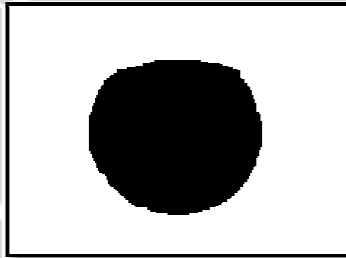
### **2.3.1.2 Erosi**

Pada operasi erosi, ukuran obyek diperkecil dengan mengikis sekeliling obyek. Cara yang dapat dilakukan juga ada dua acara Cara pertama yaitu dengan mengubah semua titik batas menjadi titik latar dan cara kedua dengan menset semua titik di sekeliling titik latar menjadi titik latar. Proses erosi pada sistem pengenalan iris mata digunakan untuk membuang *noise*.

### **2.3.1.3 Dilasi**

Dilasi merupakan kebalikan dari proses erosi. Operasi dilasi dilakukan untuk memperbesar ukuran segmen obyek dengan menambah lapisan di sekeliling obyek. Terdapat 2 cara untuk melakukan operasi ini, yaitu dengan cara mengubah semua titik latar yang bertetangga dengan titik batas menjadi

titik obyek, atau lebih mudahnya set setiap titik yang tetangganya adalah titik obyek menjadi titik obyek. Cara kedua yaitu dengan mengubah semua titik di sekeliling titik batas menjadi titik obyek, atau lebih mudahnya set semua titik tetangga sebuah titik obyek menjadi titik obyek. Pada sistem pengenalan iris mata, proses dilasi dilakukan untuk menyempurnakan bagian iris yang telah dibinerisasi dan telah dilakukan proses erosi.



**Gambar 2.4** Iris Setelah Dilakukan Dilasi



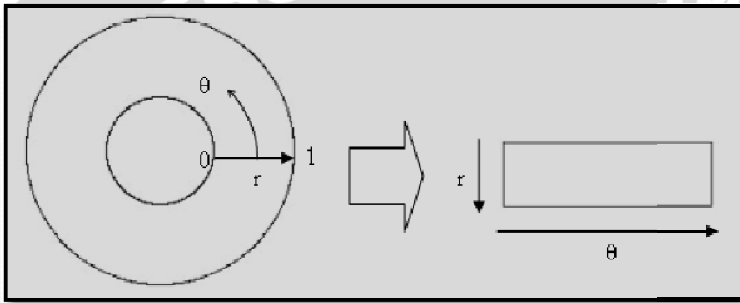
**Gambar 2.5** Iris Setelah Dilokalisasi Secara Sempurna  
Sumber: fahmi , 2007

### 2.3.2 Normalisasi

Bentuk iris yang berupa lingkaran akan sangat menyulitkan untuk dianalisis dan diolah lebih lanjut. Pola susunan piksel yang dianalisis harus mengikuti algoritma

tertentu yang memungkinkan pengambilan piksel dengan bentuk geometri lingkaran.

Hal ini akan sangat merepotkan dan tidak efisien. Untuk mengatasi hal tersebut harus terlebih dahulu mengubah citra iris ke dalam bentuk antara yang sesuai. Perubahan bentuk ini dapat dilakukan dengan melakukan transformasi koordinat dari citra iris dengan Rubber set model Daughman yang disebut dengan normalisasi ( Daughman, 1994).



**Gambar 2.6** Daughman Rubber Set Model

Sumber : Masek, 2003

Langkah dalam melakukan normalisasi yaitu perhitungan parameter dan transformasi polar (Fahmi, 2007).

### 2.3.2.1 Perhitungan Parameter Koordinat Polar

Parameter lingkaran dibutuhkan untuk pembentukan citra polar. Titik pusat dan jari-jari diperlukan sebagai referensi dalam mentransformasikan citra berbentuk lingkaran ke dalam bentuk koordinat polar.

Untuk dapat menghitung parameter-parameter tersebut, terlebih dahulu diambil beberapa titik sampel pada tepian citra yang membentuk pola lingkaran. Jumlah titik sampel yang diambil sangat berpengaruh pada ketelitian pengukuran. Semakin banyak titik sampel yang diambil maka perhitungan yang dilakukan akan semakin teliti. Namun di sisi lain perhitungan akan berjalan sangat rumit dan memakan waktu cukup lama. Untuk itu, pemilihan titik sampel harus

dipertimbangkan secara optimum karena akan sangat berpengaruh pada unjuk kerja sistem nantinya.

- Perhitungan titik pusat

Dapat digunakan rumus dasar lingkaran yang telah diketahui untuk menghitung titik pusat dari lingkaran.

- Perhitungan Jari-jari Lingkaran

Setelah mengetahui titik pusat, jari-jari lingkaran dapat dengan mudah dihitung menurut persamaan berikut :

$$R_n = \sqrt{((X_n - X_c)^2 + (Y_n - Y_c)^2)} \quad , n = 0, 1, \dots, N \quad (2.1)$$

Hasil perhitungan akhir diperoleh dengan mengambil nilai rata-rata dari N buah perhitungan jari-jari yang diperoleh dari N buah titik sampel yang berbeda.

### 2.3.2.2 Pembentukan Citra Polar

Parameter-parameter yang telah dihitung dapat dipakai untuk mengubah bentuk citra iris yang berbentuk lingkaran ke dalam koordinat polar. Citra ditransformasikan ke dalam bentuk polar dengan titik pusat sebagai acuannya. Pada gambar di bawah, lebar data citra hasil transformasi adalah sebesar a-b dan panjang data citra sebesar c-d. Lebar citra sangat tergantung kepada besar jari-jari dalam dan jari-jari luar lingkaran. Sedangkan panjang data citra tergantung kepada besarnya pengambilan piksel tiap derajat lingkaran.

### 2.3.3 Ekstraksi Fitur

fitur adalah objek yang kuantitas dapat diukur dari sebuah pola, Pengklasifikasian berdasar dari masing masing nilai dari fitur-fitur tersebut. Vektor fitur adalah sejumlah atau sekumpulan dari fitur, misalakan sebuah fitur adalah x maka kumpulan fitur dapat direpresentasikan berikut:  $X_1, \dots, X_L$ , memberikan vektor fitur sebagai berikut:  $X_1, \dots, X_L$  (Arifin, 2008).

Menurut Bertalya, 2006 menyatakan fitur adalah segala jenis aspek pembeda, kualitas atau karakteristik. Fitur



bisa berwujud simbolik misalnya warna atau numerik misalnya tinggi.

Ekstraksi fitur merupakan proses perhitungan dan ekstraksi fitur-fitur suatu citra, yang digambarkan dengan vektor fitur multidimensional (Kommit, 2006).

### **2.3.3.1 Transformasi Haar Wavelet**

*Haar wavelet transform* adalah salah satu jenis transformasi *haar wavelet* yang paling lama dikenal. Fungsi Haar sudah digunakan sejak 1910 oleh matematikawan Hungaria bernama Alfred Haar. Tetapi wavelet merupakan alat yang tergolong baru dalam matematika.

### **2.3.3.2 Dekomposisi Koefisien Haar Wavelet**

*Haar wavelet transform* adalah salah satu jenis transformasi *haar wavelet* yang paling lama dikenal. Fungsi Haar sudah digunakan sejak tahun 1910 oleh matematikawan Hungaria bernama Alfred Haar. Tetapi wavelet merupakan alat yang tergolong baru dalam matematika.

*Haar wavelet* bekerja pada data dengan metode transformasi vektor yang disebut *averaging* dan *differencing*. Hasil dari operasi *averaging* disebut *approximation coefficients*. Dan hasil dari operasi *differencing* disebut *detail coefficients*.

Pada wavelet terdapat istilah dekomposisi. Secara umum, dekomposisi wavelet didapatkan melalui penapisan subbidang berkanal dua dengan dua tapis, yaitu tapis pererata atau penyekala atau disebut tapis olos-rendah dan tapis detil atau tapis lolos-tinggi. Pengembangan untuk kasus sinyal berdimensi 2-D (sinyal citra 2-D) biasanya dilakukan dengan menerapkan bank filter secara terpisah terhadap sinyal citra. Biasanya digunakan sebuah tapis lolos-bawah (H) dan tapis lolos-atas (G). Konvolusi citra dengan tapis lolos-bawah menghasilkan sinyal yang biasa disebut dengan citra pendekatan (*approximation image*) dan konvolusi dengan tapis lolos-atas pada arah spesifik menghasilkan citra detil.



Jika terdapat sekumpulan jumlah data *array* (N), maka setengah dari jumlah data (N/2) adalah bagian *approximation coefficients* dan setengah lainnya (N/2) merupakan bagian *detail coefficients*. Nilai *approximation coefficients* akan disimpan pada bagian setengah sebelah atas sedangkan nilai *detail coefficients* akan disimpan pada sisi setengah sebelah bawah yang ditunjukkan pada Gambar 2.4. Nilai *approximation coefficients* akan menjadi input pada proses penghitungan operasi *wavelet* selanjutnya. Prosesnya berupa iterasi yang rekursif hingga diperoleh satu nilai *approximation coefficients* dan *detail coefficients* dalam proses penghitungannya. Proses iterasi merupakan pangkat dari dua ( $2^0, 2^1, 2^2, \dots N/2$ ). (Latha dkk, 2007)



**Gambar 2.7** Transformasi *Haar Wavelet*

Persamaan *Haar Wavelet Transform* untuk mendapatkan nilai *average/approximation coefficients* (A) ditunjukkan pada Persamaan 2.2. Sedangkan untuk mendapatkan nilai *wavelet/detail coefficients* (C) ditunjukkan pada persamaan 2.3 (Stolnitz, 2001).

$$A_i = \frac{d_i + d_{i+1}}{\sqrt{2}} \quad (2.2)$$

$$C_i = \frac{d_i - d_{i+1}}{\sqrt{2}} \quad (2.3)$$

Keterangan:

$A_i$  = average/ approximation coefficients data ke-i

$C_i$  = wavelet/detail coefficients data ke-i

$d_i$  = data ke-i

$i$  = 1,2... N

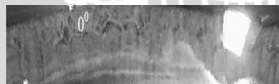
Proses *Haar Wavelet Transform* pada contoh array berdimensi 1x4 dapat ditunjukkan pada Gambar 2.8

$$\begin{array}{l}
 \text{Data} \quad [ \quad d_1 \quad \quad \quad d_2 \quad \quad \quad d_3 \quad \quad \quad d_4 \quad ] \\
 \\
 \text{Level} \quad [ \quad \frac{(\quad)}{\sqrt{\quad}} \quad \quad \quad \frac{(\quad)}{\sqrt{\quad}} \quad \quad \quad \frac{(\quad)}{\sqrt{\quad}} \quad \quad \quad \frac{(\quad)}{\sqrt{\quad}} \quad ] \\
 \text{1} \\
 \\
 \text{Level} \quad [ \quad \frac{\frac{(\quad)}{\sqrt{\quad}} \quad \frac{(\quad)}{\sqrt{\quad}}}{\sqrt{\quad}} \quad \quad \quad \frac{\frac{(\quad)}{\sqrt{\quad}} \quad \frac{(\quad)}{\sqrt{\quad}}}{\sqrt{\quad}} \quad \quad \quad \frac{\frac{(\quad)}{\sqrt{\quad}} \quad \frac{(\quad)}{\sqrt{\quad}}}{\sqrt{\quad}} \quad \quad \quad \dots ] \\
 \text{2}
 \end{array}$$

**Gambar 2.8** Proses Transformasi *Haar Wavelet*

### 2.3.3.4 Transformasi *Haar Wavelet* Pada Citra

Suatu citra dapat dianggap sebagai suatu matriks dua dimensi kemudian dilakukan transformasi terhadap baris-baris pada citra, dan dilanjutkan dengan transformasi terhadap kolom-kolom pada citra



```

0110000001000000000001000111110
1010101010101010101101011111000

```

**Gambar 2.9** Citra Iris Setelah Diekstraksi  
Sumber: Masek, 2003

### 2.3.4 Hamming Distance

*Hamming distance* diberi nama setelah Richard Hamming memperkenalkan karyanya dalam mendeteksi kesalahan dan memperbaiki kesalahan-kode pada tahun 1950. Dia memperkenalkan Hamming kode. Pada informasi awalnya sebagian besar pekerjaan koreksi-kesalahan kode terkonsentrasi pada blok kode (kode di mana semua *codewords* memiliki panjang yang sama).

Jarak (*distance*) Hamming antara dua *codewords* dalam blok kode adalah jumlah simbol oleh dua *word* yang berbeda. Misalnya, *word* 11000010 dan 10010010 adalah terbagi dalam jarak dua, karena mereka berbeda dalam angka biner kedua dan keempat. Dua kata adalah sama jika dan hanya jika jarak *hamming* antara mereka adalah nol. *Codewords* untuk berbagai sumber simbol harus berbeda dari satu sama lain dengan jarak minimal satu kecuali kode tunggal. Kode yang tahan terhadap kesalahan terdiri dari kata-kata yang dipisahkan oleh jarak yang lebih besar. Kebanyakan prosedur *ecoding* menggunakan jarak untuk memilih kriteria yang paling mungkin dimaksudkan *codeword*. Khususnya, ketika menerima kata, maka *decoder* mencari *codeword* yang valid yang paling dekat ke kata yang diterima dalam kemungkinan memiliki jarak hamming terkecil. Jika ada satu yang dekat *codeword*, diasumsikan dimaksudkan untuk menjadi *codeword*. Jika ada ikatan untuk jarak minimum, kesalahan hanya terdeteksi, namun tidak diperbaiki. Misalnya urutan 0001 yang menerima. Untuk membaca sandi ini, dilakukan dengan membaca dalam tabel kode yang dekat. Maka 0000 untuk s1, yang merupakan jarak dari satu urutan yang diterima. Semua *codewords* lainnya memiliki jarak yang lebih besar dari satu dari pola yang diterima. Misalnya, 0100 yang diterima. Ada dua yang dekat *codewords*: 0000 (untuk s1) dan 0110 (untuk S3), baik pada jarak satu. Oleh karena itu, walaupun kesalahan dapat dideteksi, ia tidak dapat diperbaiki dengan menggunakan kriteria minimum-jarak (Ko,2007).

Karakteristik penting dari kode adalah minimum d jarak antara dua *word*. Mudah untuk melihat perbedaan antara

kedua dugaan dalam kasus satu kesalahan. Jika satu kesalahan yang dibuat, dapat dideteksi jika jarak minimum antara *codewords* adalah  $d = 2$ .

Menurut Masek (2003), *hamming distance* pada pencocokan pola iris memberi ukuran berapa banyak bit-bit yang sama dari 2 pola iris. Dalam membandingkan 2 bit pola iris yaitu X dan Y, *hamming distance* mendefinisikannya dengan nilai sebagai berikut:

$$HD = \frac{1}{N} \sum_{j=1}^N X_j (XOR) Y_j \quad (2.5)$$

Untuk menghasilkan pengenalan dilakukan perhitungan *Hamming Distance* dari kode iris baru dan kode yang telah disimpan. Dengan menggunakan operasi XOR sederhana antara pasangan kode akan menghasilkan *Hamming Distancenya* (Agus, 2004).

#### 2.4 Metode Pengujian

Pengujian untuk pengenalan iris dilakukan dengan:

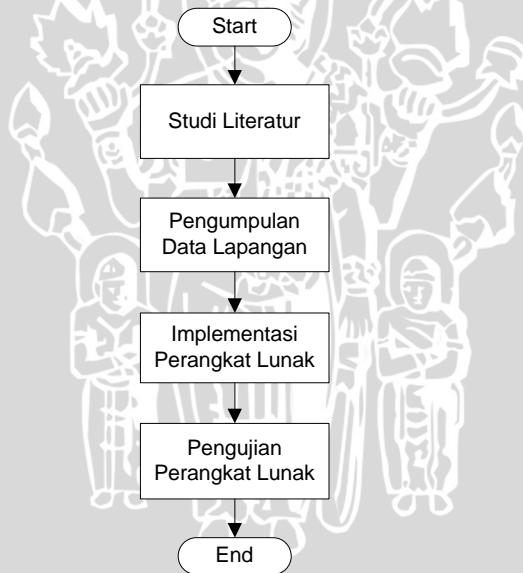
- a. *False inner rate* (FIR) yaitu tingkat kesalahan berdasarkan pengujian dengan citra data latih.
- b. *False outer rate* (FOR) yaitu tingkat kesalahan berdasarkan pengujian terhadap citra data uji *intra class* (dari individu yang sama dengan data latih).
- c. *False Acceptance Rate* (FAR) yaitu tingkat kesalahan penerimaan terhadap data yang seharusnya ditolak karena belum terdaftar.
- d. *False Reject Rate* (FRR) yaitu tingkat kesalahan penolakan terhadap data yang seharusnya dikenali (Woodwards, 2003).

## BAB III METODE PENELITIAN

Pada bab metode penelitian ini akan dibahas serta langkah-langkah yang dilakukan dalam penelitian pengenalan iris dengan transformasi *haar wavelet* untuk ekstraksi ciri dan *hamming distance* untuk pengenalan iris

### 3.1 Langkah-Langkah Penelitian

Langkah-langkah yang harus dilakukan dalam penelitian yakni studi literatur, pengumpulan data lapangan, implementasi perangkat lunak, dan pengujian perangkat lunak sebagaimana yang ditunjukkan pada gambar 3.1.



**Gambar 3.1** Flowchart langkah-langkah penelitian

Berikut penjelasan dari masing-masing langkah penelitian :



### 3.1.1 Studi Literatur

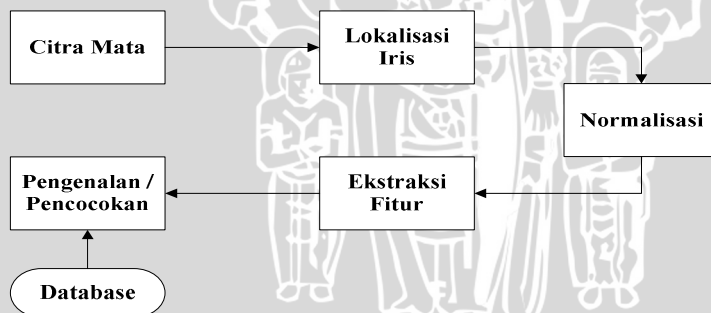
Studi literatur yang telah ditelaah adalah jurnal-jurnal penelitian tentang pengenalan iris mata dengan menggunakan berbagai macam metode. Selain itu juga ditelaah buku-buku yang berhubungan dengan pengolahan citra digital.

### 3.1.2 Pengumpulan Data Lapangan

Metode ini digunakan untuk mengumpulkan data-data yang dibutuhkan dalam penelitian. Data-data yang dimaksud ialah data citra mata yang berasal dari MMU (Multimedia University) yang di dalamnya terdapat 450 iris yang diambil dari 46 orang yang berbeda dengan jumlah mata kanan dan kiri sama yakni masing-masing 225. Data yang berasal dari MMU (Multimedia University) seluruhnya berwarna *grayscale*.

### 3.1.3 Implementasi Perangkat Lunak

Implementasi perangkat lunak ini terdiri dari beberapa tahap yaitu lokalisasi iris dari citra mata, normalisasi citra iris mata, ekstraksi fitur dan pencocokan iris mata dengan data latih yang ada dalam *database*.

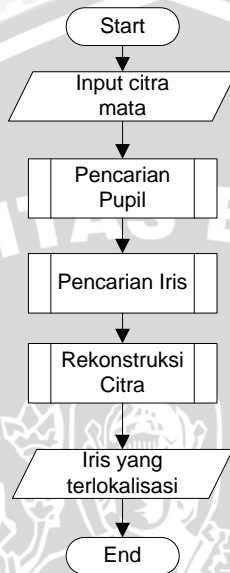


**Gambar 3.2** Proses Pengenalan Iris mata

#### 3.1.3.1 Lokalisasi Iris

Ada 3 proses yang harus dilakukan pada lokalisasi Iris yaitu pencarian pupil, pencarian iris dan rekonstruksi citra iris.



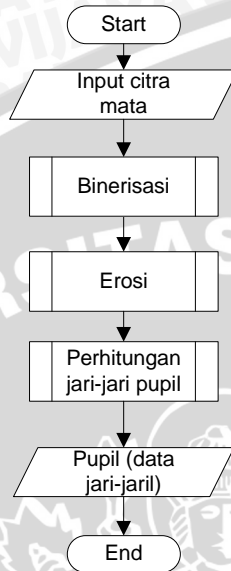


**Gambar 3.3** *Flowchart* Tahapan Lokalisasi Iris

Berikut adalah penjelasan masing-masing tahap:

### 3.1.3.1.1 Pencarian Pupil

Pencarian pupil ini sangat diperlukan untuk mengetahui titik pusat dan jari-jari pupil. Karena data titik pusat dan jari-jari pupil penting diperlukan untuk proses lokalisasi, normalisasi, dan pencocokan iris mata. Dalam pencarian pupil ada 3 proses yang dilakukan yakni binerisasi, erosi dan perhitungan jari-jari pupil.

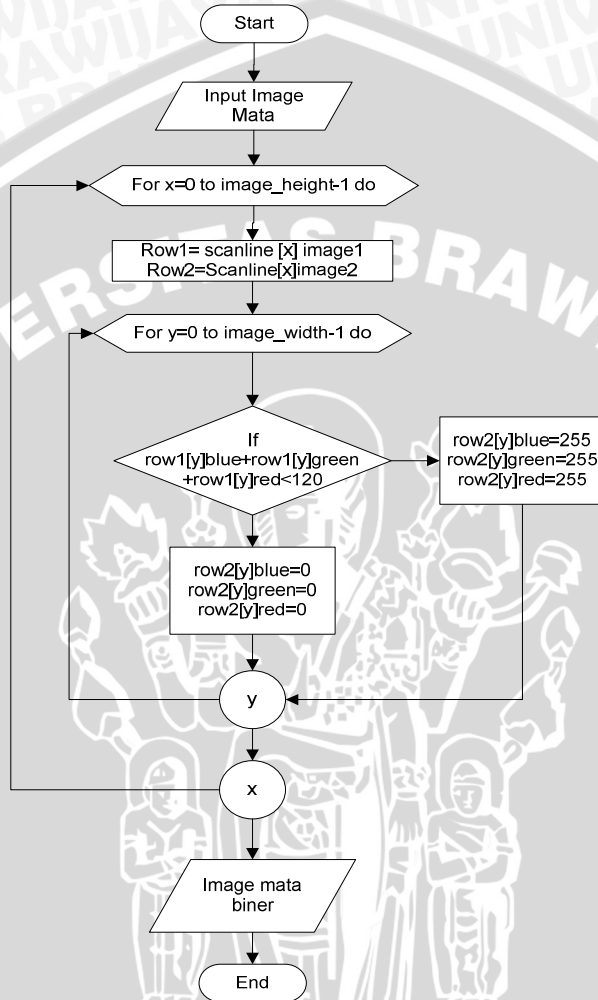


**Gambar 3.4** *Flowchart* Tahapan Pencarian Pupil

Penjelasan dari setiap tahapan pencarian pupil adalah sebagai berikut:

1. Binerisasi

Proses ini bisa disebut juga tahap segmentasi yang memisahkan iris yang diinginkan dengan bagian lain mata yang tidak dibutuhkan secara global. Pada tahap ini sistem akan mengubah citra mata dari citra warna menjadi citra *grayscale* lalu menjadi citra biner dengan *threshold* nilai derajat keabuan yang rendah yaitu antara 40-45. Hasil dari proses ini adalah bentuk pupil mata yang berwarna hitam (0) selainnya berwarna putih (255).

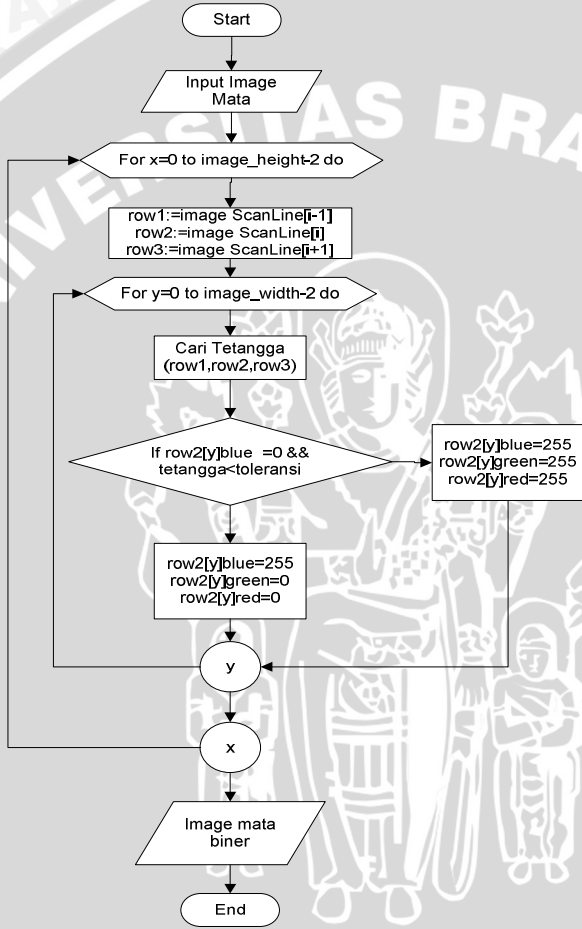


**Gambar 3.5** Flowchart Proses Binerisasi

## 2.Erosi

Pada saat pencarian pupil, *noise* yang sangat mengganggu adalah bagian bulu mata karena akan berpengaruh pada perhitungan jari-jari dan titik pusat. Setelah proses binerisasi dilakukan, dilanjutkan dengan proses

menghilangkan *noise* dilakukan dengan metode pencarian tetangga (*nearest neighbor*). Dengan asumsi apabila memiliki tetangga ukuran dari *n*. maka dihilangkan dan dianggap sebagai *noise*.



**Gambar 3.6** Flowchart Proses Erosi pada Pencarian Pupil

### 3.Perhitungan jari-jari pupil

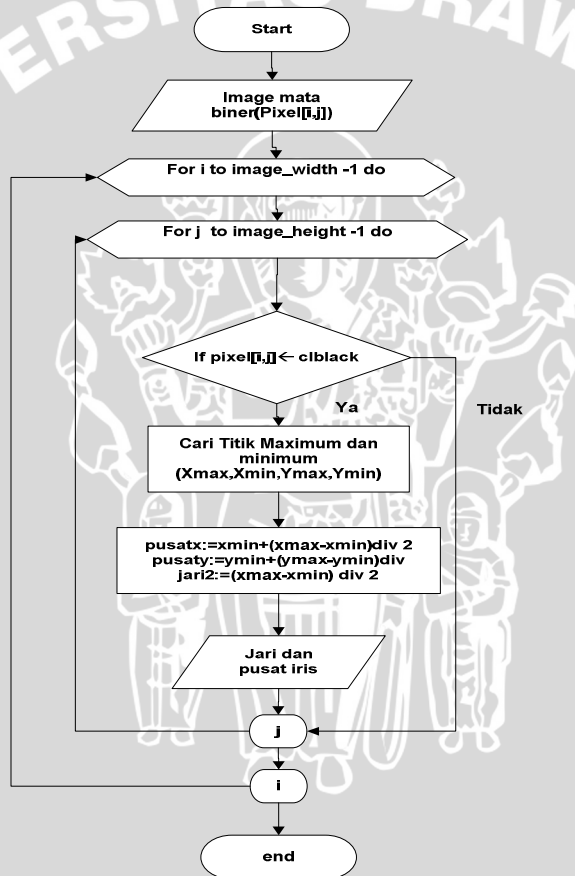
Pada tahap ini sistem melakukan *scan* terhadap gambar, sampai menemukan titik hitam terakhir dan titik

hitam pertama. Dari titik maksimal hitam dan titik minimal hitam bisa didapat titik pusat dengan rumus pada persamaan 3.1, 3.2 dan 3.3

$$\text{Pusat}_x = x_{\min} + (x_{\max} - x_{\min}) / 2 \quad (3.1)$$

$$\text{Pusat}_y = y_{\min} + (y_{\max} - y_{\min}) / 2 \quad (3.2)$$

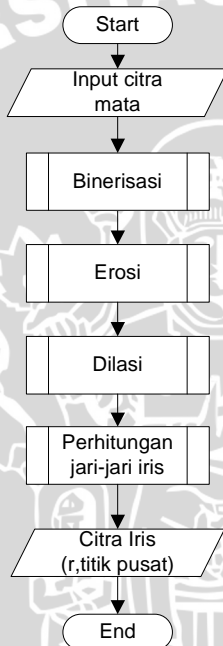
$$r = (x_{\max} - x_{\min}) / 2 \quad (3.3)$$



**Gambar 3.7** Flowchart Proses Perhitungan Jari-Jari

### 3.1.3.1.2 Pencarian iris

Setelah mengetahui jari-jari pupil dan titik pusat ( $x_p, y_p$ ), maka langkah selanjutnya adalah dengan pencarian iris. Tahap yang dilakukan hampir sama dengan saat pencarian pupil yakni binerisasi, erosi, dan perhitungan jari-jari. Perbedaannya dalam proses pencarian iris ditambah dengan proses dilasi agar data jari-jari iris valid.



**Gambar 3.8** Flowchart Tahapan Pencarian Iris

Penjelasan dari masing-masing tahap sebagai berikut:

#### 1. Binerisasi

Proses ini bisa disebut juga tahap segmentasi yang memisahkan iris yang diinginkan dengan bagian lain mata yang tidak dibutuhkan secara global. Pada tahap ini sistem akan mengubah citra mata dari citra warna menjadi citra grayscale lalu menjadi citra biner dengan *threshold* nilai derajat keabuan yang rendah yaitu antara 80-100. Hasil dari proses ini

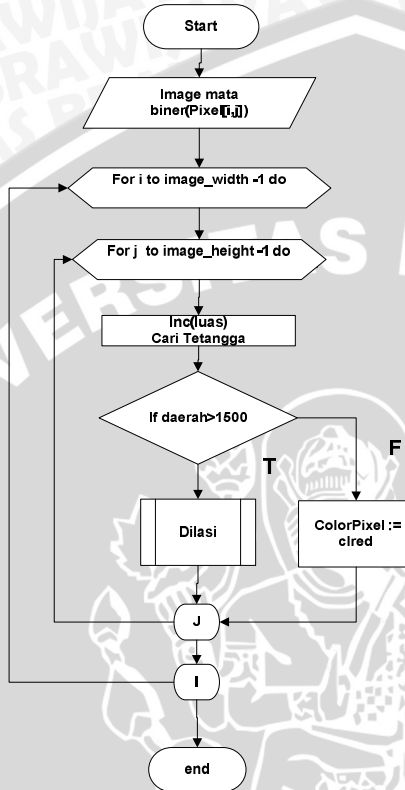


adalah bentuk pupil mata yang berwarna hitam (pixel 0) dan selainnya berwarna putih (pixel 1).

## 2. Erosi dan Dilasi

Pada saat pencarian iris, *noise* yang ada jauh lebih besar dan tidak bisa diatasi dengan pencarian tetangga. Maka untuk menghilangkannya dengan region *growing* sederhana. Dengan menghilangkan daerah yang lebih kecil. Setelah proses itu dilakukan, menentukan pixel yang bertetangga dari suatu seed serta menentukan apakah suatu pixel ditambahkan ke dalam *seed* tersebut atau tidak. Ini dilakukan untuk menyempurnakan bagian iris yang berlubang karena dianggap *noise*.





**Gambar 3.9** Flowchart Proses Erosi Dilasi

### 3. Perhitungan jari-jari iris

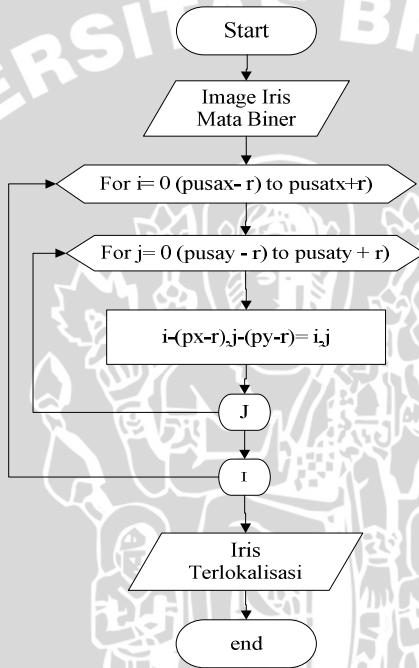
Proses perhitungan jari-jari iris sama dengan perhitungan jari-jari pada pupil.

#### 3.1.3.1.3 Rekonstruksi Citra

Dari proses segmentasi sebelumnya, akan diperoleh bagian iris sebagai bentuk bulatan hitam dengan latar putih. Langkah selanjutnya adalah untuk mengambil hanya bagian iris dengan melakukan rekonstruksi ulang citra mata asli. Untuk batasan yang digunakan dalam rekonstruksi look up, dapat diperoleh dengan pencarian titik dari pusat ke tepi iris

terluar hingga mencapai titik putih yang pertama (sebagai batas iris) pada citra biner.

Pencarian titik batas dimaksudkan untuk mengukur besarnya iris mata pada citra. Dengan titik batas ini, sistem akan mengetahui sampai pada batas mana citra asli harus dipotong untuk mendapatkan hanya bagian iris yang tersisa.



**Gambar 3.10** Flowchart Rekonstruksi Look-Up

### 3.1.3.2 Normalisasi

Sistem ini akan mengubah citra iris lingkaran menjadi persegi. Prosesnya diawali dengan Perhitungan beberapa parameter lingkaran seperti titik pusat dan jari-jari lingkaran iris dan pupil, dapat dilakukan dengan terlebih dahulu mengambil titik-titik sampel pada tepian yang telah diekstrak. Pengambilan titik-titik ini dilakukan mulai dari titik pusat sampai ke batas luar.

Proses pengambilan titik ini membutuhkan perkiraan titik tengah yang telah didapat saat proses lokalisasi iris. Titik pusat yang telah dicari di awal bisa digunakan sebagai informasi yang membantu proses normalisasi.

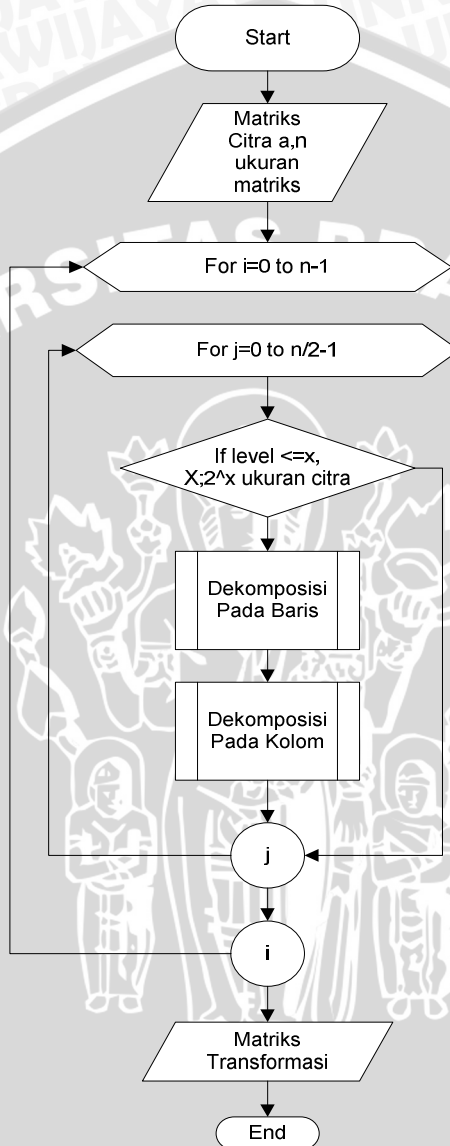
Titik acuan tersebut adalah titik pusat, jari-jari iris dan jari-jari pupil. Setelah mendapatkan titiknya dilakukan pembacaan piksel-piksel citra secara *clock wise* sejauh  $360^{\circ}$  yang dilakukan pada bidang yang dibatasi antara jari-jari pupil dan jari-jari iris.



**Gambar 3.11** Ilustrasi Pembacaan Data

### 3.1.3.3 Ekstraksi Ciri dengan Transformasi *Haar Wavelet*

Pada tahap ini, sistem akan mentransformasikan citra ke daerah frekuensi dengan inputan nilai *pixel* pada citra. Citra digital merupakan *input* matrik 2 dimensi. Matrik ini akan ditransformasikan menggunakan filter *lowpass* dan *highpass* seperti pada persamaan 2.14 dan 2.15 pada tinjauan pustaka. Proses transformasi akan dilakukan hingga kedalaman level ke  $n$ , dimana  $N=2^n$  dengan  $N$  adalah ukuran citra. Pada tiap level akan dilakukan transformasi baris terlebih dahulu melalui filter *lowpass* dan *highpass*. Baru kemudian dilakukan transformasi terhadap kolom. Setelah proses transformasi selesai maka akan terbentuk matrik transformasi domain frekuensi dengan energi yang terkumpul di pojok kiri atas.



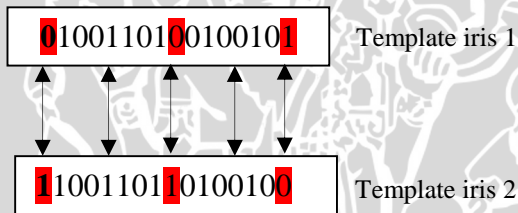
**Gambar 3.12** Flowchart Tranformasi Haar Wavelet pada Citra

### 3.1.3.4 Pencocokan dengan *Hamming Distance*

Pada proses pencocokan menggunakan algoritma *hamming distance*, citra iris mata yang telah diekstrak dengan transformasi *haar wavelet* akan dibandingkan citra yang tersimpan dalam database. Formula *hamming distance* secara lengkap sebagai berikut

$$= - \sum ( \quad ) \quad (3.4)$$

$X_j$  dan  $Y_j$  adalah vector fitur yang merupakan hasil dari citra terekstrak. Citra yang pertama merupakan citra masukan yang telah diproses. Sedangkan citra kedua merupakan citra yang telah disimpan dalam database.  $N$  merupakan angka yang merepresentasikan setiap gambar. Contoh Proses perhitungannya sebagai berikut



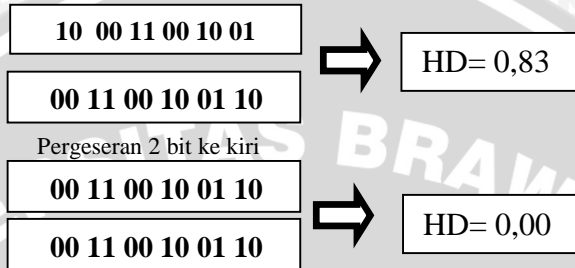
**Gambar 3.13** Contoh Template Iris

Pada contoh di atas iris template1 dibandingkan dengan iris template2 , maka

$$= \frac{\text{Number of mismatches}}{\text{Total length}} = \frac{5}{32} = 0,19$$

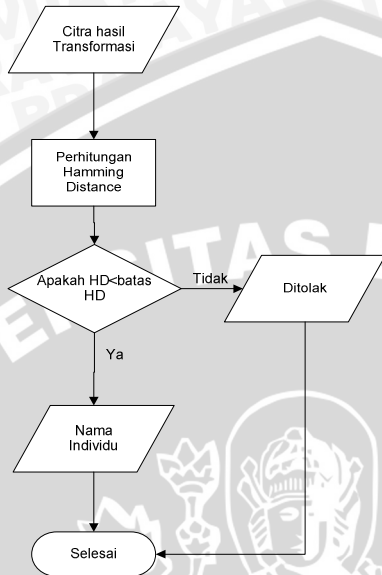
Setelah dilakukan proses perhitungan, 2 iris yang berasal dari gambar yang sama akan memiliki nilai *hamming distance* = 0 tetapi ini jarang terjadi . Untuk mengatasi hal itu Daughman mencetuskan adanya proses penggeseran 2 bit ke sebelah kiri dan sebelah kanan. Dari perhitungan *hamming distance* diambil yang memiliki nilai terkecil. Ilustrasinya penggeseran sebagai berikut:





**Gambar 3.14** Ilustrasi proses *shifting* (penggeseran) *hamming distance*

Sistem akan mengambil nilai *hamming distance* terkecil dari ketiga perhitungan *hamming distance* tersebut . Jika *hamming distance* = 0 berarti iris tersebut 100% cocok. Pencocokan dilakukan one-to-one yaitu hasil pengenalan dibandingkan dengan iris template yang ada di database dan dikenali sebagai orang yang memiliki nilai *hamming distance* terkecil.



**Gambar 3.15** Flowchart Pencocokan Iris

Dalam implementasi perangkat lunak ini, untuk dapat mengenali individu yang terdapat dalam data latih dibutuhkan data individu yang tersimpan dalam *database*. *Database* yang digunakan terdiri satu tabel saja yakni tabel Data berisi atribut-atribut ID sebagai *primary key*, Nama yang menyimpan nama individu, Ambang untuk menyimpan *threshold* saat pencarian iris, dan Matriks yang menyimpan hasil transformasi *haar wavelet*. Untuk lebih detailnya atribut tabel Data dapat dilihat pada Tabel 3.1.

**Tabel 3.1** Tabel DATA

Field	Type	Panjang	Allow Null
ID	int ( <i>primary key</i> )	4	No
Nama	Nvarchar	50	No
Ambang	Integer	5	Yes
Matriks	Nvarchar	1024	Yes

### 3.5 Pengujian Perangkat Lunak

#### 3.5.1 Pengujian dengan Nilai *Hamming Distance* Terkecil

Pengukuran tingkat keakuratan dilakukan untuk memperoleh ukuran ketepatan sistem dalam mengenali individu yang terdapat dalam *database* dengan data masukan yang akan dikenali atau diujikan. Tingkat kesalahan yang diukur dinyatakan dalam persen (%) dengan perhitungan pada rumus 3.5.

$$error = \frac{\text{jumlah\_kesalahan\_mengenali}}{\text{jumlah\_sampel}} \times 100\% \quad (3.5)$$

Pengukuran tingkat kesalahan ini menggunakan data sampel citra iris yang hendak diujikan atau dikenali. Proses pengukuran tingkat kesalahan dilakukan dengan dua bentuk pengujian kesalahan, yaitu:

1. *False inner rate* (FIR) yang menunjukkan kesalahan dalam mengenali citra individu dalam data latih. Sebuah gambar masukan yang berasal dari dalam data latih yang seharusnya dapat dikenali dengan benar berubah menjadi salah dikenali.
2. *False outer rate* (FOR) yang menunjukkan kesalahan dalam mengenali identitas gambar masukan data uji yang berasal dari individu yang terdapat dalam data latih (*intra*class).

Besarnya nilai FIR dan FOR didapat dengan cara mengubah jumlah individu yang dilatih dalam data latih dengan penambahan 5 individu untuk setiap pengujiannya.

**Tabel 3.2** Rancangan Tabel Pengujian Citra Mata

No	Input	Dikenali	<i>Hamming Distance</i>	Hasil
			Error	$\frac{\text{Jumlah\_kesalahan}}{\text{jumlahsampel}} \times 100\%$

### 3.5.2 Pengujian dengan Ambang *hamming distance*

Ambang *hamming distance* didapatkan dari rata-rata *hamming distance* hasil pengujian, nilai minimum, dan maksimum ambang *hamming distance*. Penentuan ambang ini didasarkan pada percobaan John Daughman.

Dilanjutkan dengan mencari tingkat kesalahan penerimaan pada data uji yang berasal dari 75 data uji individu asing (*interclass*) dan pengujian pada data uji *intra*class.

**Tabel 3.3** Rancangan Tabel Pengujian Citra Mata 2

No	Input	<i>Hamming Distance</i>	Ambang	Dikenali /Tidak

Selanjutnya dihitung tingkat kesalahannya dengan False Accepted Rate dan False Rejected Rate. FAR (*false acceptance rate*) menyatakan tingkat kesalahan penerimaan yang muncul akibat dari sistem menganggap sah pengguna yang tidak sah, FRR (*false rejection rate*) menyatakan tingkat kesalahan penolakan yang muncul akibat dari sistem menganggap tidak sah pengguna yang sah, sedangkan EER (*equal error rate*) menyatakan tingkat kesalahan ketika FAR = FRR.

$$(FIR) = \frac{\text{Jumlah data yang salah diterima}}{\text{Banyaknya data}} \times 100\%$$

$$(FRR) = \frac{\text{Jumlah data yang salah ditolak}}{\text{Banyaknya data}} \times 100\%$$

$$\text{Equal Error Rate (ERR)} = |FRR - FAR|$$

## 3.2 Contoh Perhitungan Manual

### 3.2.1 Perhitungan Transformasi Haar Wavelet

Perhitungan transformasi *haar wavelet* dilakukan  $n$  level sesuai dengan iris mata yang diekstraksi. Berikut contoh perhitungan pada matriks  $8 \times 8$  dari citra iris yang ternormalisasi. Transformasi dilakukan hingga level 3 karena  $8 = 2^n$ ,  $n = 3$ .

Misalnya Data digital iris sebagai berikut:

68	75	78	51	55	84	41	62
88	44	43	79	82	100	70	69
98	92	76	42	100	44	67	62
82	84	100	99	94	64	43	88
40	67	87	68	77	54	45	93
41	44	70	84	56	87	52	86
100	56	53	74	55	97	53	46
92	91	43	54	84	42	53	69

**Gambar 3.16** Matriks  $8 \times 8$

Pada matrik data iris dilakukan dekomposisi baris untuk mendapatkan nilai *average/approximation coefficients* (A) seperti yang ditunjukkan pada Persamaan 2.2. Pada level 1 dihasilkan nilai A seperti pada matriks berikut:

101,12	91,22	98,29	72,83	-4,95	19,09	-2,83	-20,51
93,34	86,27	128,69	98,29	31,11	-25,46	-2,12	-12,73
134,35	83,44	101,82	91,22	4,24	24,04	-41,01	39,60
117,38	140,71	111,72	92,63	-1,41	0,71	3,54	21,21
75,66	109,60	92,63	97,58	-19,9	13,44	-6,36	16,26
60,10	108,89	101,12	97,58	-2,12	-9,90	19,80	-21,92
110,31	89,80	107,48	70,00	31,11	-14,85	13,44	-29,70
129,40	68,59	89,10	86,27	0,71	-7,78	-21,21	29,70

**Gambar 3.17** Matriks Hasil Dekomposisi Baris Level 1

Pada level 2 dihasilkan matriks sebagai berikut :

136	121	10	-16,5	7	18	55	-17
127	160,5	4	-10,5	5	21,5	47,5	40
154	136,5	20	-1	36	7,5	61,5	-14
182,5	144,5	-0,5	17,5	-16,5	13,5	66,5	-1,5
131	134,5	-4	7	-24	-3,5	82,5	-23
119,5	140,5	-8,5	-1,5	-34,5	2,5	70,5	5,5
141,5	125,5	11,5	-11,5	14,5	26,5	27,5	32,5
140	124	-5	6	43	2	60,5	6

**Gambar 3.18** Matriks Hasil Dekomposisi Baris Level 2

Pada level 3 dihasilkan matriks sebagai berikut :



181,73	-4,60	17,68	26,87	10,61	18,74	-16,62	-7,78
203,29	-4,60	18,74	61,87	-23,69	10,25	-10,96	-11,67
205,41	13,44	30,76	33,59	12,37	14,85	-26,16	20,15
231,22	12,02	-2,12	45,96	26,87	-12,73	24,04	-21,21
187,74	2,12	-19,45	42,07	-2,47	-7,78	21,92	-14,50
183,85	-7,07	-22,63	53,74	-14,85	-4,95	23,33	-26,16
188,80	0,00	28,99	42,43	11,31	16,26	-18,38	-8,49
186,68	0,71	31,82	47,02	11,31	-7,78	-26,16	28,99

**Gambar 3.19** Matriks Hasil Dekomposisi Baris Level 3

Selanjutkan untuk mendapatkan nilai *wavelet/detail coefficients* (C) ditunjukkan pada persamaan 2.3 dilakukan dekomposisi pada kolom. Pada level 1 hasilnya sebagai berikut :

272,25	-6,51	25,75	62,75	-9,25	20,50	-19,50	-13,75
308,74	18,00	20,25	56,25	27,75	1,50	-1,50	-0,75
262,75	-3,50	-29,76	67,75	-12,25	-9,00	32,00	-28,75
265,50	0,50	43,00	63,25	15,99	6,00	-31,49	14,50
-15,25	0,00	-0,75	-24,75	24,25	6,00	-4,00	2,75
-18,25	1,00	23,25	-8,75	-10,25	19,50	-35,50	29,25
2,75	6,50	2,25	-8,25	8,75	-2,00	-1,00	8,24
1,50	-0,50	-2,00	-3,25	0,00	17,00	5,50	-26,50

**Gambar 3.20** Matriks Hasil Dekomposisi Kolom Level 1

Pada level 2 dihasilkan matriks sebagai berikut :

410,83	8,13	32,53	84,15	13,08	15,56	-14,85	-10,26
373,54	-2,12	9,37	92,63	2,65	-2,13	0,36	-10,08
-23,69	0,71	15,91	-23,69	9,90	18,04	-27,93	22,63
3,01	4,24	0,17	-8,13	6,19	10,61	3,19	-12,91
-25,81	-17,33	3,89	4,59	-26,16	13,44	-12,73	-9,20
-1,94	-2,83	-51,45	3,18	-19,97	-10,61	44,90	-30,58
2,13	-0,71	-16,97	-11,32	24,40	-9,55	22,27	-18,74
0,89	4,95	3,01	-3,54	6,19	-13,44	-4,60	24,57

**Gambar 3.21** Matriks Hasil Dekomposisi Kolom Level 2

Pada level 3 dihasilkan matriks sebagai berikut :

554,63	4,25	29,62	125,00	11,12	9,50	-10,25	-14,38
-14,62	3,50	11,37	-22,50	11,38	20,25	-17,50	6,87
-19,62	-14,26	-33,63	5,50	-32,62	2,00	22,74	-28,13
2,13	3,00	-9,87	-10,50	21,63	-16,25	12,50	4,13
26,37	7,25	16,38	-6,00	7,38	12,50	-10,75	-0,12
-18,87	-2,50	11,13	-11,00	2,62	5,25	-22,00	25,13
-16,87	-10,25	39,13	1,00	-4,38	17,00	-40,75	15,12
0,88	-4,00	-14,12	-5,50	12,88	2,75	19,00	-30,62

**Gambar 3.22** Matriks Hasil Dekomposisi Kolom Level 3

Dengan ketentuan jika nilai *wavelet/detail coefficients* ( $C$ )  $\geq 0$  maka  $C = 1$  dan jika nilai *wavelet/detail coefficients* ( $C$ )  $< 0$  maka  $C = 0$ , hasil akhir matriks transformasi wavelet sebagai berikut:

1	1	1	1	1	1	0	0
0	1	1	0	1	1	0	1
0	0	0	1	0	1	1	0
1	1	0	0	1	0	1	0
1	1	1	0	1	1	0	0
0	0	1	0	1	1	0	1
0	0	1	1	0	1	0	1
1	0	0	0	1	1	1	0

**Tabel 3.23** Matriks Hasil Akhir Transformasi *Haar Wavelet*

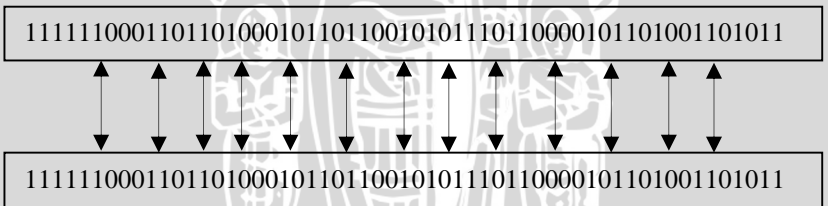
### 3.2.2. Contoh Perhitungan nilai *hamming distance*

Matriks hasil akhir transformasi *haar wavelet* pada tabel 3.9 dijadikan linier hasilnya sebagai berikut:

111110001101101000101101100101011101100001011010011010110001110

Kemudian dilakukan perhitungan nilai *hamming distance* dengan data matriks yang ada di *database*.

1. Terhadap data latih

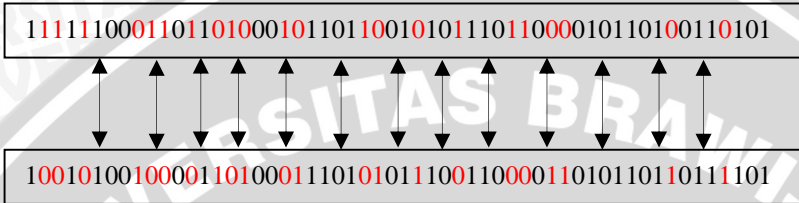


**Gambar 3.24** Pencocokan Dengan Data Latih

Pada perbandingan *template* iris di atas tidak ada perbedaan antara *template* yang pertama dan kedua. Maka bisa dihitung nilai *hamming distance* (HD) sebagai berikut:

$$= \frac{\text{---}}{\text{---}} = \text{---} = 0,00$$

2. Terhadap data uji



**Gambar 3.25** Pencocokan Dengan Data Uji

Warna merah mempresentasikan bit template x yang bukan anggota y. Maka bisa dihitung nilai *hamming distance* (HD) sebagai berikut:

$$= \frac{\text{---}}{\text{---}} = \text{---} = 0,42$$

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

Pada bab ini akan dibahas mengenai implementasi dari representasi rancangan ke bahasa pemrograman yang dimengerti oleh komputer. Bab ini akan membahas hasil implementasi yang dihasilkan oleh perangkat lunak

#### **4.1 Lingkungan Implementasi**

Perangkat keras yang digunakan dalam pembuatan aplikasi penjadwalan sumber daya proyek adalah sebagai berikut :

1. Prosesor Intel Pentium 4 3Ghz
2. Memori 1 GB
3. *Harddisk* dengan kapasitas 150 GB
4. Monitor 14''
5. *Keyboard*
6. *Mouse*

Perangkat lunak yang digunakan dalam pembuatan aplikasi penjadwalan sumber daya proyek adalah sebagai berikut :

1. Sistem Operasi Windows XP
2. Borland Delphi 7
3. EMS *Interbase*

#### **4.2 Implementasi Basis Data**

Pada bab III telah dijelaskan rancangan basis data yang berisi tabel Data untuk menyimpan identitas masing-masing individu, matriks data dan ambang/ *threshol* saat melakukan pencarian iris. Tabel tersebut diimplementasikan ke dalam *Interbase*.

Data	
PK	<u>NO_ID</u>
	Nama Matriks Ambang

**Gambar 4.1** Struktur Basis Data

### 4.3 Implementasi Program

Berdasarkan perancangan perangkat lunak pada subbab 3.4 maka pada subbab ini akan dibahas mengenai implementasi dari perancangan tersebut. Tahap pertama yang harus dilakukan dalam proses pengenalan iris adalah lokalisasi/segmentasi iris dari bagian mata yang yang lain seperti sclera, kelopak mata, dan bulu mata. Karena bagian yang lain tersebut akan mengganggu proses pengenalan iris.

#### 4.3.1 Lokalisasi

##### 4.3.1.1 Pencarian Pupil

Hal pertama yang dilakukan adalah melakukan pendeteksian terhadap pupil untuk mendapatkan titik pusat dan jari-jari iris. Pendeteksian iris ini sangat diperlukan untuk memisahkan bagian pupil dengan bagian iris dan menghilangkan bagian lain. Proses pertama yang harus dilakukan adalah proses binerisasi. Piksel citra yang memiliki nilai kurang dari ambang (*threshol*d) diberi warna hitam (0) dan dideteksi sebagai pupil.



```

procedure TFormUtama.BtPupilClick(Sender:
TObject);
var
  Ww,hh,x,y:integer;
  row1:PRGBTripleArray;
  row2:PRGBTripleArray;
begin
PageControll1.TabIndex:= 0;
Ww := ImageAsli.Picture.Bitmap.Width;
Hh := ImageAsli.Picture.Bitmap.Height;
for x:=0 to hh-1 do
begin
ROW1:=ImageAsli.Picture.Bitmap.ScanLine[x];
Row2:=ImagePupil.Picture.Bitmap.ScanLine[x];
  for y:=0 to ww-1 DO
    begin
      If (Row1[y].rgbtBlue+Row1[y].rgbtGreen+
Row1[y].rgbtRed) < 120 then
        begin
          Row2[Y].rgbtBlue:=0;
          Row2[Y].rgbtGreen:= 0;
          Row2[y].rgbtRed:= 0;
        end
      else
        begin
          Row2[Y].rgbtBlue:=255;
          Row2[Y].rgbtGreen:= 255;
          Row2[y].rgbtRed:= 255;
        end;
    End;
  End;
End;

```

**Sourcecode 4.1 Sourcecode Proses Binerisasi citra untuk menemukan pupil**

Setelah ditemukan bagian pupil , maka tahap selanjutnya adalah dengan mencari titik pusat x dan y dan jari-jari iris Pencarian titik pusat berguna untuk proses selanjutnya dalam proses lokalisasi iris.Sedangkan pencarian jari-jari berguna saat proses normalisasi.

```

procedure CariPusatdanJari2 (var gambar:Timage;var
pusatx,pusaty,jari2:integer);
var
  w,h,i, j: integer;
begin
  xmax:= 0;
  xmin:= 1000;
  ymax:= 0;
  ymin:= 1000;
  w:= gambar.Picture.Bitmap.Width;
  h:= gambar.Picture.bitmap.Height;
  for i:= 0 to w-1 do
    for j:= 0 to h-1 do
      begin
        if (gambar.Canvas.Pixels[i,j]= clblack) then
          begin
            if I >Xmax then
              xMax := i ;
            if i< XMIN then
              xMin := i;
            if j >Ymax then
              yMax := j ;
            if j< yMIN then
              yMin := j;
            end
          end;
        pusatx:=xmin+(xmax-xmin)div 2;
        pusaty:=ymin+(ymax-ymin)div 2;
        jari2:=(xmax-xmin) div 2;
      end;
    end;
  end;
end;

```

**Sourcecode 4.2 Sourcecode** procedure cari jari-jari dan titik pusat pupil

Saat proses pencarian pupil, untuk citra gambar yang memiliki bulu mata sangat tebal dan hitam akan menemui hambatan yakni sebagian bulu mata akan dianggap sebagai bagian dari pupil karena nilai pikselnya kurang dari ambang. Oleh karena itu, perlu dibuat 1 prosedur lagi yang berfungsi untuk membuang *noise* berupa bulu mata. Hal ini dilakukan dengan mencari tetangga. Apabila tetangga kurang dari toleransi yang ditetapkan yakni 8 maka diberi warna merah dan tidak dianggap sebagai bagian dari pupil.

```

procedure BuangNoise(var
gambar:timame;toleransi:integer);
var w,h,i,j,tetangga:integer;
gambarBuffer:array[0..1000,0..1000]of boolean;
Row,Row1,row2,row3:PRGBTripleArray;
begin
w:= gambar.Picture.Bitmap.Width;
h:= gambar.Picture.bitmap.Height;

for i:= 0 to h-1 do
for j:= 0 to w-1 do
gambarBuffer[i,j]:=false;

for i:= 1 to h-2 do
begin
row1:=gambar.Picture.Bitmap.ScanLine[i-1];
row2:=gambar.Picture.Bitmap.ScanLine[i];
row3:=gambar.Picture.Bitmap.ScanLine[i+1];
for j:= 1 to w-2 do
begin
tetangga:=0;
if row1[j-1].rgbtBlue = 0 then inc(tetangga);
if row1[j].rgbtBlue = 0 then inc(tetangga);
if row1[j+1].rgbtBlue = 0 then inc(tetangga);

if row2[j-1].rgbtBlue = 0 then inc(tetangga);
if row2[j+1].rgbtBlue = 0 then inc(tetangga);
if row3[j-1].rgbtBlue = 0 then inc(tetangga);
if row3[j].rgbtBlue = 0 then inc(tetangga);
if row3[j+1].rgbtBlue = 0 then inc(tetangga)
if ((row2[j].rgbtBlue=0)and(Tetangga>=toleransi))
then
gambarbuffer[j,i]:=false
else if row2[j].rgbtBlue=0 then
gambarbuffer[j,i]:=true;
end;
end;//end for
for i:= 0 to h-1 do
begin
row:=gambar.Picture.Bitmap.ScanLine[i];
for j:= 0 to w-1 do
.....

```

**Sourcecode 4.3** Sourcecode procedure untuk membuang noise kecil dengan pencarian tetangga

### 4.3.1.2 Pencarian Iris

Pencarian iris dilakukan hampir sama dengan pencarian pupil yaitu dengan melakukan binerisasi. Perbedaan antara pencarian iris dan pupil adalah pada nilai *threshold* (ambang). Ini terjadi karena warna iris yang cenderung abu-abu bukan hitam. Pada pencarian iris juga dilakukan prosedur membuang *noise* dan menyempurnakan bagian iris yang lubang.

```
procedure TFormUtama.BtBinerisasiClick(Sender:
TObject);
var
Ww,hh,i, j,t:integer;
row1:PRGBTripleArray;
Row2:PRGBTripleArray;
begin
t:=SpinTreshold.value ;
Ww := ImageAsli.Picture.Bitmap.Width;
Hh := ImageAsli.Picture.Bitmap.Height;
for i:=0 to hh-1 do
begin
Row1:=ImageAsli.Picture.Bitmap.ScanLine[i];
Row2:=ImageBinerisasi.Picture.Bitmap.ScanLine[i];
for j:=0 to ww-1 DO
begin
if
(Row1[j].rgbtBlue+Row1[j].rgbtGreen+Row1[j].rgbtRed) < t*3 then
BEGIN
Row2[j].rgbtBlue:= 0;
Row2[j].rgbtGreen:=0;
Row2[j].rgbtRed:=0;
End
.....

```

**Sourcecode 4.4** Sourcecode procedure untuk mencari bagian Iris dengan binerisasi

Sebagaimana dalam pencarian pupil, dalam proses mencari iris juga akan ditemui *noise*. Dalam pencarian iris, *noise* yang ditemukan lebih besar dari pencarian pupil yakni berupa bulu mata dan sebagian sklera. Ini terjadi karena warna

sklera dan iris yang hampir sama. Oleh karena itu dibutuhkan prosedur untuk membuang *noise*.

```
procedure TFormUtama.BtNoiseClick(Sender: TObject);
var i,j:integer;
//temp:tkordinatimage;
begin
  jumArrayKI:=0;
  for i:=0 to imagebinerisasi.Picture.Width do
    for j:=0 to imagebinerisasi.Picture.Height do
      begin
        if imagebinerisasi.Canvas.Pixels[i,j]=clblack then
          begin
            inc(jumArrayKI);
            arrayKordinatImage[jumArrayKI].luas:=0;
            countRecursive(imagebinerisasi,clblack,i,j,0,
              arrayKordinatImage[jumArrayKI]);
            if arrayKordinatImage[jumArrayKI].luas<1500
              then dec(jumArrayKI);
          end;
        end;
      end;

  for i:=0 to imagebinerisasi.Picture.bitmap.Width do
    for j:=0 to imagebinerisasi.Picture.bitmap.Height do
      imagebinerisasi.Canvas.Pixels[i,j]:=clwhite;
    for i:=1 to jumArrayKI do
      for j:=1 to arrayKordinatImage[i].luas do
        imagebinerisasi.Canvas.Pixels[arrayKordinatImage[i].
          posisi[j].x,arrayKordinatImage[i].posisi[j].y]:=clbl
          ack;
      end;
    end;

  bailout(imagebinerisasi,SpinEditNoise.Value);
  CariPusatdanJari2(ImageBinerisasi,px,py,rIris);
end;
```

#### **Sourcecode 4.5 Sourcecode** prosedur erosi dan dilasi

Prosedur membuang *noise* (erosi) di atas membutuhkan prosedur lain yang dipanggil yakni prosedur mencari daerah iris dengan membagi gambar bebrapa sub daerah berdasarkan kesamaan warna, implementasinya sebagai berikut:

```

procedure countRecursive(var image13: TImage; const
curcolor:TColor;const X,Y,Tolerance : integer; var
area : tkordinatimage);
begin
  if (X+1 >=image13.Picture.Bitmap.Width) or
  (Y+1 >= image13.Picture.Bitmap.Height)or
  (X-1<0)or
  (Y-1<0) then exit
  image13.Canvas.Pixels[X,Y]:=clBlue;
  inc(area.luas);
  area.posisi[area.luas].x:=x;
  area.posisi[area.luas].y:=y;
  if (image13.Canvas.Pixels[X+1,Y]=curColor) then
countRecursive(image13,curColor,X+1,Y,Tolerance,ar
ea);
  if (image13.Canvas.Pixels[X-1,Y]=curColor) then
countRecursive(image13,curColor,X1,Y,Tolerance,area);
  if (image13.Canvas.Pixels[X,Y-1]=curColor) then
countRecursive(image13,curColor,X,Y1,Tolerance,area);
  if (image13.Canvas.Pixels[X,Y+1]=curColor) then
countRecursive(image13,curColor,X,Y+1,Tolerance,area);
  if (image13.Canvas.Pixels[X+1,Y+1]=curColor) then
countRecursive(image13,curColor,X+1,Y+1,Tolerance,area)
  if (image13.Canvas.Pixels[X-1,Y+1]=curColor) then
countRecursive(image13,curColor,X1,Y+1,Tolerance,area);
  .....

```

**Sourcecode 4.6** Sourcecode procedur untuk mencari daerah yang sama

Selanjutnya dilakukan proses dilasi untuk menyempurnakan bagian iris. Implementasinya sebagai berikut:



```

procedure BailOut(var gambar:timage;toleransi:integer);
var w,h,i,j,tetangga:integer;
    gambarBuffer:array[0..1000,0..1000]of boolean;
    Row1,row2,row3:PRGBTripleArray;
begin
w:= gambar.Picture.Bitmap.Width;
h:= gambar.Picture.bitmap.Height;
for i:= 0 to w-1 do
    for j:= 0 to h-1 do
        gambarBuffer[i,j]:=false;

for i:= 1 to h-2 do
    begin
ROW1:= gambar.Picture.Bitmap.ScanLine[i-1];
ROW2:= gambar.Picture.Bitmap.ScanLine[i];
ROW3:= gambar.Picture.Bitmap.ScanLine[i+1];
        for j:= 1 to w-2 do
            begin
                tetangga:=0;
                if Row1[j-1].rgbtBlue=0 then inc(tetangga);
                if Row1[j].rgbtBlue=0 then inc(tetangga);
                if Row1[j+1].rgbtBlue=0 then inc(tetangga);

                if Row2[j-1].rgbtBlue=0 then inc(tetangga);
                if Row2[j+1].rgbtBlue=0 then inc(tetangga);

                if Row3[j-1].rgbtBlue=0 then inc(tetangga);
                if Row3[j].rgbtBlue=0 then inc(tetangga);
                if Row3[j+1].rgbtBlue=0 then inc(tetangga);

                if (Tetangga>0) and (tetangga<=toleransi) then
                    gambarbuffer[j,i]:=true;
            end;
        end;
    for i:= 1 to h-2 do
        begin
            ROW1:= gambar.Picture.Bitmap.ScanLine[i-1];
            ROW2:= gambar.Picture.Bitmap.ScanLine[i];
            ROW3:= gambar.Picture.Bitmap.ScanLine[i+1];
        End;
    End;
end;

```

**Sourcecode 4.7** Sourcecode procedur membuang noise pada iris dan menyempurnakan iris

```

for j:= 1 to w-2 do
  begin
    if gambarBuffer[j,i] then
      begin
        Row1[j-1].rgbtBlue:= 0 ;
        Row1[j-1].rgbtred:= 0 ;
        Row1[j-1].rgbtgreen:= 0 ;

        Row1[j].rgbtBlue:= 0 ;
        Row1[j].rgbtred:= 0 ;
        Row3[j].rgbtred:= 0 ;

        .....

```

**Sourcecode 4.7** Sourcecode prosedur membuang noise pada iris dan menyempurnakan iris (lanjutan)

#### 4.3.1.3 Rekontruksi Look-Up

Sebagaimana dijelaskan pada bab Metodologi Penelitian subbab 3.4.2 untuk kemudian diambil bagian mata hanya pada bagian iris saja maka dilakukan rekonstruksi look up yakni melakukan lokalisasi iris dengan gambar yang berasal dari citra asli dengan informasi pusat dan jari-jari iris yang telah diperoleh pada proses sebelumnya. Hasil implementasi lokalisasi / segmentasi iris untuk rekonstruksi citra dapat ditunjukkan pada *sourcecode 4.8*

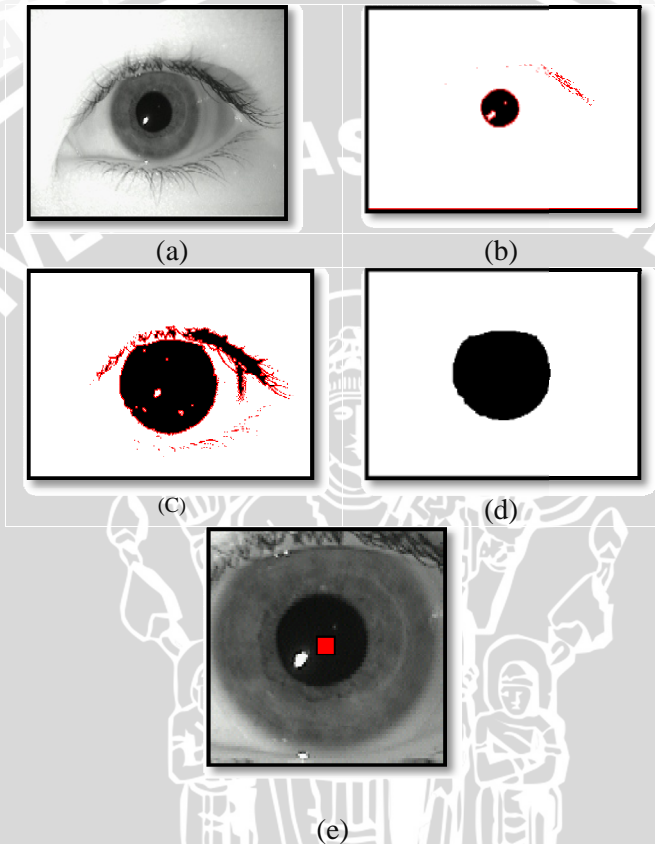
```

procedure TFormUtama.BtLokalisasiClick(Sender:
TObject);
var i,j : integer;
begin
  for i:= px-riris to px+riris do
    for j:= py-riris to py+riris do
      begin
        ImageLokalisasi.canvas.pixels[i-(px-riris),j-(py-
riris)] :=
          px:=px-(px-riris);
          py:=py-(py-riris);
        end;

```

**Sourcecode 4.8** Sourcecode rekontruksi Look-Up

Hasil pengujian lokalisasi/segmentasi citra iri dapat ditunjukkan pada gambar 4.1



**Gambar 4.2** (a) Citra Asli (b) Citra pupil (c) Citra Iris (d) Citra Iris setelah dibuang *noise* (e) Hasil Akhir Lokalisasi

Pada tahap lokalisasi pada sistem ini memiliki kelemahan yakni apabila citra memiliki bulu mata yang tebal dan menutupi bagian iris akan ikut dinormalisasi karena dianggap bagian dari iris yang akan mengganggu hasil dari proses selanjutnya.

### 4.3.2 Normalisasi

Tahap kedua dari proses pengenalan iris adalah melakukan normalisasi yakni mengubah citra iris lingkaran menjadi persegi. Hasil implementasi proses normalisasi adalah sebagai berikut :

```
procedure TFormUtama.BtPolarisasiClick(Sender:
TObject);
var i,j,keliling,radius : integer;
begin

    keliling:= round(2*pi*Riris);
    ImagePolarisasi.Picture.Bitmap.width:=keliling;
    ImagePolarisasi.Picture.Bitmap.height:=1;
    for j:=rPupil to Riris do
    begin
        keliling:=round(2*pi*j);
        ImagePolarisasi.Picture.Bitmap.height:=ImagePolar
        isasi.Picture.Bitmap.height+1;

        for i:=0 to keliling do
            begin
                ImagePolarisasi.Canvas.Pixels[i,j-rpupil]:=
                ImageLokalisasi.Canvas.Pixels[px+round(j*cos
                (i/keliling*360)),py+round(j*sin(i/keliling*
                360))];
            end;
        ImagePolarisasi.Canvas.Pixels[ImagePolarisasi.Pic
        ture.Bitmap.width-1,j-rpupil]:=clred;
    end;
end;
```

#### **Sourcecode 4.7** Sourcecode Proses Normalisasi/polarisasi

Setelah dilakukan proses normalisasi/polarisasi untuk proses selanjutnya yaitu transformasi *haar wavelet* , diharuskannya ukuran citra  $2^n$ . Pada data latih, karena ukuran citra iris yang berbeda-beda (ada yang besar dan kecil) , maka diambil ukuran citra  $2^4$  (16X16).

```

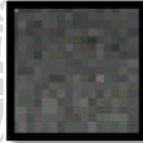
for i:= 0 to 15 do
  for j:= 0 to 15 do
    begin
      ImgCrop.Canvas.pixels[i,j]:=
      ImagePolarisasi.Canvas.Pixels[10+i,j+10];
    End;
  
```

**Source Code 4.8** Source Code Prosedur Proses Crop

Hasil pengujian normalisasi citra iris dapat ditunjukkan pada gambar 4.2 dan hasil *crop* pada gambar 4.3



**Gambar 4.3** Hasil normalisasi citra iris



**Gambar 4.4** Hasil crop citra normalisasi

### 4.3.3 Transformasi *Haar Wavelet*

Tahap selanjutnya yang harus dilakukan adalah dengan melakukan transformasi *haar wavelet*. Proses ini diperlukan karena data yang diolah harus berupa data biner (0 dan 1). Sebelum itu, citra hasil normalisasi diambil datanya. Berikut implementasinya:

```

procedure TFWavelet.inisiasi;
var i,j:integer;
begin
  for i:=0 to 31 do
    for j:=0 to 31 do

      Matriks[i,j]:=getRvalue(formutama.ImgCrop.Canvas.Pixe
      ls[i,j]);
      drawMatriks(stringGrid1,Matriks);
    end;
  
```

**Source Code 4.9** Source Code prosedur mengambil data pada citra

Hasil pengujian pada proses ekstraksi citra dapat ditunjukkan pada gambar 4.3

112,00	64,00	64,00	56,00	64,00	72,00	72,00	80,00	80,00	88,00	88,00	96,00	96,00	96,00	80,00		
56,00	72,00	88,00	72,00	80,00	80,00	80,00	72,00	64,00	88,00	96,00	88,00	80,00	104,00	96,00	104,00	
56,00	72,00	64,00	64,00	72,00	72,00	72,00	80,00	72,00	80,00	88,00	80,00	80,00	88,00	72,00	80,00	
64,00	80,00	56,00	72,00	64,00	72,00	56,00	80,00	88,00	88,00	80,00	72,00	72,00	88,00	88,00	96,00	
72,00	72,00	80,00	72,00	72,00	72,00	80,00	80,00	80,00	80,00	80,00	72,00	96,00	104,00	96,00	88,00	88,00
64,00	72,00	80,00	64,00	64,00	80,00	80,00	80,00	80,00	80,00	64,00	96,00	80,00	96,00	104,00	88,00	96,00
64,00	64,00	64,00	64,00	72,00	72,00	80,00	80,00	80,00	80,00	64,00	88,00	96,00	80,00	88,00	88,00	80,00
56,00	80,00	64,00	72,00	72,00	88,00	72,00	72,00	80,00	88,00	80,00	80,00	80,00	72,00	80,00	88,00	88,00
72,00	56,00	56,00	80,00	72,00	80,00	80,00	72,00	72,00	96,00	88,00	88,00	96,00	88,00	80,00	80,00	80,00
72,00	64,00	64,00	72,00	56,00	72,00	80,00	80,00	72,00	80,00	88,00	80,00	104,00	88,00	88,00	80,00	80,00
72,00	72,00	64,00	88,00	48,00	72,00	72,00	80,00	72,00	72,00	88,00	88,00	72,00	88,00	80,00	104,00	80,00
64,00	72,00	64,00	80,00	72,00	88,00	88,00	80,00	80,00	112,00	88,00	72,00	88,00	80,00	96,00	80,00	80,00
72,00	72,00	64,00	64,00	80,00	64,00	72,00	80,00	80,00	80,00	80,00	96,00	88,00	72,00	96,00	88,00	88,00
56,00	72,00	72,00	64,00	64,00	48,00	80,00	80,00	72,00	72,00	88,00	80,00	96,00	88,00	104,00	80,00	80,00
80,00	80,00	72,00	80,00	64,00	72,00	64,00	80,00	88,00	72,00	96,00	88,00	88,00	72,00	104,00	88,00	88,00
80,00	72,00	72,00	56,00	80,00	72,00	64,00	96,00	88,00	88,00	80,00	72,00	80,00	80,00	80,00	72,00	80,00

Gambar 4.5 Template Iris

Matriks hasil inisiasi di atas dilewatkan pada filter Matrik ini akan ditransformasikan menggunakan filter *lowpass* dan *highpass* seperti pada persamaan 2.2 dan 2.3. Implementasi proses transformasi *haar wavelet* tahap dekomposisi pada baris /filter *lowpass* terhadap template iris adalah sebagai berikut:





```

procedure TFWavelet.WaveBarisClick(Sender: TObject);
var i,j,level:integer;
begin
inisiiasi;
for level:=1 to 5 do
begin
for i:=0 to 31 do
for j:=0 to 15 do
begin
temp[i,j]:=(matriks[i,j*2]+matriks[i,j*2+1])
/sqrt(2);
temp[i,j+16]:=(matriks[i,j*2]-
matriks[i,j*2+1])/sqrt(2);
end;
for i:=0 to 31 do
for j:=0 to 31 do
begin
matriks[i,j]:=temp[i,j];
end;

```

**SourceCode 4.10** Source code Prosedur transformasi *haar wavelet* terhadap baris

Hasil pengujian transformasi *haar wavelet* (dekomposisi) terhadap baris dapat ditunjukkan pada gambar 4.5



326,00	10,00	10,00	6,00	-2,00	10,00	10,00	14,00	-34,00	10,00	10,00	14,00	6,00	26,00	26,00	6,00
330,00	-10,00	-18,00	-22,00	-18,00	2,00	-14,00	-10,00	-30,00	14,00	6,00	2,00	6,00	-6,00	-6,00	-2,00
298,00	-10,00	-2,00	-6,00	-10,00	2,00	-6,00	-10,00	-22,00	-2,00	-2,00	2,00	-10,00	-6,00	10,00	-2,00
304,00	-20,00	4,00	0,00	-4,00	8,00	16,00	-4,00	-32,00	-12,00	4,00	8,00	4,00	-8,00	-8,00	-4,00
326,00	-2,00	-2,00	6,00	-14,00	-6,00	-6,00	2,00	-26,00	6,00	-10,00	-10,00	10,00	10,00	10,00	-6,00
322,00	2,00	-10,00	-10,00	-22,00	18,00	-10,00	-2,00	-30,00	-6,00	-2,00	-10,00	10,00	-6,00	14,00	-2,00
306,00	2,00	-14,00	2,00	-14,00	2,00	-6,00	10,00	-26,00	-2,00	6,00	-2,00	-10,00	-2,00	14,00	-10,00
306,00	-14,00	2,00	-6,00	-6,00	-6,00	2,00	-6,00	-18,00	-10,00	6,00	-10,00	-10,00	-2,00	-10,00	6,00
314,00	-6,00	2,00	2,00	-10,00	-10,00	-10,00	6,00	-30,00	2,00	-6,00	10,00	-10,00	6,00	6,00	22,00
310,00	2,00	-6,00	-2,00	-14,00	-2,00	-2,00	2,00	-30,00	-10,00	-10,00	2,00	6,00	10,00	18,00	14,00
308,00	-24,00	-24,00	4,00	0,00	12,00	4,00	8,00	-24,00	-4,00	4,00	0,00	12,00	-8,00	8,00	12,00
326,00	-6,00	2,00	-18,00	-10,00	-14,00	10,00	-2,00	-22,00	-10,00	-10,00	10,00	-14,00	6,00	-10,00	18,00
312,00	4,00	-8,00	12,00	-8,00	-12,00	8,00	-4,00	-28,00	0,00	12,00	0,00	-4,00	8,00	4,00	-8,00
304,00	12,00	-20,00	-8,00	-16,00	-12,00	4,00	-8,00	-36,00	-8,00	-8,00	4,00	12,00	0,00	16,00	-12,00
322,00	6,00	-14,00	6,00	6,00	2,00	6,00	2,00	-26,00	-22,00	14,00	2,00	10,00	6,00	2,00	-2,00
306,00	2,00	10,00	2,00	-6,00	10,00	10,00	-14,00	-10,00	-2,00	-2,00	14,00	-10,00	14,00	6,00	-10,00

**Gambar 4.6** *template* iris hasil dekomposisi baris level 4

Setelah dilakukan proses dekomposisi pada baris, selanjutnya dilakukan proses dekomposisi pada kolom, implementasinya sebagai berikut:

```

procedure TFWavelet.WaveKolomClick(Sender: TObject);
var i,j,level:integer;

begin
for i:=0 to 31 do
  for j:=0 to 31 do
    Matriks[i,j]:=strtofloat(stringGrid3.Cells[j,i]);

for level:=1 to 5 do
  begin
  for i:=0 to 15 do
    for j:=0 to 31 do
      begin
        temp[i,j] :=(matriks[i*2,j] +
          matriks[i*2+1,j])/sqrt(2);
        temp[i+16,j]:= (matriks[i*2,j] -
          matriks[i*2+1,j])/sqrt(2);
      end;

  for i:=0 to 31 do
    for j:=0 to 31 do
      matriks[i,j]:=temp[i,j];
      if level=1 then
        drawMatriks(stringGrid4,Matriks);
    end;

  drawMatriks(stringGrid5,Matriks);
  drawMatriksWavelet(stringGrid6,Matriks);

```

**Source Code 4.11** *Source Code* Prosedur transformasi *haar wavelet* terhadap kolom

Hasil pengujian transformasi *haar wavelet* terhadap kolom dapat ditunjukkan pada gambar 4.5

1	0	0	0	0	1	1	0	0	0	1	1	1	1	1	1
1	1	0	1	1	0	0	1	0	1	1	0	0	1	1	1
1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1
1	0	1	1	1	0	1	0	1	1	0	1	0	1	0	1
1	0	1	0	1	1	1	1	0	1	0	1	0	0	0	1
0	0	1	1	1	1	0	0	1	1	0	1	0	1	1	1
1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1
1	1	1	0	1	1	1	1	0	0	1	1	1	1	1	1
1	0	1	0	0	1	0	1	0	1	1	0	1	0	1	0
0	1	1	0	0	0	1	1	1	1	0	1	0	1	1	0
1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	0	0	1	0	1	1	1	1
0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0
1	1	1	0	1	0	1	1	1	0	1	0	0	1	1	1
1	1	0	0	1	0	1	1	0	1	1	1	0	0	0	1
0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0

**Gambar 4.7** template iris hasil dekomposisi kolom level 4

Setelah dilakukan proses transformasi selanjutnya data diubah menjadi linier (1 dimensi). Hal ini dilakukan untuk mempermudah proses pelatihan maupun pengenalan. Berikut implementasi prosedur yang berfungsi untuk mengubah matriks data hasil transformasi menjadi linier:

```

for i:=0 To 31 do
  for j:=0 To 31 do
    begin
      temp:=Temp+StringGrid6.Cells[j,i];
    end;

    edit1.Text:=temp;
  
```

**Source Code 4.12** Source Code prosedur mengubah matriks data menjadi data 1 dimensi

Hasil pengujian proses mengubah matriks 16X16 menjadi 1 dimensi adalah sebagai berikut:

```

1000011000111111110110010110011111000001010011111
0111010110101011010111101010001001111001101011111
1111011111011111101111001111111010010101101010011
0001111010110110011111111111100011111001011110111
1110011111101110101110100111110010110111000101110
1110011111101110101110100111110010110111000101110

```

**Gambar 4.8** *template iris 1 dimensi*

Setelah data diubah menjadi linier, maka data disimpan dalam database untuk melakukan proses registrasi penyimpanan data

```

procedure TFWavelet.BtSimpanClick(Sender: TObject);
begin
  ADOQuery2.Close;
  adoquery2.SQL.Text:='insert into data (nama,iris)
  values('+
  QuotedStr(edit2.Text)+' ','+QuotedStr(edit1.Text)+' '));
  ADOQuery2.ExecSQL;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

```

**Source Code 4.13** *Source Code penyimpanan data training ke database*

#### 4.3.5 Pencocokan Iris

Tahap terakhir dari sistem adalah pencocokan iris dengan data latih di *database*. Berikut implementasinya

```

function Hamming(const string1,string2:string):double;
var i,salah:integer;HammingDistance:double;
begin
  salah:=0;
  for i:=0 to 255 do
    if not ( string1[i]=string2[i]) then inc(salah);

  HammingDistance:= Salah/ 256;
  result:= HammingDistance;
end;

```

**Source Code 4.14** *Source Code menghitung nilai hamming distance*

```

function GeserKanan(source:string):string;
var i:integer;
temp:string;
begin
temp:=temp+source[1023]+source[1024];

for i:=1 to 1022 do
temp:=temp+source[i];

result:=temp;

```

**Source Code 4.15** Source Code proses Shift kanan

```

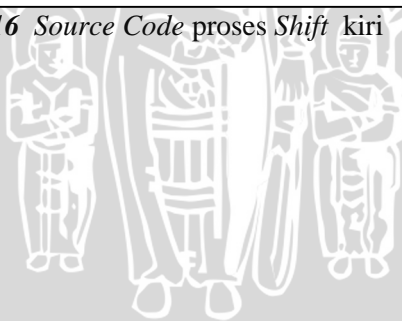
function GeserKiri(source:string):string;
var i:integer;
temp:string;
begin
for i:=3 to 1024 do
temp:=temp+source[i];

temp:=temp+source[1]+source[2];

result:=temp;

```

**Source Code 4.16** Source Code proses Shift kiri





```

procedure TFormUtama.BtRecognizeClick(Sender: TObject);
begin
    StringGrid1.RowCount:=2;
    StringGrid1.Cells[0,0]:='Nama';
    StringGrid1.Cells[1,0]:='Cocok';
    StringGrid1.Cells[2,0]:='Kiri';
    StringGrid1.Cells[3,0]:='Kanan';

    FWavelet.IBQuery1.Close;
    FWavelet.IBQuery1.SQL.Text:='select * from data';
    FWavelet.IBQuery1.open;
    FWavelet.IBQuery1.First;
    while not (FWavelet.IBQuery1.Eof) do
        begin
            StringGrid1.Cells[0,StringGrid1.RowCount-1]:=
                FWavelet.IBQuery1.FieldName('Nama').AsString;
            StringGrid1.Cells[1,StringGrid1.RowCount-1]:=
                format('%2.2f',[cocokkan(edit1.Text,FWavelet.IBQuery
                1.FieldName('iris').AsString)]);
            StringGrid1.Cells[2,StringGrid1.RowCount-1]:=
                format('%2.2f',[cocokkan(hammingkiri(edit1.Text),FWa
                velet.IBQuery1.FieldName('iris').AsString)]);
            StringGrid1.Cells[3,StringGrid1.RowCount-1]:=
                format('%2.2f',[cocokkan(hammingKanan(edit1.Text),FW
                avelet.IBQuery1.FieldName('iris').AsString)]);
            StringGrid1.RowCount:=StringGrid1.RowCount+1;
            FWavelet.IBQuery1.next;
        end;
    end;
end;

```

*Source Code 4.17 Source Code Pencocokan data*

#### **4.4 Implementasi Antarmuka**

Berdasarkan rancangan antarmuka pada bab sebelumnya, maka dibuatlah antarmuka seperti ditunjukkan pada gambar 4.8.



## 4.5 Implementasi Uji Coba

Pada subbab ini akan dilakukan pembahasan mengenai pengujian yang telah dilakukan pada sistem dan hasil pengukuran tingkat kesalahan pengenalan iris pada FIR (*False Inner Rate*) dan FOR (*False Outer Rate*). Pada setiap pengujian dilakukan penambahan data latih/uji sejumlah 5 individu sehingga untuk pengujian pertama dilakukan terhadap 5 data latih/uji awal selanjutnya ditambah lagi 5 data latih berikutnya sampai pengujian keempat didapatkan 20 data latih/uji. Proses pelatihan data latih pada penelitian ini untuk masing-masing individu data latih bersifat terbebas terhadap individu data latih lainnya sehingga penambahan data latih tidak mempengaruhi parameter masing-masing individu data latih yang telah dilatih.



Pengujian juga dilakukan kepada 120 data di luar data latih yang berasal dari individu yang sama (*intraclass*) untuk mendapatkan batas nilai *hamming distance* yang bisa diterima sehingga data yang memiliki nilai di atas ambang *hamming distance* tidak dikenali oleh sistem. Pengujian terakhir dilakukan pada data di luar data latih yang berasal dari individu asing (*interclass*).









### 4.5.1 Hasil pengujian FIR

#### 4.5.1.1 Hasil pengujian FIR terhadap 5 data latih

Hasil pengujian FIR terhadap 5 data latih dapat ditunjukkan pada tabel 4.1.

**Tabel 4.1** Tabel Hasil Pengujian FIR terhadap 5 Data Latih

No	Input	Dikenali	HD	Akurasi	Hasil
	 Aeval1.bmp	 Aeval11.bmp	0,00	100%	Benar

2	 Bryan1.bmp	 Bryan1.bmp	0,00	100%	Benar
3	 Ching1.bmp	 Ching1.bmp	0,00	100%	Benar
	 Hock1.bmp	 Hock1.bmp	0,00	100%	Benar
5	 Chual1.bmp	 chual1.bmp	0,00	100%	Benar








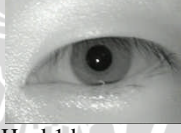
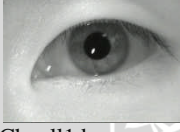

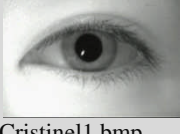
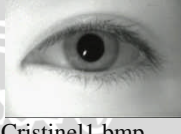
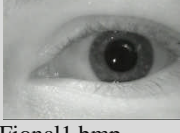
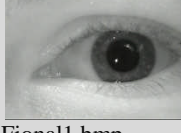
$$\begin{aligned}
 (\text{FAR}) &= \frac{\text{Jumlah yang salah dikenali}}{\text{Banyaknya data}} \\
 &= \frac{0}{5} = 0,00
 \end{aligned}$$

Dari hasil pengujian pada tabel 4.1 dapat diketahui bahwa FIR untuk 5 data latih sebesar 0% , dengan nilai rata-rata *hamming distance* 0. Hal ini menunjukkan pada pengujian 5 data latih memiliki tingkat keakuratan 100 %.





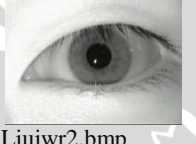

#### 4.5.1.2 Hasil pengujian FIR terhadap 10 data latih

Hasil pengujian FIR terhadap 10 data latih dapat ditunjukkan pada tabel 4.2.

**Tabel 4.2** Tabel Hasil Pengujian FIR terhadap 10 Data Latih

No	Input	Dikenali	HD	Hasil
1	 Aeval1.bmp	 Aeval1.bmp	0,00	Benar
2	 Bryan1.bmp	 Bryan1.bmp	0,00	Benar
3	 Ching1.bmp	 Ching1.bmp	0,00	Benar
4	 Hock1.bmp	 Hock1.bmp	0,00	Benar
5	 Chuall1.bmp	 Chuall1.bmp	0,00	Benar
6.	 Cristinel1.bmp	 Cristinel1.bmp	0,00	Benar
7.	 Fional1.bmp	 Fional1.bmp	0,00	Benar





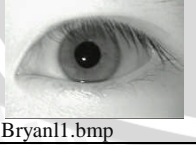
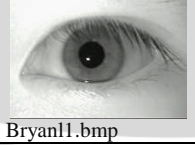
8.	 Zulaikhal1.bmp	 Zulaikhal1.bmp	0,00	Benar
9.	 Lpjl1.bmp	 Lpjl1.bmp	0,00	Benar
10.	 Liujwr2.bmp	 Liujwr2.bmp	0,00	Benar

$$(FIR) = \frac{\text{Benar}}{\text{Benar} + \text{Salah}} = \frac{3}{3} = 1,00$$











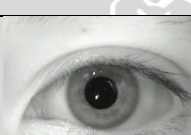

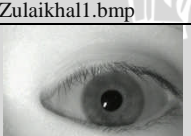

#### 4.5.1.3 Hasil pengujian FIR terhadap 15 data latih












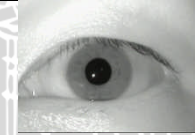
Hasil pengujian FIR terhadap 15 data latih dapat ditunjukkan pada tabel 4.3.

**Tabel 4.3** Tabel Hasil Pengujian FIR terhadap 10 Data Latih

No	Input	Dikenali	HD	Hasil
1	 Aeval1.bmp	 Aeval1.bmp	0,00	Benar
2	 Bryan11.bmp	 Bryan11.bmp	0,00	Benar



3	 Ching11.bmp	 Ching11.bmp	0,00	Benar
4	 Hock1.bmp	 Hock1.bmp	0,00	Benar
5	 Chuall1.bmp	 Chuall1.bmp	0,00	Benar
6.	 Cristinel1.bmp	 Cristinel1.bmp	0,00	Benar
7.	 Fional1.bmp	 Fional1.bmp	0,00	Benar
8.	 Zulaikhal1.bmp	 Zulaikhal1.bmp	0,00	Benar
9.	 Lpjl1.bmp	 Lpjl1.bmp	0,00	Benar

10.	 Liujwr2.bmp	 Liujwr2.bmp	0,00	Benar
11	 Norsuhail1.bmp	 Norsuhail1.bmp	0,00	Benar
12	 Lpjl1.bmp	 Lpjl1.bmp	0,00	Benar
13	 Mahsk11.bmp	 Mahsk11.bmp	0,00	Benar
14	 Mimil1.bmp	 Mimil1.bmp	0,00	Benar
15	 Ngkok11.bmp	 Ngkok11.bmp	0,00	Benar






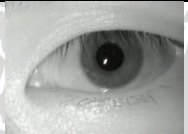
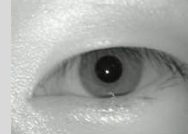
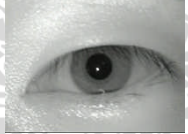




$$(FIR) = \frac{\text{Jumlah yang salah dikenali}}{\text{Banyaknya data}}$$















$$= - = 0,00$$









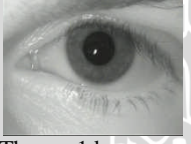





#### 4.5.1.4 Hasil pengujian FIR terhadap 20 data latih

Hasil pengujian FIR terhadap 20 data latih dapat ditunjukkan pada tabel 4.4

**Tabel 4.4** Tabel Hasil Pengujian FIR terhadap 20 Data Latih

No	Input	Dikenali	HD	Hasil
1	 Aeval1.bmp	 Aeval1.bmp	0,00	Benar
2	 Bryan1.bmp	 Bryan1.bmp	0,00	Benar
3	 Ching1.bmp	 Ching1.bmp	0,00	Benar
4	 Hock1.bmp	 Hock1.bmp	0,00	Benar
5	 Chuall1.bmp	 Chuall1.bmp	0,00	Benar
6.	 Cristinel1.bmp	 Cristinel1.bmp	0,00	Benar

7.	 Fional1.bmp	 Fional1.bmp	0,00	Benar
8.	 Zulaikhal1.bmp	 Zulaikhal1.bmp	0,00	Benar
9.	 Lpjl1.bmp	 Lpjl1.bmp	0,00	Benar
10.	 Liujwr2.bmp	 Liujwr2.bmp	0,00	Benar
11	 Norsuhail1.bmp	 Norsuhail1.bmp	0,00	Benar
12	 Lpjl1.bmp	 Lpjl1.bmp	0,00	Benar
13	 Mahskl1.bmp	 Mahskl1.bmp	0,00	Benar

14			0,00	Benar
	Mimil1.bmp	Mimil1.bmp		
15			0,00	Benar
	Ngkok1.bmp	Ngkok1.bmp		
16			0,00	Benar
	Pcl1.bmp	Pcl1.bmp		
17			0,00	Benar
	Ong1.bmp	Ong1.bmp		
18.			0,00	Benar
	Thomas1.bmp	Thomas1.bmp		
19			0,00	Benar
	Siti1.bmp	Siti1.bmp		
20			0,00	Benar
	Sarinal.bmp	Sarinal.bmp		



$$(FIR) = \frac{0}{20} = 0,00$$

Hasil pengujian pada tabel 4.4 menunjukkan bahwa *False Inner Rate* (FIR) untuk 20 data latih sebesar 0,00. Nilai ini didapatkan dengan mengujikan data sampel yang terdapat dalam 20 data latih secara random. Sehingga dapat disimpulkan untuk pengujian *False Inner Rate* (FIR), sistem pengenalan iris memiliki tingkat akurasi 100 %.




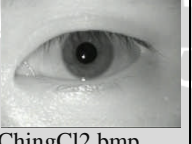

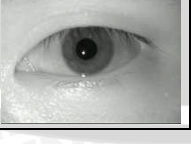
#### 4.5.2 Hasil pengujian FOR

Pada pengujian FOR ini, citra masukan berasal dari citra di luar data latih. Citra masukan ini memiliki format penamaan *nama\_individu\_14* hingga *nama\_individu\_15* untuk iris bagian kiri atau *nama\_individu\_14* hingga *nama\_individu\_15* untuk iris bagian kanan dan dilakukan pengujian secara acak dari rentang tersebut.





##### 4.5.2.1 Hasil pengujian FOR terhadap 5 data latih

Hasil pengujian FOR terhadap 5 data latih dapat ditunjukkan pada tabel 4.5.

**Tabel 4.5** Tabel Hasil Pengujian FOR terhadap 5 Data Latih

No	Input	Dikenali	HD	Hasil
1	 Aeval4.bmp	 Aeval2.bmp	0,32	Benar
2	 Bryan13.bmp	 ChingCl2.bmp	0,42	Salah
3			0,39	Benar



4	Ching4.bmp  Cristine4.bmp	ChingCI2.bmp  Chingcr1.bmp	0,37	Salah
5	 Fionar4 .bmp	 Fional4.bmp	0,28	Benar


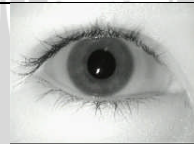

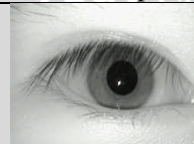
$$\begin{aligned}
 (\text{FOR}) &= \frac{\text{Jumlah yg salah dikenali}}{\text{Banyaknya data}} \\
 &= \frac{2}{5} \times 100 \% = 40 \%
 \end{aligned}$$

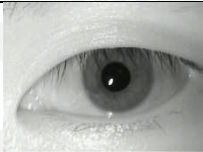
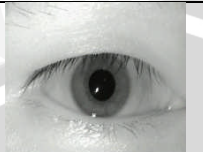


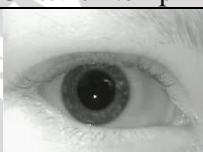
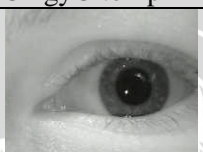


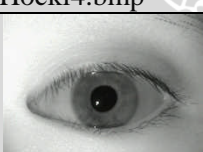

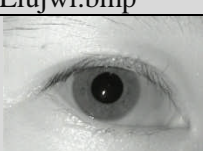

Dari hasil pengujian pada tabel 4.1 dapat diketahui bahwa FOR untuk 5 data latih sebesar 40%.





#### 4.5.2.2 Hasil pengujian FOR terhadap 10 data uji

Hasil pengujian FOR terhadap 10 data uji dapat ditunjukkan pada tabel 4.6

**Tabel 4.6** Tabel Hasil Pengujian FOR terhadap 10 Data uji

No	Input	Dikenali	HD	Hasil
1	 Aeval4.bmp	 Aevar2.bmp	0,32	Benar
2	 Bryanr5.bmp	 Bryanr2.bmp	0,23	Benar

3	 Chingcl5.bmp	 Chingcr2.bmp	0,24	Salah
4	 Cristinel4.bmp	 CingyCr.bmp	0,37	Salah
5	 Fionar4.bmp	 Fional1.bmp	0,39	Benar
6.	 Hockl4.bmp	 hockL1.bmp	0,35	Benar
7.	 Liujw1.bmp	 Cristiner1.bmp	0,30	Salah
8.	 Mahskr5.bmp	 Mahsk11.bmp	0,36	Benar





9.	 Lpjl4.bmp	 Lpjr1.bmp	0,41	Benar
10.	 Zulaikahl3.bmp	 Zulaikahl11.bmp	0,1	Benar











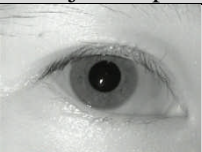



$$\begin{aligned}
 (\text{FOR}) &= \frac{\text{Jumlah yg salah dikenali}}{\text{Banyaknya data}} \\
 &= \frac{3}{10} \times 100 \% = 33 \%
 \end{aligned}$$

#### 4.5.2.3 Hasil pengujian FOR terhadap 15 data uji











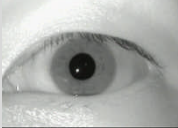

Hasil pengujian FOR terhadap 15 data uji dapat ditunjukkan pada tabel 4.7

**Tabel 4.7** Tabel Hasil Pengujian FOR terhadap 15 Data uji

No	Input	Dikenali	HD	Hasil
1	 Aeval4.bmp	 Aevar2.bmp	0,32	Benar
2	 Bryanr5.bmp	 Bryanr2.bmp	0,23	Benar

3	 Chingcl5.bmp	 NgkokWhl2.bmp	0,34	Salah
4	 Cristiner4.bmp	 Chual1.bmp	0,36	Salah
5	 Fionar4.bmp	 Fional1.bmp	0,28	Benar
6.	 Hockl4.bmp	 hockL1.bmp	0,35	Benar
7.	 Liujuwl.bmp	 Cristiner1.bmp	0,30	Salah
8.	 Mahskr5.bmp	 Mahskl1.bmp	0,36	Benar
9.	 Lpjl4.bmp	 Lpjr1.bmp	0,41	Benar







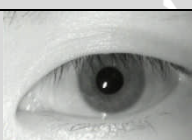



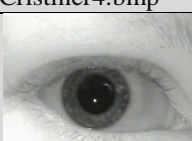

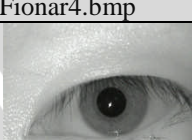
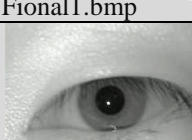
10.	 Zulaikah13.bmp	 Zulaikah11.bmp	0,1	Benar
11	 NorShaidar5.bmp	 Norshaidhr.bmp	0,26	Benar
12	 Nk14.bmp	 Nk12.bmp	0,14	Benar
13	 Chual4.bmp	 Nurshaidh.bmp	0,23	Salah
14	 Mimil4.bmp	 Mimil1.bmp	0,25	Benar
15	 Ngkokw14.bmp	 Ngkokw11.bmp	0,39	Benar

$$\begin{aligned}
 (\text{FOR}) &= \frac{\text{Jumlah yg salah dikenali}}{\text{Banyaknya data}} = \frac{3}{10} \times 100 \% \\
 &= 27 \%
 \end{aligned}$$



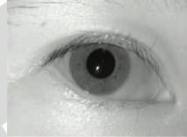





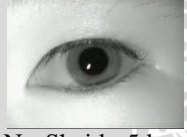
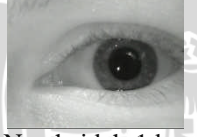




#### 4.5.2.4 Hasil pengujian FOR terhadap 20 data uji







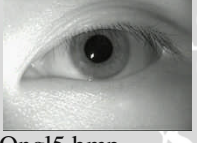







Hasil pengujian FOR terhadap 20 data uji dapat ditunjukkan pada tabel 4.8.

**Tabel 4.8** Tabel Hasil Pengujian FOR terhadap 20 Data uji

No	Input	Dikenali	HD	Hasil
1	 Aeval4.bmp	 Aear2.bmp	0,32	Benar
2	 Bryan5.bmp	 Bryan2.bmp	0,23	Benar
3	 Chingcl5.bmp	 NgkokWhl2.bmp	0,34	Salah
4	 Cristiner4.bmp	 Chual1.bmp	0,36	Salah
5	 Fionar4.bmp	 Fional1.bmp	0,28	Benar
6.	 Hockl4.bmp	 hockL1.bmp	0,35	Benar



7.	 Liujwl.bmp	 Cristiner1.bmp	0,30	Salah
8.	 Mahskr5.bmp	 Mahskl1.bmp	0,36	Benar
9.	 Lpjl4.bmp	 Lpjr1.bmp	0,41	Benar
10.	 Zulaikah13.bmp	 Zulaikah111.bmp	0,1	Benar
11	 NorShaidar5.bmp	 Norshaidahr1.bmp	0,26	Benar
12	 Nkl4.bmp	 Nkl2.bmp	0,12	Benar
13	 Chual4.bmp	 Nursuhaidah.bmp	0,23	Salah

14	 Mimil4.bmp	 Mimil1.bmp	0,25	Benar
15	 Ngkokwl4.bmp	 Ngkokwl1.bmp	0,39	Benar
16	 Pcl5.bmp	 Pcl11.bmp	0,32	Benar
17	 Ongl5.bmp	 Thomasl2.bmp	0,32	Benar
18.	 Thomasl5.bmp	 Hockl2.bmp	0,35	Salah
19	 Sitil3.bmp	 Sitir1.bmp	0,32	Benar
20	 Sarinal5.bmp	 Sarinal2.bmp	0,33	Benar

$$(FOR) = \frac{\text{Jumlah yg salah dikenali}}{\text{Banyaknya data}}$$

$$= \frac{5}{20} \times 100 \% = 25 \%$$

#### 4.5.2.5 Hasil pengujian FOR terhadap 105 data uji

Percobaan pada 105 data yang berasal dari dalam dan luar data uji yang berasal dari Individu di luar data uji dapat ditunjukkan pada tabel 4.9.

**Tabel 4.9** Tabel Hasil Pengujian FOR terhadap 105 Data uji

No	Input	Dikenali Sebagai	<i>tresh old</i>	<i>HD</i>	Hasil
1	aeval4.bmp	aevar2.bmp	90	0,39	Benar
2	lpjl5.bmp	ngkokwhr2.bmp	90	0,4	Salah
3	aevar4.bmp	aevar2.bmp	85	0,3	Benar
4	fional5.bmp	chualsr1.bmp	90	0,39	Salah
5	bryanr4.bmp	bryanr2.bmp	100	0,36	Benar
6	bryanl4.bmp	mimir1.bmp	90	0,35	Salah
7	bryanr3.bmp	bryanr2.bmp	90	0,3	Benar
8	chingycl4.bmp	hockr1.bmp	90	0,36	Salah
9	bryanr5.bmp	bryanr2.bmp	100	0,23	Benar
10	chingycl3.bmp	hockr1.bmp	90	0,35	Salah
11	chingycr3.bmp	liujwr1.bmp	90	0,35	Salah
12	mahskl3.bmp	hockl1.bmp	90	0,36	Salah
13	chingycl5.bmp	ngkokwhl2.bmp	90	0,34	Salah
14	hockl4.bmp	hockl1.bmp	90	0,35	Benar
15	chingycr4.bmp	bryanl1.bmp	90	0,29	salah
16	chualsl4.bmp	norshaidhl1.bmp	75	0,23	Salah
17	chualsl4.bmp	norshaidhl1.bmp	75	0,23	Salah
18	chualsr5.bmp	thomasr2.bmp	90	0,24	Salah
19	fional3.bmp	christinel3.bmp	90	0,3	salah
20	fional4.bmp	fional1.bmp	90	0,28	Benar
21	liujwr4.bmp	zulaikahr2.bmp	90	0,39	Salah
22	fionar3.bmp	thomasr1.bmp	90	0,32	salah
23	fionar5.bmp	fionar2.bmp	90	0,34	benar

24	liujwl3.bmp	hockl1.bmp	100	0,25	salah
25	liujwl4.bmp	nkl12.bmp	90	0,17	salah
26	lpjl4.bmp	lpjr1.bmp	90	0,41	Benar
27	liujwr3.bmp	sitir3.bmp	100	0,33	salah
28	mahskl3.bmp	Mahskl1.bmp	90	0,39	betul
29	liujwr5.bmp	hockl2.bmp	90	0,3	salah
30	lpjr4.bmp	aevar3.bmp	90	0,38	Salah
31	ongbll3.bmp	chingycl1.bmp	90	0,41	Salah
32	sarinal3.bmp	fional1.bmp	85	0,4	Salah
33	ngkowhr4.bmp	mimil2.bmp	110	0,37	Salah
34	mimir4.bmp	lpjl2.bmp	90	0,38	Salah
35	lpjr5.bmp	hockl2.bmp	90	0,34	Salah
36	hockl3.bmp	hockl2.bmp	90	0,32	benar
37	hockr4.bmp	thomasl2.bmp	90	0,35	Salah
38	hockl5.bmp	thomasl2.bmp	90	0,32	Salah
39	hockr3.bmp	hockr1.bmp	90	0,3	salah
40	ngkkwlr3.bmp	chingycr2.bmp	110	0,35	Salah
41	hockr5.bmp	hockr2.bmp	100	0,32	benar
42	mimil5.bmp	chingycl2.bmp	75	0,39	Salah
43	mahskl5.bmp	mahskl2.bmp	85	0,36	Benar
44	mahskl4.bmp	sarinal1.bmp	85	0,25	salah
45	lpjr3.bmp	liujwr1.bmp	90	0,37	Salah
46	mahskr3.bmp	thomasl2.bmp	90	0,3	salah
47	mahskr4.bmp	chualsr2.bmp	90	0,34	Salah
48	mahskr5.bmp	ongblr1.bmp	90	0,34	Salah
49	mimil3.bmp	mimir2.bmp	90	0,31	benar
50	mimil4.bmp	mimil1.bmp	90	0,25	Benar
51	ngkokwhl4.bmp	ngkokwhy11.bmp	110	0,39	Salah
52	mimir3.bmp	mimir2.bmp	90	0,21	Benar
53	mimir5.bmp	mimir1.bmp	90	0,17	Benar
54	ngkokwhl3.bmp	ngkokwhy11.bmp	110	0,27	Benar
55	nkl15.bmp	nkl11.bmp	90	0,39	benar
56	ngkokwh5.bmp	ngkokwh1.bmp	110	0,31	benar
57	ngkowhr5.bmp	ngkokwhyr1.bmp	110	0,35	Benar
58	ongblr5.bmp	thomasr2.bmp	90	0,37	Salah
59	thomasl4.bmp	hockl2.bmp	90	0,35	Benar



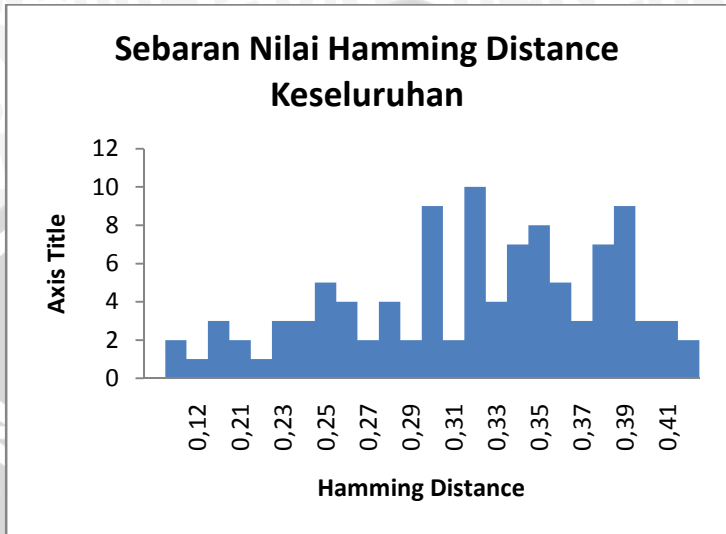
60	nkl13.bmp	nkl12.bmp	90	0,1	benar
61	nkl14.bmp	nkl12.bmp	90	0,12	benar
62	nklr3.bmp	thomasr2.bmp	90	0,39	Salah
63	aeval5.bmp	hockl1.bmp	90	0,4	Salah
64	nklr4.bmp	nklr2.bmp	90	0,26	Benar
65	nklr5.bmp	nklr2.bmp	90	0,29	Benar
66	norshaidl4.bmp	hockl2.bmp	75	0,28	Salah
67	norshaidl5.bmp	pell2.bmp	75	0,27	Salah
68	norshaidr3.bmp	norsuhaidhr1.bmp	75	0,22	Benar
69	norshaidr4.bmp	thomasl2.bmp	75	0,3	salah
70	norshaidr5.bmp	norshaidahr1.bmp	75	0,26	Benar
71	ongbl14.bmp	hockl1.bmp	90	0,32	salah
72	ongbl15.bmp	thomasl2.bmp	90	0,32	salah
73	ongblr3.bmp	ongblr2.bmp	90	0,25	Benar
74	ongblr4.bmp	ongblr2.bmp	90	0,26	Benar
75	lpjl3.bmp	mimir1.bmp	90	0,38	Salah
76	pell3.bmp	chingycr2.bmp	90	0,3	salah
77	pell4.bmp	mimir1.bmp	85	0,32	salah
78	pell5.bmp	pell1.bmp	85	0,32	benar
79	pc1r5.bmp	zulaikahr1.bmp	85	0,44	Salah
80	pc1r4.bmp	chingycl1.bmp	90	0,38	Salah
81	pc1r3.bmp	zulaikahr1.bmp	90	0,44	Salah
82	sarinar5.bmp	fionar2.bmp	85	0,38	Salah
83	liujw15.bmp	lpjr1.bmp	90	0,41	Salah
84	sarinal4.bmp	thomasr2.bmp	85	0,33	Salah
85	sarinal5.bmp	sarinal2.bmp	85	0,33	Benar
86	sarinar3.bmp	fionar2.bmp	85	0,32	salah
87	sarinar4.bmp	thomasr2.bmp	85	0,25	salah
88	thomasr5.bmp	mahskr1.bmp	90	0,38	Salah
89	siti13.bmp	sitir1.bmp	90	0,32	Benar
90	siti14.bmp	bryanl2.bmp	90	0,34	Salah
91	siti15.bmp	mimil2.bmp	90	0,34	Salah
92	sitir4.bmp	sitir1.bmp	90	0,21	Benar
93	sitir5.bmp	pell2.bmp	90	0,26	Salah
94	thomasl3.bmp	ongblr1.bmp	90	0,25	Salah
95	bryanl3.bmp	mimir1.bmp	90	0,36	Salah

96	thomas15.bmp	hock11.bmp	90	0,33	Salah
97	thomasr3.bmp	norsuhaidhr2.bmp	90	0,3	Salah
98	thomasr4.bmp	thomasr2.bmp	90	0,17	Benar
99	zulaikahr4.bmp	chingycl1.bmp	90	0,38	Salah
100	zulaikahl3.bmp	zulaikahl1.bmp	90	0,1	Benar
101	zulaikahl4.bmp	zulaikahl2.bmp	90	0,28	Benar
102	zulaikahl5.bmp	christiner1.bmp	90	0,28	Salah
103	zulaikahr3.bmp	zulaikahr1.bmp	90	0,24	Benar
104	aevar5.bmp	chingycl1.bmp	75	0,39	Salah
105	zulaikahr5.bmp	zulaikahr1.bmp	90	0,24	Benar

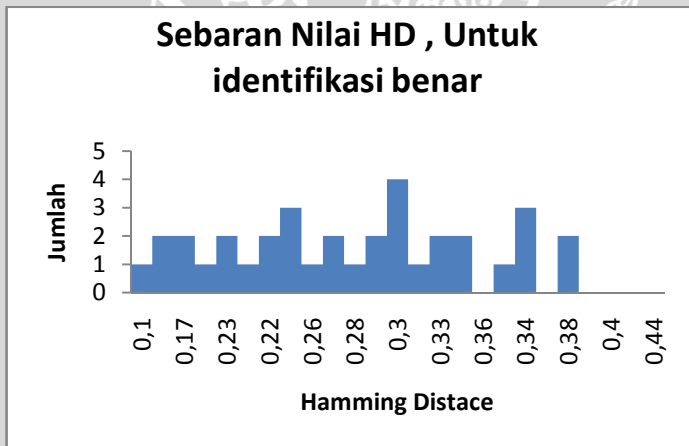
$$\begin{aligned}
 (\text{FOR}) &= \frac{\text{Jumlah yg salah dikenali}}{\text{Banyaknya data}} \\
 &= \frac{67}{105} \times 100 \% = 63 \%
 \end{aligned}$$

Dari Percobaan terhadap 105 data uji didapatkan nilai *hamming distance* dengan persebaran seperti yang tampak pada gambar 4.9,4.10 dan 4.11. Untuk mendapatkan nilai batas *hamming distance* dilakukan perhitungan rata-rata nilai *hamming distance* dan standar deviasi sebagaimana pada tabel 4.10

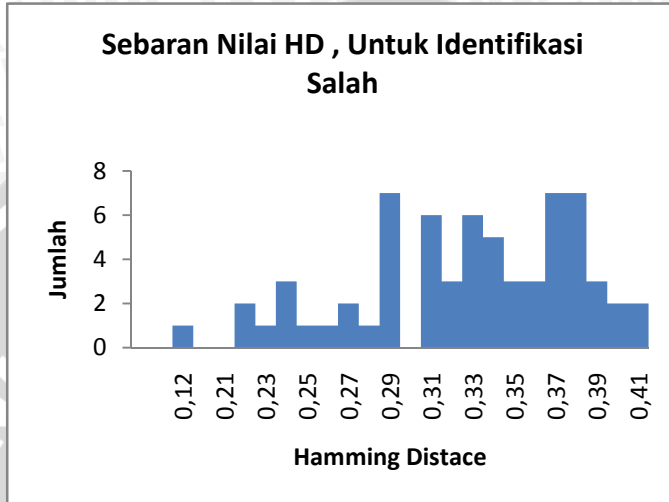




**Gambar 4.10** Diagram batang persebaran nilai *hamming distance* pada semua data



**Gambar 4.11** Diagram batang persebaran nilai *hamming Distance* pada data yang dikenali benar



**Gambar 4.12** Diagram batang persebaran nilai *hamming distance* pada data yang dikenali salah

No	Percobaan	Mean HD	SD	Batas HD	Mean HD (true)	SD	Batas HD	Mean HD (Fals)	SD	Batas HD
1	Percobaan 5 data	0,36	0,05	<b>0,41</b>	0,33	0,05	0,38	<b>0,40</b>	0,04	<b>0,43</b>
2.	Percobaan 10 data	0,31	0,09	<b>0,40</b>	0,31	0,11	0,42	<b>0,30</b>	0,07	<b>0,37</b>
3.	Percobaan 15 data	0,29	0,09	<b>0,38</b>	0,28	0,10	0,38	<b>0,31</b>	0,06	<b>0,37</b>
4.	Percobaan 20 data	0,30	0,08	<b>0,38</b>	0,29	0,08	0,37	<b>0,32</b>	0,05	<b>0,37</b>
5	Percobaan 105 data	0,31	0,08	<b>0,39</b>	0,26	0,09	0,35	<b>0,33</b>	0,05	<b>0,39</b>

**Tabel 4.13** Tabel perbandingan nilai *hamming distance*

Keterangan

Mean HD : Nilai rata-rata *hamming distance*

SD : Standar deviasi

Batas HD : Nilai rata-rata + standar deviasi

### 4.5.3 Hasil Pengujian dengan Nilai Batas *Hamming Distance*

Setelah mendapatkan nilai rata-rata seperti pada tabel 4.10, diambil nilai rata-rata *hamming distance* yang minimum, mean, dan maksimum. Dan dilakukan pengujian untuk masing-masing nilai sebagai batas/ambang nilai *hamming distance*. Pada pengujian ini memberikan perlakuan khusus yakni jika template iris memiliki nilai *hamming distance* lebih dari ambang maka data tidak dikenali sehingga bisa didapatkan FAR (*False Acceptance Rate*) dan FRR (*False Reject Rate*).

#### 4.5.3.1 Pengujian terhadap Data Uji Asing

##### 4.5.3.1.1 Pengujian dengan Batas Nilai *Hamming Distance* = 0,26

Pengujian terhadap data asing dilakukan untuk mengetahui FAR (*False Acceptance Rate*). Pengujian pertama dilakukan dengan batas nilai *hamming distance* minimum yakni 0,26. Hasil pengujian dapat ditunjukkan pada tabel 4.11.

**Tabel 4.11** Tabel Hasil Pengujian FAR terhadap 75 Data Uji Asing

No	Input	Dikenali /Tidak	<i>Threshold</i>	<i>HD</i>
1	chongpk11.bmp	0	110	0,41
2	chongpk12.bmp	0	110	0,44
3	chongpk13.bmp	0	110	0,37
4	chongpk14.bmp	0	110	0,37
5	chongpk15.bmp	0	110	0,39
6	chongpk11.bmp	0	90	0,43
7	chongpk12.bmp	0	90	0,44
8	chongpk13.bmp	0	90	0,42
9	chongpk14.bmp	0	90	0,43
10	chongpk15.bmp	0	90	0,39
11	chongpkr2.bmp	0	90	0,36
12	chongpkr1.bmp	0	90	0,44
13	chongpkr3.bmp	0	110	0,36
14	chongpkr4.bmp	0	110	0,44

15	chongpkr5.bmp	0	110	0,42
16	minglil1.bmp	0	110	0,4
17	minglil2.bmp	0	110	0,39
18	minglil1.bmp	0	90	0,43
19	minglil2.bmp	0	90	0,41
20	minglil3.bmp	0	90	0,44
21	minglil4.bmp	0	90	0,37
22	minglil5.bmp	0	90	0,37
23	minglir1.bmp	0	90	0,4
24	minglir2.bmp	0	90	0,44
25	minglir3.bmp	0	90	0,44
26	minglir5.bmp	0	90	0,43
27	minglir4.bmp	0	90	0,43
28	philip1.bmp	0	90	0,36
29	philip2.bmp	0	90	0,43
30	philip4.bmp	0	90	0,41
31	philip5.bmp	0	90	0,4
32	philipr1.bmp	0	90	0,41
33	philipr2.bmp	0	90	0,41
34	philipr3.bmp	0	90	0,36
35	philipr4.bmp	0	90	0,41
36	philipr5.bmp	0	90	0,44
37	suzailil1.bmp	0	90	0,32
38	suzailil2.bmp	0	90	0,39
39	suzailil3.bmp	0	75	0,32
40	suzailil4.bmp	0	75	0,36
41	suzailil5.bmp	0	75	0,34
42	suzailir1.bmp	1	75	0,2
43	suzailir2.bmp	0	75	0,39
44	suzailir3.bmp	0	75	0,3
45	suzailir4.bmp	1	75	0,21
46	suzailir5.bmp	1	75	0,23
47	tanwnl1.bmp	0	75	0,44
48	tanwnl1.bmp	0	90	0,41
49	tanwnl2.bmp	0	90	0,42
50	tanwnl3.bmp	0	90	0,41

51	tanwnl4.bmp	0	90	0,41
52	tanwnl5.bmp	0	90	0,36
53	tanwnl1.bmp	0	75	0,44
54	tanwnl2.bmp	0	75	0,42
55	tanwnr1.bmp	0	75	0,28
56	tanwnr1.bmp	0	90	0,28
57	tanwnr2.bmp	0	90	0,43
58	tanwnr3.bmp	0	90	0,44
59	tanwnr4.bmp	0	90	0,44
60	tanwnr5.bmp	0	90	0,44
61	tickl1.bmp	0	90	0,27
62	tickl2.bmp	0	90	0,34
63	tickl3.bmp	1	90	0,26
64	tickl4.bmp	0	90	0,29
65	tickl5.bmp	0	90	0,35
66	tickr1.bmp	0	90	0,34
67	tickr2.bmp	0	90	0,35
68	tickr3.bmp	0	90	0,31
69	tickr4.bmp	0	90	0,36
70	tickr5.bmp	0	90	0,36
71	tingcyl1.bmp	0	90	0,37
72	tingcyl2.bmp	0	90	0,37
73	tingcyl3.bmp	0	90	0,4
74	tingcyl4.bmp	0	90	0,28
75	tingcyl5.bmp	0	90	0,32

Keterangan

1=Diterima/dikenali

0=Tidak diterima/dikenali

Data uji asing dalam sistem seharusnya ditolak oleh sistem, jika data diterima/dikenali oleh sistem maka dianggap sebagai suatu kesalahan yakni *False Accept Rate (FAR)*. Tabel 4.11 menunjukkan nilai FAR sebagai berikut:

$$(FAR) = \frac{\text{Jumlah yg salah diterima}}{\text{Banyaknya data}} = \frac{4}{75} = 0,05$$

#### 4.5.3.1.2 Pengujian dengan Batas Nilai *Hamming Distance* = 0,37

**Tabel 4.12** Tabel Hasil Pengujian FAR terhadap 75 data uji asing2

No	Input	Dikenali /Tidak	<i>Threshold</i>	<i>HD</i>
1	chongpk11.bmp	0	110	0,41
2	chongpk12.bmp	0	110	0,44
3	chongpk13.bmp	0	110	0,37
4	chongpk14.bmp	0	110	0,37
5	chongpk15.bmp	0	110	0,39
6	chongpk11.bmp	0	90	0,43
7	chongpk12.bmp	0	90	0,44
8	chongpk13.bmp	0	90	0,42
9	chongpk14.bmp	0	90	0,43
10	chongpk15.bmp	0	90	0,39
11	chongpkr2.bmp	0	90	0,36
12	chongpkr1.bmp	0	90	0,44
13	chongpkr3.bmp	0	110	0,36
14	chongpkr4.bmp	0	110	0,44
15	chongpkr5.bmp	0	110	0,42
16	mingli11.bmp	0	110	0,4
17	mingli2.bmp	0	110	0,39
18	mingli11.bmp	0	90	0,43
19	mingli2.bmp	0	90	0,41
20	mingli3.bmp	0	90	0,44
21	mingli4.bmp	0	90	0,37
22	mingli5.bmp	0	90	0,37
23	minglir1.bmp	0	90	0,4
24	minglir2.bmp	0	90	0,44
25	minglir3.bmp	0	90	0,44
26	minglir5.bmp	0	90	0,43
27	minglir4.bmp	0	90	0,43
28	philip11.bmp	0	90	0,36



29	philipl2.bmp	0	90	0,43
30	philipl4.bmp	0	90	0,41
31	philipl5.bmp	0	90	0,4
32	philipr1.bmp	0	90	0,41
33	philipr2.bmp	0	90	0,41
34	philipr3.bmp	0	90	0,36
35	philipr4.bmp	0	90	0,41
36	philipr5.bmp	0	90	0,44
37	suzailil1.bmp	1	90	0,32
38	suzailil2.bmp	0	90	0,39
39	suzailil3.bmp	1	75	0,32
40	suzailil4.bmp	0	75	0,36
41	suzailil5.bmp	1	75	0,34
42	suzailir1.bmp	1	75	0,2
43	suzailir2.bmp	0	75	0,39
44	suzailir3.bmp	1	75	0,3
45	suzailir4.bmp	1	75	0,21
46	suzailir5.bmp	1	75	0,23
47	tanwnl1.bmp	0	75	0,44
48	tanwnl1.bmp	0	90	0,41
49	tanwnl2.bmp	0	90	0,42
50	tanwnl3.bmp	0	90	0,41
51	tanwnl4.bmp	0	90	0,41
52	tanwnl5.bmp	0	90	0,36
53	tanwnl1.bmp	0	75	0,44
54	tanwnl2.bmp	0	75	0,42
55	tanwnr1.bmp	1	75	0,28
56	tanwnr1.bmp	1	90	0,28
57	tanwnr2.bmp	0	90	0,43
58	tanwnr3.bmp	0	90	0,44
59	tanwnr4.bmp	0	90	0,44
60	tanwnr5.bmp	0	90	0,44
61	tickl1.bmp	1	90	0,27
62	tickl2.bmp	1	90	0,34
63	tickl3.bmp	1	90	0,26
64	tickl4.bmp	1	90	0,29

65	tickl5.bmp	1	90	0,35
66	tickr1.bmp	1	90	0,34
67	tickr2.bmp	1	90	0,35
68	tickr3.bmp	1	90	0,31
69	tickr4.bmp	0	90	0,36
70	tickr5.bmp	0	90	0,36
71	tingcy11.bmp	0	90	0,37
72	tingcy12.bmp	0	90	0,37
73	tingcy13.bmp	0	90	0,4
74	tingcy14.bmp	0	90	0,28
75	tingcy15.bmp	1	90	0,32

Keterangan

1=diterima/dikenali

0=Tidak diterima/dikenali

Dari tabel 4.12 bisa dihitung nilai FIR (*False Accept Rate*) sebagai berikut:

$$\begin{aligned}
 (\text{FAR}) &= \frac{\text{Jumlah yg salah diterima}}{\text{Banyaknya data}} \\
 &= 18/75 = 0,24
 \end{aligned}$$

#### 4.5.3.1.3 Pengujian dengan Batas Nilai *Hamming Distance* = 0,41

**Tabel 4.13** Tabel Hasil Pengujian FAR terhadap 75 data uji asing

No	Input	Dikenali /Tidak	<i>Treshold</i>	<i>HD</i>
1	chongpk11.bmp	0	110	0,41
2	chongpk12.bmp	0	110	0,44
3	chongpk13.bmp	1	110	0,37
4	chongpk14.bmp	1	110	0,37
5	chongpk15.bmp	1	110	0,39
6	chongpk11.bmp	0	90	0,43
7	chongpk12.bmp	0	90	0,44

8	chongpk13.bmp	0	90	0,42
9	chongpk14.bmp	0	90	0,43
10	chongpk15.bmp	1	90	0,39
11	chongpkr2.bmp	1	90	0,36
12	chongpkr1.bmp	0	90	0,44
13	chongpkr3.bmp	1	110	0,36
14	chongpkr4.bmp	0	110	0,44
15	chongpkr5.bmp	0	110	0,42
16	minglil1.bmp	1	110	0,4
17	minglil2.bmp	1	110	0,39
18	minglil1.bmp	0	90	0,43
19	minglil2.bmp	0	90	0,41
20	minglil3.bmp	0	90	0,44
21	minglil4.bmp	1	90	0,37
22	minglil5.bmp	1	90	0,37
23	minglir1.bmp	1	90	0,4
24	minglir2.bmp	0	90	0,44
25	minglir3.bmp	0	90	0,44
26	minglir5.bmp	0	90	0,43
27	minglir4.bmp	0	90	0,43
28	philip11.bmp	1	90	0,36
29	philip12.bmp	0	90	0,43
30	philip14.bmp	0	90	0,41
31	philip15.bmp	1	90	0,4
32	philipr1.bmp	0	90	0,41
33	philipr2.bmp	0	90	0,41
34	philipr3.bmp	1	90	0,36
35	philipr4.bmp	0	90	0,41
36	philipr5.bmp	0	90	0,44
37	suzailil1.bmp	1	90	0,32
38	suzailil2.bmp	1	90	0,39
39	suzailil3.bmp	1	75	0,32
40	suzailil4.bmp	1	75	0,36
41	suzailil5.bmp	1	75	0,34
42	suzailir1.bmp	1	75	0,2
43	suzailir2.bmp	1	75	0,39

44	suzailir3.bmp	1	75	0,3
45	suzailir4.bmp	1	75	0,21
46	suzailir5.bmp	1	75	0,23
47	tanwnl1.bmp	0	75	0,44
48	tanwnl1.bmp	0	90	0,41
49	tanwnl2.bmp	0	90	0,42
50	tanwnl3.bmp	0	90	0,41
51	tanwnl4.bmp	0	90	0,41
52	tanwnl5.bmp	1	90	0,36
53	tanwnl1.bmp	0	75	0,44
54	tanwnl2.bmp	0	75	0,42
55	tanwnr1.bmp	1	75	0,28
56	tanwnr1.bmp	1	90	0,28
57	tanwnr2.bmp	0	90	0,43
58	tanwnr3.bmp	0	90	0,44
59	tanwnr4.bmp	0	90	0,44
60	tanwnr5.bmp	0	90	0,44
61	tickl1.bmp	1	90	0,27
62	tickl2.bmp	1	90	0,34
63	tickl3.bmp	1	90	0,26
64	tickl4.bmp	1	90	0,29
65	tickl5.bmp	1	90	0,35
66	tickr1.bmp	1	90	0,34
67	tickr2.bmp	1	90	0,35
68	tickr3.bmp	1	90	0,31
69	tickr4.bmp	1	90	0,36
70	tickr5.bmp	1	90	0,36
71	tingcyl1.bmp	1	90	0,37
72	tingcyl2.bmp	1	90	0,37
73	tingcyl3.bmp	1	90	0,4
74	tingcyl4.bmp	1	90	0,28
75	tingcyl5.bmp	1	90	0,32

Keterangan

1=diterima/dikenali

0=Tidak diterima/dikenali

Dari tabel 4.13 bisa dihitung nilai FIR (*False Accept Rate*) sebagai berikut:

$$(FAR) = \frac{\text{Jumlah yg salah diterima}}{\text{Banyaknya data}} = \frac{50}{75} = 0,67$$

#### 4.5.3.2 Hasil Pengujian terhadap Data Latih

Dari tabel 4.1 hingga tabel 4.5 bisa diketahui semua nilai *hamming distance* di bawah ambang/batas nilai *hamming distance* sehingga semua data diterima dan tidak ada satupun penolakan . Dari pengujian 4.5.1 bisa didapat nilai FRR (*False Reject Rate*) yakni kesalahan menolak data yang seharusnya dikenali sebagai berikut:

$$(FAR) = \frac{\text{Jumlah yg salah ditolak}}{\text{Banyaknya data}} = \frac{0}{20} = 0,00$$

#### 4.5.3.3 Hasil Pengujian terhadap Data Uji *Intraclass*

**Tabel 4.14** Tabel Hasil Pengujian FRR terhadap 75 data uji yang berasal dari individu yang sama dengan data latih.

No.	Input	HD	Dikenali/ Tidak	Dikenali/ Tidak	Dikenali/ Tidak
1	aeval4.bmp	0,39	0	0	1
2	lpjl5.bmp	0,4	0	0	1
3	aevar4.bmp	0,3	0	1	1
4	fional5.bmp	0,39	0	0	1
5	bryanr4.bmp	0,36	0	0	1
6	bryanl4.bmp	0,35	0	1	1
7	bryanr3.bmp	0,3	0	1	1
8	chingycl4.bmp	0,36	0	0	1
9	bryanr5.bmp	0,23	1	1	1
10	chingycl3.bmp	0,35	0	1	1



11	chingycr3.bmp	0,35	0	1	1
12	mahskl3.bmp	0,36	0	0	1
13	chingycl5.bmp	0,34	0	1	1
14	hockl4.bmp	0,35	0	1	1
15	chingycr4.bmp	0,29	0	1	1
16	chualsl4.bmp	0,23	1	1	1
17	chualsl4.bmp	0,23	1	1	1
18	chualsr5.bmp	0,24	1	1	1
19	fional3.bmp	0,3	0	1	1
20	fional4.bmp	0,28	0	1	1
21	liujwr4.bmp	0,39	0	0	1
22	fionar3.bmp	0,32	0	1	1
23	fionar5.bmp	0,34	0	1	1
24	liujwl3.bmp	0,25	1	1	1
25	liujwl4.bmp	0,17	1	1	1
26	lpjl4.bmp	0,41	0	0	1
27	liujwr3.bmp	0,33	0	1	1
28	mahskl3.bmp	0,39	0	0	1
29	liujwr5.bmp	0,3	0	1	1
30	lpjr4.bmp	0,38	0	0	1
31	ongbll3.bmp	0,41	0	0	1
32	sarinal3.bmp	0,4	0	0	1
33	ngkowhr4.bmp	0,37	0	0	1
34	mimir4.bmp	0,38	0	0	1
35	lpjr5.bmp	0,34	0	1	1
36	hockl3.bmp	0,32	0	1	1
37	hockr4.bmp	0,35	0	1	1
38	hockl5.bmp	0,32	0	1	1
39	hockr3.bmp	0,3	0	1	1
40	ngkkwhr3.bmp	0,35	0	1	1
41	hockr5.bmp	0,32	0	1	1
42	mimil5.bmp	0,39	0	0	1
43	mahskl5.bmp	0,36	0	0	1
44	mahskl4.bmp	0,25	1	1	1
45	lpjr3.bmp	0,37	0	0	1
46	mahskr3.bmp	0,3	0	1	1



47	mahskr4.bmp	0,34	0	1	1
48	mahskr5.bmp	0,34	0	1	1
49	mimil3.bmp	0,31	0	1	1
50	mimil4.bmp	0,25	1	1	1
51	ngkokwhl4.bmp	0,39	0	0	1
52	mimir3.bmp	0,21	1	1	1
53	mimir5.bmp	0,17	1	1	1
54	ngkokwhl3.bmp	0,27	0	1	1
55	nkl15.bmp	0,39	0	0	1
56	ngkokwhl5.bmp	0,31	0	1	1
57	ngkowhr5.bmp	0,35	0	1	1
58	ongblr5.bmp	0,37	0	0	1
59	thomasl4.bmp	0,35	0	1	1
60	nkl13.bmp	0,1	1	1	1
61	nkl14.bmp	0,12	1	1	1
62	nklr3.bmp	0,39	0	0	1
63	aeval5.bmp	0,4	0	0	1
64	nklr4.bmp	0,26	1	1	1
65	nklr5.bmp	0,29	0	1	1
66	norshaidl4.bmp	0,28	0	1	1
67	norshaidl5.bmp	0,27	0	1	1
68	norshaidr3.bmp	0,22	1	1	1
69	norshaidr4.bmp	0,3	0	1	1
70	norshaidr5.bmp	0,26	1	1	1
71	ongbll4.bmp	0,32	0	1	1
72	ongbll5.bmp	0,32	0	1	1
73	ongblr3.bmp	0,25	1	1	1
74	ongblr4.bmp	0,26	1	1	1

#### 4.5.3.3.1 Hasil Pengujian dengan nilai batas *hamming distance* = 0,26

Dari tabel 4.14 bisa dihitung nilai FRR (*False Reject Rate*) sebagai berikut:

$$(FAR) = \frac{1}{1} = 0,79$$

#### 4.5.3.3.2 Hasil Pengujian dengan nilai batas *hamming distance* = 0,35

Dari tabel 4.14 bisa dihitung nilai FRR (*False Reject Rate*) sebagai berikut:

$$(FAR) = \frac{10}{39} = 0,29$$

#### 4.5.3.3.3 Hasil Pengujian dengan nilai batas *hamming distance* = 0,41

Dari tabel 4.14 bisa dihitung nilai FRR (*False Reject Rate*) sebagai berikut:

$$(FAR) = \frac{0}{39} = 0,00$$

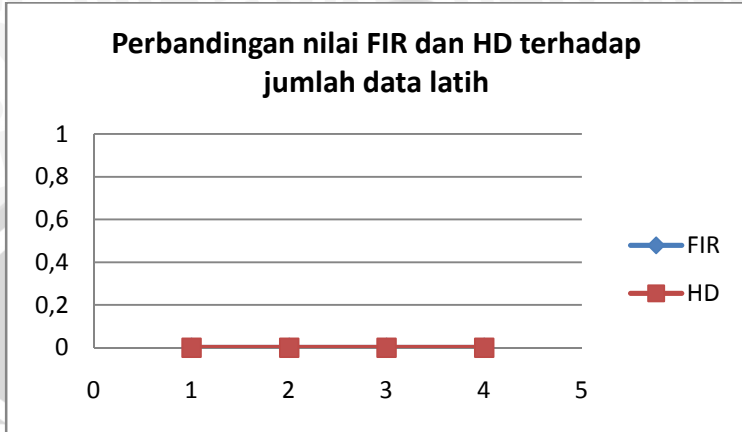
### 4.6 Analisis Hasil

#### 4.6.1 Analisis Hasil Pengujian dengan Nilai Hamming Distance Terkecil

Pengujian dengan *hamming distance* terkecil ini dilakukan terhadap data latih dan terhadap data uji *intraclass*.

##### 4.6.1.1 Analisis Hasil Pengujian terhadap Data Latih

Dari perhitungan FIR untuk 5 sampai 20 data latih dapat dibuat suatu diagram perbandingan FIR seperti yang ditunjukkan pada gambar 4.4.



**Gambar 4.13** Grafik Perbandingan Nilai FIR dengan Jumlah Data Latihan

Dari gambar 4.7 dapat diketahui bahwa nilai FIR pada data latihan yang berjumlah 5,10,15, dan 20 adalah 0 % dan keberhasilan identifikasi 100 %.

Jika dibandingkan dengan penelitian yang dilakukan sebelumnya. Tingkat akurasi dari sistem pengenalan iris ini sama dengan yang dilakukan oleh selzer dengan perbedaan data latihan yakni memiliki tingkat akurasi tertinggi. Pada penelitian ini menggunakan data MMU (Multi Media University Database) dan gagal digunakan untuk data CASIA, sedangkan penelitian Selzer gagal untuk data MMU dan berhasil untuk data CASIA. Berikut tabel perbandingan penelitian yang telah dilakukan :

**Tabel 4.15** Tabel Perbandingan dengan Penelitian Lain

Penelitian	Parameter	Keberhasilan Identifikasi	Keterangan
Selzer	<i>Threshold</i> = 120	100%	Tidak seluruh data CASIA V1.0 digunakan dalam penelitian , gagal untuk seluruh data MMU

#### 4.15 Tabel Perbandingan dengan Penelitian Lain (Lanjutan)

Nur Maulidiyah	<i>Threshold</i> antara 75-100	100%	Tidak semua data MMU bisa digunakan dalam penelitian, gagal untuk data CASIA
Wildes	Hysteresis <i>Thresholds</i> , tinggi =44, rendah=39, kernel Gaussian =5	86,49%	Dilakukan oleh Proenca dan Alexandre
Henry Dwiyasa	Luasan pupil minimal=500	84,25%	-
Masek	Dimensi kernel Gaussian =5, Parameter Kovesi = 39.34	83,92%	-
John Daughman	-	54,44%	Dilakukan oleh Proenca dan Alexandre
Maimuna	-	5,98%	-

Hal ini disebabkan oleh beberapa faktor, yaitu:

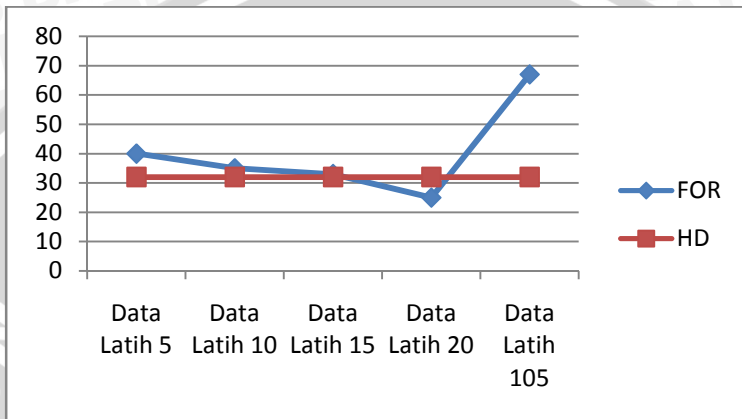
1. Proses yang jelas dan terurut memberikan hasil yang sama
2. Proses transformasi *haar wavelet* tidak menyebabkan *noise*

##### 4.6.1.2 Analisis Hasil Pengujian terhadap Data Uji

Dari hasil pengujian pada tabel 4.8 dapat diketahui bahwa FOR untuk 20 data latih sebesar 35%. Nilai FOR untuk 20 data latih ini dapat lebih besar dari 35 % karena data uji diambil secara acak.

Dari perhitungan FOR dan *hamming distance* untuk 5 sampai 20 data latih dapat dibuat suatu diagram

perbandingan FOR dan *hamming distance* seperti yang ditunjukkan pada gambar 4.5.



**Gambar 4.14** Grafik Perbandingan Nilai FOR dengan Jumlah Data Uji

Dari gambar 4.10 dapat diketahui bahwa nilai FOR tidak selalu semakin besar jika jumlah data diperbesar sehingga tingkat keakuratan pada penelitian ini terhadap data uji tidak berpengaruh terhadap banyaknya data uji karena sifatnya yang acak. Tetapi hal ini lebih dipengaruhi karena pola iris yang hampir sama. Hasil pengujian FOR menunjukkan lebih besar daripada nilai FIR karena citra uji pada FOR karena data yang diujikan meskipun berasal dari individu yang sama memiliki tingkat pencahayaan dan sudut pengambilan yang berbeda.











Secara umum tingkat akurasi ujian FOR ini disebabkan oleh beberapa faktor yaitu:

1. Citra pada MMU (*Multimedia University*) memiliki pencahayaan dan posisi pengambilan gambar yang berbeda-beda sehingga pada saat proses binerisasi, beberapa citra yang bulu mata dan kelopak mata terkadang ikut terdeteksi sehingga nilai ambang harus diturunkan. Hal ini akan mempengaruhi proses normalisasi dan ekstraksi citra.



2. Penerapan lokalisasi dengan ambang (*threshol*) yang berbeda-beda menyebabkan pada proses lokalisasi saat *training* (pelatihan) dan *recognize* (pengenalan) menghasilkan *template* iris yang jauh berbeda dengan citra dalam data latih. Ambang yang berbeda disebabkan oleh faktor pertama.
3. Pada proses crop, karena ada perbedaan iris yang dilokalisasi maka daerahnya tidak sama, sehingga sangat dimungkinkan hasil ekstraksi berbeda

**Tabel 4.16** Citra Mata Bryan

Kiri	Kanan
	
	
	
	
	



#### 4.6.2 Analisis Hasil Pengujian dengan Nilai Batas Hamming Distance

**Tabel 4.17** Perbandingan Nilai FAR dan FRR

	Batas 0,16	Batas 0,35	Batas 0,41
FAR	0,05	0,24	0,67
FRR	0,78	0,29	0,00
EER	0,73	0,05	0,67

Dari tabel 4.16 terlihat FAR terus meningkat sebanding dengan besarnya nilai batas hamming distance, sebaliknya nilai FRR semakin menurun.

Dari sana bisa dicari nilai EER terkecil agar sistem meelakukan kesalahan penerimaan dan penolakan sekecil mungkin sehingga mendapatkan tingkat akurasi sebesar mungkin. Dari tabel 4.16 menunjukkan nilai EER terkecil adalah pada ambang nilai 0,35 . Sehingga bisa disimpulkan pada pengujian nilai batas 0,35 adalah yang paling efektif dalam pengenalan iris mata.

UNIVERSITAS BRAWIJAYA



## BAB V PENUTUP

### 5.1 Kesimpulan

Kesimpulan yang diperoleh dari hasil pengerjaan skripsi ini adalah:

1. Sistem Pengenalan iris mata diimplementasikan dengan beberapa tahapan yakni lokalisasi iris, normalisasi, ekstraksi ciri dengan *transformasi haar wavelet* dan pencocokan dengan *hamming distance* terhadap citra *grayscale*.
2. Berdasarkan pengujian yang dilakukan terhadap data latih didapatkan rata-rata nilai *hamming distance* adalah 0,00 sedangkan untuk pengujian terhadap data uji *intraclass* nilai rata-rata *hamming distance* untuk hasil tes benar yang didapatkan adalah 0,32.
3. Tingkat akurasi terbaik untuk pengenalan iris dengan pengujian terhadap data latih didapatkan rata-rata tingkat akurasi sebesar 100% sedangkan pengenalan iris dengan pengujian terhadap data uji *intraclass* didapatkan tingkat akurasi terbaik adalah 75%.
4. Nilai EER terbaik pada pengujian terhadap data uji *intraclass* dan *interclass* dengan batas nilai *hamming distance* tertentu adalah pada batas nilai *hamming distance* sama dengan 0,35.

### 5.2 Saran

Saran yang diberikan untuk pengembangan penelitian ini antara lain:

1. Sistem pengenalan iris ini belum mampu menghilangkan *noise* berupa bulu mata sehingga pada penelitian selanjutnya diharapkan ada proses penghilangan *noise* yang berupa bulu mata.

2. Melakukan perbaikan citra terlebih dahulu sebelum proses lokalisasi yang berfungsi melakukan perataan pencahayaan bagi citra iris mata
3. Menggunakan metode lain dalam setiap tahap pengenalan iris baik proses lokalisasi, ekstraksi fitur, maupun pengenalan.

UNIVERSITAS BRAWIJAYA



## DAFTAR PUSTAKA

Achmad, Balza. 2005. *Teknik Pengolahan Citra Digital Menggunakan Delphi*. Yogyakarta: Andi Publishing.

Agus. 2004. *Perbandingan 3 Metode Iris Scan* . Jakarta: Jurnal Ilmiah Ilmu Komputer .

Arifin, Zainal, 2008. *A Course On Pattern Recognition*. Surabaya: Teknik Informatika ITS.

Daughman, John. 1993. *How Iris Recognition Work* .IEEE Transaction on Circuit And Systems for Video Technology, Vol 14, no 1

Fahmi.2007. *Perancangan Algoritma Perancangan Citra Mata*. Medan: Karya Ilmiah Universitas Sumatera Utara.

Ganorkar, Sanjay dan Gathol, Ashok. 2004. *Iris Recognition : An Emerging Biometric Technology*. Department of Electronics and Telecommunication Engineering Sinhgad College of Engineering Pune University.

Kaplan, Ian. 2004. *Applying the Haar Wavelet Transform to Time Series Information*. [http: // www. bearcave.com /misl/misl\\_tech/ wavelets/ haar.html](http://www.bearcave.com/misl/misl_tech/wavelets/haar.html) . 14 Maret 2009

Junquerira, Carlos dkk. 1988. *Basic Histology 8th edition*. McGrawhill.

Ko, Jong-Gook, dkk. 2007. *A Novel and Efficient Feature Extraction Method for Iris Recognition*. ETRY Journal

Maimunah. 2007. *Sistem Pengenalan Iris Mata dengan Menggunakan Transformasi haar wavelet*. SNATI Universitas Muhammadiyah Magelang.

Mallat, Sthephen.2000. *A wavelet tour of signal processing* 2nd Edition Academic Press.

Masek, Libor. 2003.*Recognition of Human Iris Pattern for Biometric Indentification*. University Western of Australia. Crawley.

Mei, Xie. 2006. *Iris Recognition Technique*. School of Electronic Engineering, University of Electronic Science and Technology of China .

Minarni. 2004. *Klasifikasi Sidik Jari dengan Pemrosesan Awal Transformasi haar wavelet*. Transmisi, Vol. 8, No. 2.

Murni, Aniati dan Dina Cahyati. 2003. *Pengolahan Citra Digital*. Fakultas Ilmu Komputer Universitas Indonesia.









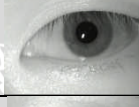
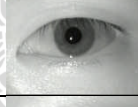
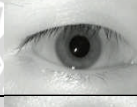

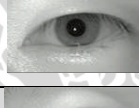
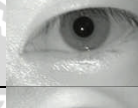
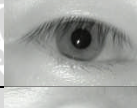
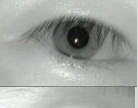

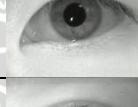





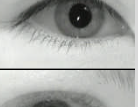
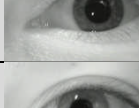
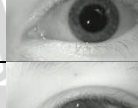
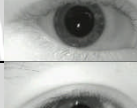
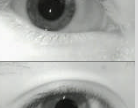
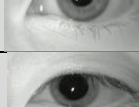
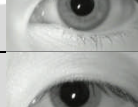
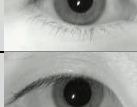
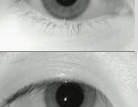
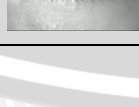



Stollnitz, Eric. 2001. *Wavelets for Computer Graphics: Theory and Application*. Morgan Kauffman Publisher.





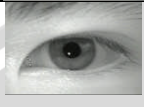









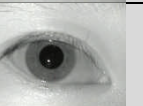
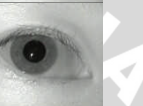



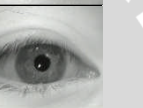





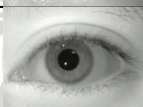

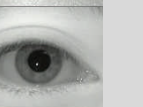
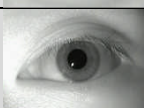
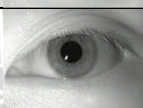
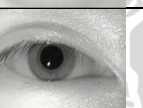



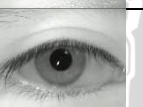





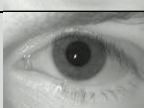
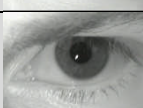

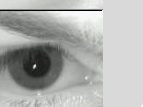
Woodwards, John. 2003. *Biometrics*. McGrawhill California.



## LAMPIRAN

### Lampiran 1. Data Latih 20 Citra

N o	Nama	Citra ke-1	Citra ke-2	Citra ke-3	Citra ke-4
1	Aeva				
2	Bryan				
3	Ching				
4	Hock				
5	Chual				
6	Christine				
7	Fiona				
8	Zulaikha				
9	Nkl				

10	Liujw				
11	Pcl				
12	Lpj				
13	Mahsk				
14	Mimi				
15	Ngkok				
16	NurSuha				
17	Ong				
18	Siti				
19	Sarina				
20	Thomas				

## BIODATA

1. Nama Lengkap : Nur Maulidiyah
2. Tempat, tanggal lahir : Malang, 7 Agustus 1987
3. Nomor Induk Mahasiswa : 0510963043
4. Lulus SD : MI Islamiyah
5. Lulus SLTP : MTSN Malang I
6. Lulus SLTA : SMU Negeri 3 Malang
7. Jenis Kelamin : Perempuan
8. Agama : Islam
9. Status Kawin / Belum Kawin : Kawin
10. Alamat asal : Jl.M.Panjaitan 17 A no.92 B
11. No. Telpn Rumah /HP : 085646344204
12. Alamat di Malang : Jl .M.Panjaitan 17 A no.92 B
13. Nama Orang Tua : Yusuf M
14. Pekerjaan Orang Tua : Wirausaha
15. Alamat Orang Tua : Jl. Kebonsari 27 Malang
16. Tanggal Mulai Tugas Akhir : 4 April 2009
17. Dosen pembimbing I : Drs.Mardji,MT
18. Dosen pembimbing II : Candra Dewi, S.kom., M.Sc
19. Lama Studi : 4 Tahun 5 Bulan
20. Judul Skripsi : Pengenalan iris mata dengan menggunakan transformasi *haar wavelet* dan *hamming distance*
21. Hobby : Membaca, Programming
22. Indeks Prestasi Kumulatif : 3,2
23. Harapan setelah selesai studi : S2



UNIVERSITAS BRAWIJAYA

