

**IDENTIFIKASI IRIS MATA MANUSIA  
MENGUNAKAN METODE *ADAPTIVE  
NEURO-FUZZY INFERENCE SYSTEM***

**TUGAS AKHIR**

Oleh

**DWI KRIS SETIANTO**

**06 06 042 481**



**DEPARTEMEN TEKNIK ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
GENAP 2007/2008**

**IDENTIFIKASI IRIS MATA MANUSIA  
MENGUNAKAN METODE *ADAPTIVE  
NEURO-FUZZY INFERENCE SYSTEM***

**TUGAS AKHIR**

Oleh

**DWI KRIS SETIANTO**

**06 06 042 481**



**TUGAS AKHIR INI DIAJUKAN UNTUK MELENGKAPI  
SEBAGIANPERSYARATAN MENJADI SARJANA TEKNIK**

**DEPARTEMEN TEKNIK ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
GENAP 2007/2008**

## PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa tugas akhir dengan judul :

### **IDENTIFIKASI IRIS MATA MANUSIA MENGUNAKAN METODE *ADAPTIVE* *NEURO-FUZZY INFERENCE SYSTEM***

yang dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada program studi Teknik Elektro, Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari tugas akhir yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau Instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Depok, 14 Juli 2008

Dwi Kris Setianto

NPM 0606042481

## PENGESAHAN

Tugas akhir dengan judul :

### **IDENTIFIKASI IRIS MATA MANUSIA MENGUNAKAN METODE *ADAPTIVE* *NEURO-FUZZY INFERENCE SYSTEM***

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada program studi Teknik Elektro, Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia. Tugas akhir ini telah diujikan pada sidang ujian tugas akhir pada tanggal 10 Juli 2008 dan dinyatakan memenuhi syarat/sah sebagai tugas akhir pada Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia.

Depok, 14 Juli 2008

Dosen Pembimbing

Ir. Aries Subiantoro, M.SEE

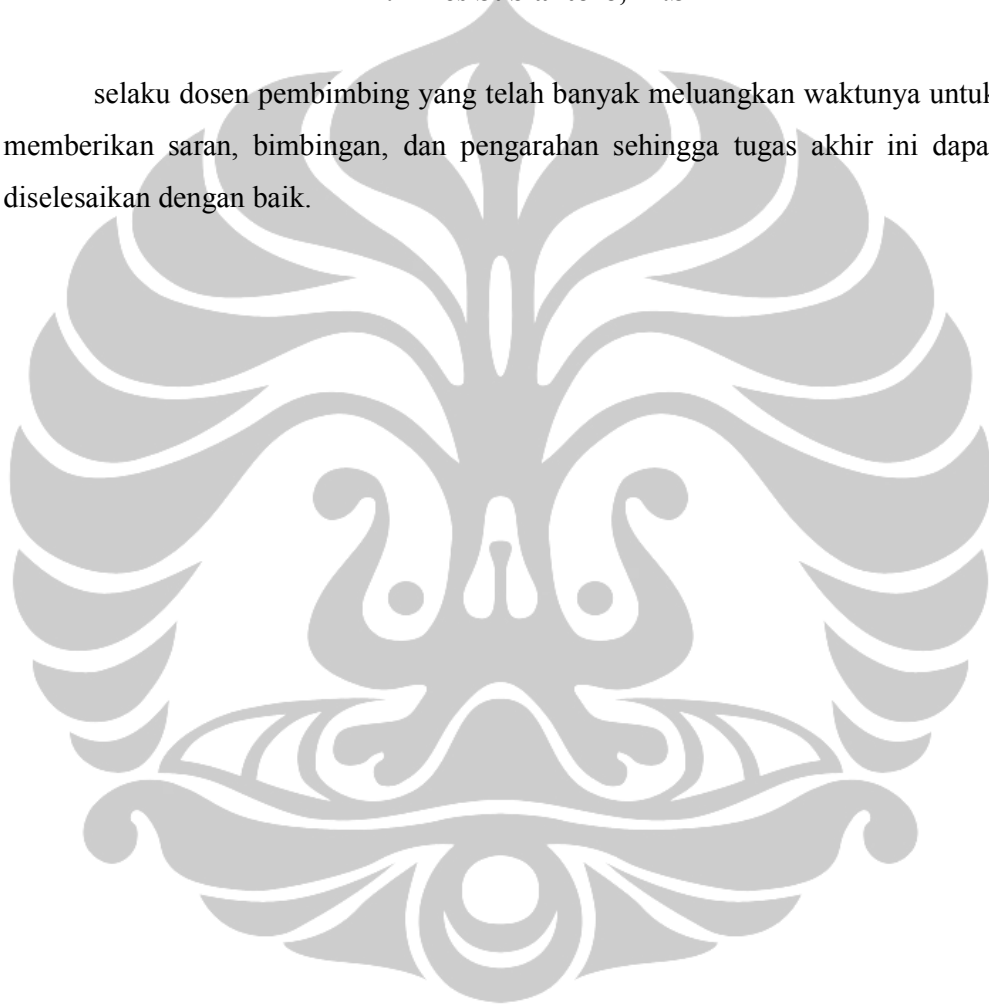
NIP 132137887

## **UCAPAN TERIMA KASIH**

Puji syukur kepada Allah SWT atas segala rahmat dan karunia-Nya sehingga tugas akhir ini dapat diselesaikan. Shalawat dan salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW. Penulis ingin mengucapkan terima kasih kepada Bapak:

**Ir. Aries Subianto, M.SEE**

selaku dosen pembimbing yang telah banyak meluangkan waktunya untuk memberikan saran, bimbingan, dan pengarahan sehingga tugas akhir ini dapat diselesaikan dengan baik.



**IDENTIFIKASI IRIS MATA MANUSIA MENGGUNAKAN METODE  
*ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM***

**ABSTRAK**

Tugas akhir ini dilakukan sebagai penelitian untuk menganalisa proses pengenalan iris mata manusia dengan teknik pengolahan citra menggunakan metode *adaptive neuro-fuzzy inference system* (ANFIS). Hal ini didasarkan pada teori bahwa setiap iris mata manusia mempunyai bagian-bagian yang unik dan berbeda antara iris yang satu dengan yang lain. Data iris yang digunakan dalam penulisan tugas akhir ini diambil dari <http://pesona.mmu.edu.my/~ccteo/>. Data yang didapat sudah dalam *grayscale* dengan demikian dari data tersebut sudah dapat diolah dengan pengolahan selanjutnya dengan menentukan *region of interest*, mengubah koordinat citra dari koordinat kartesian menjadi koordinat polar, mengekstrak citra menjadi 3 bagian dan membuat blok-blok dari matriks tersebut, kemudian mengkodekan dengan filter *wavelet*. Dari hasil pengkodean ini didapat tiga buah parameter yaitu matriks diagonal, matriks vertikal, dan matriks horisontal. Dari ketiga parameter ini diambil nilai rata-ratanya dan hasilnya digunakan untuk pembentukan database yang kemudian digunakan untuk proses pelatihan, pengujian, dan pengecekan pada *adaptive neuro-fuzzy inference system* dengan menggunakan fungsi keanggotaan gaussian dan metode *subtrctive clustering*. Hasil yang diadapat dari metode ini untuk citra mata yang telah dilatih sebesar 90% sedangkan untuk data uji atau citra yang tidak dilatih sebesar 25%.

**Kata Kunci : Citra Biometrik, Fuzzy Logic, ANFIS, Identifikasi Iris, Polar**

Dwi Kris Setianto  
NPM 06 06 04 2481  
Electrical Engineering Department

Dosen Pembimbing  
Aries Subiantoro, ST, M.SEE

**IDENTIFIKASI IRIS MATA MANUSIA MENGGUNAKAN METODE  
*ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM***

***ABSTRACT***

This paper was made for studying identification process of human iris with image processing using adaptive neuro-fuzzy inference system (ANFIS) method. The study based on the theory that every human iris have unique parts. Data that used in this paper taken from <http://pesona.mmu.edu.my/~ccte/>. Data format in grayscale level therefore this data could be process with the further processing decisively region of interest, transform from rectangular coordinate to polar coordinate, extracted the image to 3 parts and made blocs from this matrix, afterwards encode the matrix using wavelet filter. From the results of this coding was gotten three parameters that is the diagonal matrix, the vertical matrix, and the horizontal matrix. From the three parameters was taken average value and results was used for database formation, afterwards was used for training process, testing, and checking in adaptive neuro-fuzzy inference system used the function of the gaussian membership and subtractive clustering method. The result for his method 90% for image that was trained and 25% for test data or image that was not trained.

**Key Word : Biometric, Fuzzy Logic, ANFIS, Iris Identification, Polar**

# DAFTAR ISI

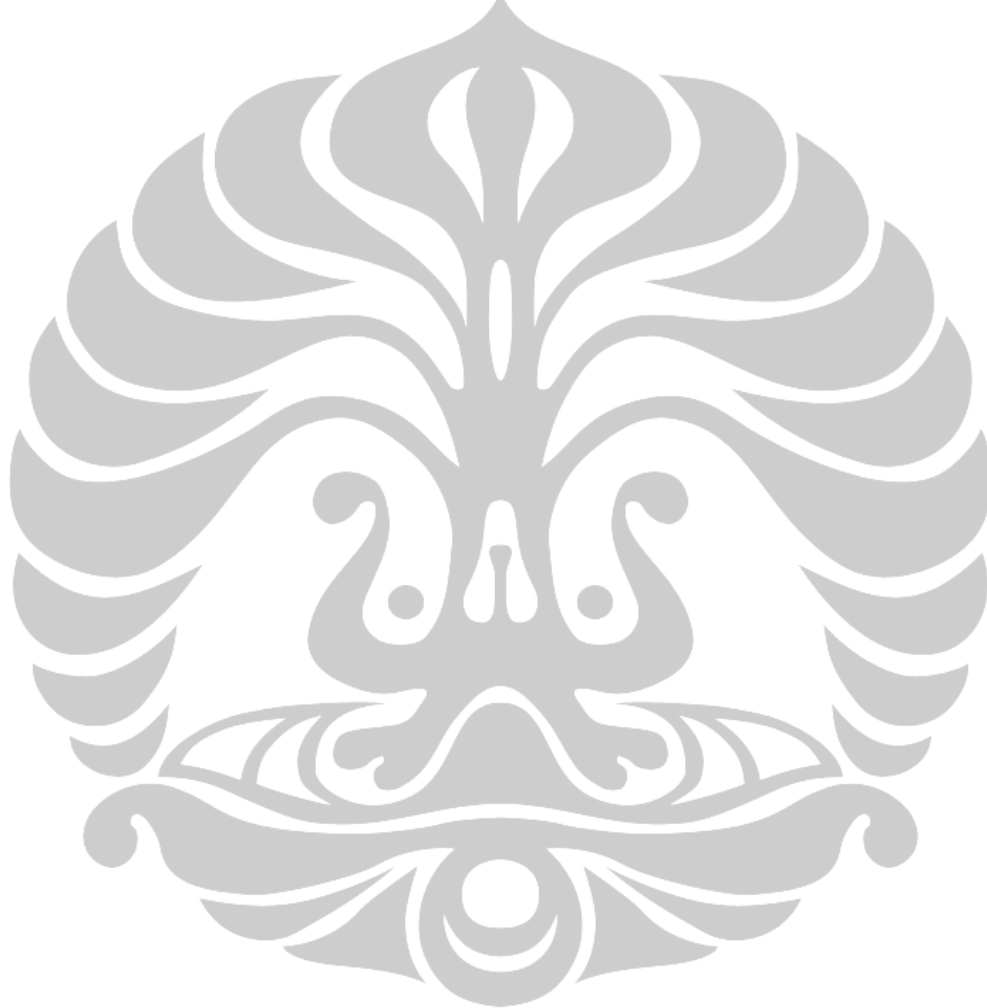
	Halaman
PERNYATAAN KEASLIAN SRIPSI	ii
LEMBAR PENGESAHAN	iii
UCAPAN TERIMA KASIH	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
<b>BAB I PENDAHULUAN</b>	<b>1</b>
1.1    LATAR BELAKANG	1
1.2    TUJUAN PENULISAN	3
1.3    PEMBATASAN MASALAH	3
1.4    METODOLOGI PENULISAN	3
<b>BAB II DASAR TEORI</b>	<b>6</b>
2.1    CITRA DIGITAL	6
2.1.1    Sampling dan Graylevel Quantization	8
2.2    IMAGE RECOGNITION	9
2.2.1    Pemisahan Objek Iris dari Citra Mata	10
2.2.2    Binerisasi	10
2.2.3    Operasi Morphologi	10
2.2.4    Rekonstruksi Look Up	11
2.2.5    Transformasi Koordinat Polar	11
2.2.6    Perhitungan Parameter Koordinat Polar	11
2.2.7    Perhitungan Titik Pusat	12
2.2.8    Perhitungan Jari-Jari Lingkaran	12
2.2.9    Pembentukan Citra Polar	13



2.3	LOGIKA FUZZY	13
2.3.1.	Fuzzy Set	14
2.3.2.	Fuzzy Inference System	16
2.3.3	Fungsi Keanggotaan	18
2.3.4	Fuzzyfikasi	20
2.3.5	If then Rule	20
2.3.6	ANFIS	20
<b>BAB III ALGORITMA PERANCANGAN SISTEM</b>		
	IDENTIFIKASI IRIS DENGAN <i>ADAPTIVE NEURO-FUZZY</i>	
	<i>INFERENCE SYSTEM</i>	25
3.1	PROSEDUR IDENTIFIKASI	25
3.2	Algoritma Pemisahan Bagian Iris Mata/ROI	28
3.3	Binerisasi	28
3.4	Perataan Ilminasi	28
3.5	Operasi Morphology	28
3.6	Pencarian Titik Batas	29
3.7	Rekonstruksi Lokup	30
3.8	Algoritma Pemrosesan Awal/Preprocessing	30
3.9	Algoritma Transformasi Polar	30
3.10	Pembentukan Iris Code dengan dwt transform	32
3.11	Algoritma Matching degan ANFIS	33
<b>BAB IV HASIL SIMULASI DAN ANALISA</b>		25
4.1	PROSEDUR SIMULASI	25
4.2	ANALISA DATA	28
<b>BAB V KESIMPULAN</b>		37
	DAFTAR ACUAN	38
	DAFTAR PUSTAKA	39

## DAFTAR TABEL

	Halaman
<b>Tabel 3.1.</b> Parameter dari Tiap-Tiap iris	37
<b>Tabel 4.1</b> Hasil Uji Coba Untuk Semua Data Training dan Data Uji	44
<b>Tabel 4.2</b> Hasil Uji untuk Data yang Telah di Training	44
<b>Tabel 4.3</b> Hasil Uji untuk Citra yang Tidak di Training	45

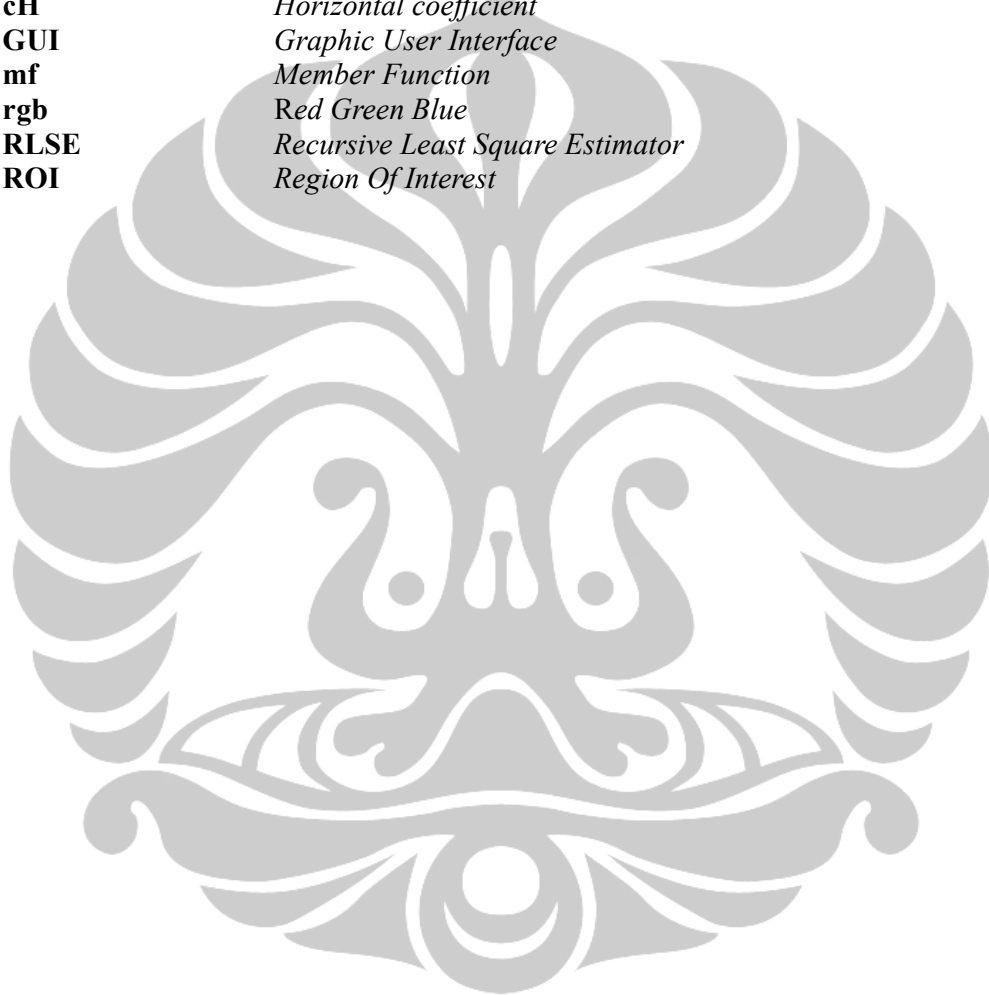


## DAFTAR GAMBAR

	Halaman
<b>Gambar 2.1</b> Ilustrasi Transformasi Koordinat Polar	13
<b>Gambar 2.2</b> Karakteristik dari Himpunan Keanggotaan Biasa	14
<b>Gambar 2.3</b> Karakteristik dari Fungsi Keanggotaan pada Fuzzy Set	15
<b>Gambar 2.4</b> Fungsi Keanggotaan pada Hari-Hari Kerja	16
<b>Gambar 2.5</b> Mekanisme Rule Sugeno	17
<b>Gambar 2.6</b> Fungsi Keanggotaan Segitiga dan Trapesium	19
<b>Gambar 2.7</b> Fungsi Keanggotaan Gaussian	19
<b>Gambar 2.8</b> Model Fuzzy Sugeno Orde Satu dengan Dua Indut	21
<b>Gambar 2.9</b> Arsitektur ANFIS	22
<b>Gambar 3.1</b> Diagram Alir Program Utama	26
<b>Gambar 3.2</b> Diagram Alir Proses Pengnalan Iris Mata	27
<b>Gambar 3.3</b> Citra Mata Asli dan Citra Maa Hasil Morphology	29
<b>Gambar 3.4</b> Citra Mata Asli dan Hasil Rekonstruksi	30
<b>Gambar 3.5</b> Citra Mata dan Hasil dalam Koordinat Polar	33
<b>Gambar 3.6</b> Citra Polar dan Iris Code	34
<b>Gambar 3.7</b> ANFIS Editor	35
<b>Gambar 3.8</b> Pembentukan database	38
<b>Gambar 3.9</b> Data Setelah di Training	40
<b>Gambar 3.10</b> Hasil Pembentukan rule base dari Database	40
<b>Gambar 3.11</b> Himpunan Keanggotaan Masukkan Pertama	41
<b>Gambar 4.1</b> Tampilan <i>Figure</i> Program	42

## DAFTAR SINGKATAN

<b>ANFIS</b>	<i>Adaptive Neuro-Fuzzy Inference System</i>
<b>bmp</b>	<i>bitmap</i>
<b>cA</b>	<i>Approximation coefficient</i>
<b>cV</b>	<i>Vertical coefficient</i>
<b>cD</b>	<i>Diagonal coefficient</i>
<b>cH</b>	<i>Horizontal coefficient</i>
<b>GUI</b>	<i>Graphic User Interface</i>
<b>mf</b>	<i>Member Function</i>
<b>rgb</b>	<i>Red Green Blue</i>
<b>RLSE</b>	<i>Recursive Least Square Estimator</i>
<b>ROI</b>	<i>Region Of Interest</i>



# **BAB I**

## **PENDAHULUAN**

### **1.1. LATAR BELAKANG**

Sistem biometrik saat ini telah mencapai perkembangan yang luar biasa dalam menggantikan sistem verifikasi konvensional. Pemanfaatan anggota tubuh secara unik untuk membedakan antara satu orang dengan orang lain, telah banyak dibuktikan memberikan hasil yang akurat dalam pengidentifikasian. Pada dasarnya pada diri setiap manusia memiliki sesuatu yang unik/khas yang hanya dimiliki oleh dirinya sendiri dan tidak akan berubah dalam waktu yang lama. Hal ini menimbulkan gagasan untuk menjadikan keunikan manusia itu sebagai identitas diri. Teknologi ini disebut sebagai biometrik. Biometrik adalah metode untuk mengidentifikasi atau mengenali seseorang berdasarkan karakteristik fisik atau perilakunya.

Bagian-bagian dari tubuh manusia yang bersifat unik/ spesifik dan juga akurat adalah :

1. Sidik jari
2. Telapak Tangan
3. Suara
4. Struktur wajah
5. Iris dan retina mata

Identifikasi personal merupakan bagian penting dari transaksi, dimana masyarakat sering menggunakan identifikasi personal yang membutuhkan sesuatu yang seseorang ketahui, seperti password, kartu, kunci dan sebagainya. Identifikasi iris merupakan metode otomatis yang memberikan identifikasi asli dari seseorang dengan mengambil gambaran iris. Sistem pembacaan iris telah banyak digunakan pada pengimplementasian kontrol akses keamanan tinggi seperti bidang militer, institusi keuangan, bahkan keamanan penjara. Selain itu, identifikasi iris memiliki tingkat penolakan kesalahan yang rendah, hampir nol persen, karena pola-pola dalam iris yang unik [1].

Identifikasi pembacaan iris bekerja berdasarkan pola-pola pada bagian iris

mata. Pengambilan iris mata dapat dilakukan dengan mudah dengan menggunakan kamera digital biasa.

Pola-pola dari iris ini dapat dipelajari oleh suatu metode logika, baik itu *Neural Network*, *Markov* ataupun *Fuzzy*. Metode neural network pada dasarnya akan mempelajari pola perubahan nilai dari suatu matriks masukan, oleh karena itu dikenal sebagai Sistem Pengenal Pola (*Pattern Recognition*). Neural network meniru suatu jaringan syaraf biologi mengadopsi kemampuan sebuah jaringan syaraf biologi untuk mengenali suatu objek walaupun objek tersebut mengalami perubahan bentuk selama perubahan tersebut tidak terlalu signifikan. Metode ini juga harus dilatih terlebih dahulu, pelatihan akan membuat system tersebut mengingat dan mengasosiasikan satu pola perubahan nilai dari suatu matriks masukan ke satu nilai keluaran.

Metode Markov merupakan contoh matematis dari proses stokastik, yaitu proses yang menghasilkan urutan acak dari hasil keluaran yang berdasar pada kemungkinan tertentu. Sedangkan metode logika fuzzy, mampu membahasakan variabel sistem dan memberikan rentang nilai kebenaran yang kontinu dalam interval nol sampai dengan satu, berbeda dengan sistem biner yang memberikan keputusan nilai kebenaran yang kaku (benar atau salah) nol atau satu. Logika fuzzy dapat mengenali citra dengan parameter karakteristik citra yang mirip karena kemampuannya menggambarkan parameter karakteristik citra dalam kurva fungsi keanggotaan yang fleksibel. Semakin banyak parameter karakteristik citra yang digunakan semakin besar pula kemungkinan citra dapat dikenali. Bila dibandingkan dengan metode lainnya, metode fuzzy dipilih karena lebih mudah dimengerti, fleksibel, serta toleran terhadap data yang kurang tepat [2]. Selain itu metode fuzzy juga mampu memeragakan fungsi nonlinear dari kerumitan yang berubah-ubah, dan dapat dipadukan dengan teknik control konvensional.

Logika fuzzy dapat digunakan sebagai pengatur pada alat-alat rumah tangga seperti mesin cuci (yang mensensor beban dan konsentrasi deterjen serta mengatur perputaran pencucian) dan lemari es, Contoh lain dimana logika fuzzy digunakan yaitu pada *automobile* dan subsistem kendaraan lainnya, *air conditioner*, kamera, *digital image processing*, *elevator*, mesin pencuci piring, *pattern recognition* pada remote sensing, *microcontroller*, dan *microprosesor*.

Selain metode yang telah disebutkan diatas ada beberapa metode yang telah banyak digunakan seperti menggunakan *hamming distance* yang dilakukan oleh Libor Masek dari *School of Computer Science and Software Engineering, The University of Western Australia* tahun 2003, Mehmet SARIOĞLU dan Yusuf Arif ERDEM dari *Electrical and Electronic Engineering Department Eastern Mediterranean University* tahun 2005 menggunakan *eigenface* yaitu dengan menggunakan perbandingan rata-rata database dan rata-rata dari masing-masing citra. P. Gupta, H. Mehrotra, A. Rattani, A. Chatterjee, A.K. Kaushik dari *Department of Information Technology, New Delhi Indian Institute of Technology Kanpur, India* menggunakan *corner detection*.

## **1.2 TUJUAN PENULISAN**

Penulisan tugas akhir ini bertujuan untuk mengaplikasikan metode Logika Fuzzy (*Fuzzy Logic*) pada sistem identifikasi citra biometrik iris dengan memanfaatkan teknik pengolahan citra.

## **1.3 PEMBATAAN MASALAH**

Pokok bahasan pada tugas akhir ini dibatasi pada pengidentifikasian iris mata manusia menggunakan metode Logika Fuzzy dengan memanfaatkan teknik pengolahan citra. Data mata yang digunakan dalam penulisan tugas akhir ini diambil dari <http://pesona.mmu.edu.my/~ccte/>.

## **1.4 METODOLOGI PENYELESAIAN MASALAH**

Metodologi pembahasan yang digunakan dalam penulisan tugas akhir ini adalah:

1. Studi pustaka :
  - a) Pencarian referensi

Mencari referensi yang berhubungan dengan pengolahan citra digital, *fuzzy logic*, data iris dan hal-hal lain yang berkaitan dengan judul pada tugas akhir ini.

b) Pendalaman materi

Mempelajari dan memahami materi yang berhubungan dengan tugas akhir ini, seperti melakukan observasi lewat internet untuk mencari pengetahuan tambahan yang tidak terdapat pada buku-buku literatur serta melakukan konsultasi kepada Pembimbing Tugas Akhir maupun diskusi dengan teman-teman.

2. Perancangan Perangkat Lunak.

Perancangan Perangkat Lunak dengan menggunakan konsep analisis dan desain yang terstruktur, dan dimodelkan menggunakan diagram proses.

3. Implementasi.

Implementasi secara coding berdasarkan analisis dan desain yang telah dibuat dan perangkat lunak itu sendiri akan dibuat dengan menggunakan bahasa pemrograman perangkat lunak komputasi numerik.

4. Analisis fungsi hasil implementasi.

Aplikasi yang telah selesai diimplementasikan akan dievaluasi. Akan dilakukan pengujian dari sistem yang telah dibangun pada tahap implementasi kemudian menganalisa tingkat kesalahan dari proses pengenalan iris manusia yang diujikan, serta akan dilakukan koreksi dan penyempurnaan program apabila diperlukan.

5. Penyusunan laporan tugas akhir dan kesimpulan akhir.

## 1.5 SISTEMATIKA PENULISAN

Sistematika penulisan pada tugas akhir ini adalah sebagai berikut :

### **BAB I PENDAHULUAN**

Bab ini menjelaskan latar belakang, tujuan penulisan, pembatasan masalah dan sistematika penulisan.

### **BAB II DASAR TEORI**

Bab ini menjelaskan teori-teori pendukung yaitu pengetahuan dasar tentang biometrik, prinsip-prinsip dasar retina, prinsip-prinsip dasar logika fuzzy secara umum yang diaplikasikan dalam suatu Sistem Identifikasi Citra Biometrik Iris, serta teknik pengolahan



citra.

**BAB III PERANCANGAN SISTEM**

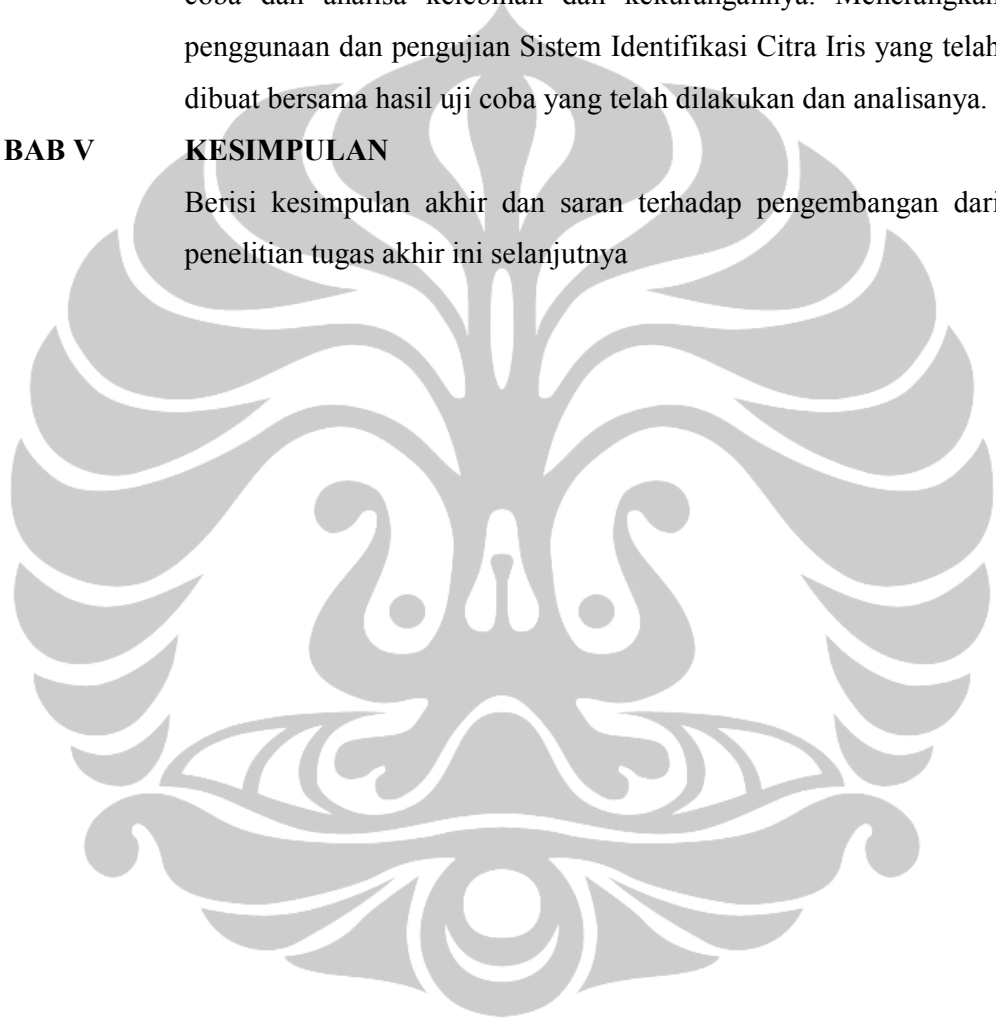
Bab ini membahas tentang analisis dan perancangan awal Sistem Identifikasi Citra Iris.

**BAB IV UJI COBA DAN ANALISA SISTEM**

Bab ini membahas pengujian perangkat lunak/simulasi, hasil uji coba dan analisa kelebihan dan kekurangannya. Menerangkan penggunaan dan pengujian Sistem Identifikasi Citra Iris yang telah dibuat bersama hasil uji coba yang telah dilakukan dan analisisnya.

**BAB V KESIMPULAN**

Berisi kesimpulan akhir dan saran terhadap pengembangan dari penelitian tugas akhir ini selanjutnya



## BAB II

### DASAR TEORI

#### 2.1 CITRA DIGITAL

Perkembangan citra digital dimulai pada tahun 1921, yaitu pertama kalinya sebuah foto berhasil ditransmisikan secara digital melalui kable laut dari New York ke London (Bartlane Cable Picture Transmission System). Keuntungan yang dirasakan pada saat itu adalah pengurangan waktu pengiriman foto dari sekitar 1 minggu menjadi kurang dari 3 jam. Foto tersebut dikirim dalam bentuk kode digital dan kemudian diubah kembali oleh printer relegraph.

Contoh aplikasi dalam berbagai disiplin ilmu:

- a. Bidang kedokteran  
Sistem mendeteksi diagnosis suatu kelainan dalam tubuh manusia melalui citra yang dihasilkan oleh *scanner*.
- b. Bidang industri  
Sistem pemeriksaan suatu produk melalui kamera video.
- c. Bidang perdagangan  
Sistem untuk mengenali angka/huruf dalam suatu formulir secara otomatis dalam mesin pembaca.
- d. Bidang militer  
Sistem pengenalan target peluru kendali melalui sensor visual.

Istilah “citra” yang digunakan dalam pengolahan citra dapat diartikan sebagai suatu fungsi kontinu dari intensitas cahaya dalam bidang dua dimensi. Pemrosesan citra dengan komputer digital membutuhkan citra digital sebagai masukannya. Citra digital adalah citra kontinu yang diubah dalam bentuk diskrit, baik koordinat ruang maupun intensitas cahayanya. Pengolahan digitalisasi terdiri dari dua proses yaitu pencuplikan (*sampling*) posisi, dan kuantisasi intensitas. Citra digital dapat dinyatakan dalam matriks dua dimensi  $f(x,y)$  dimana ‘ $x$ ’ dan ‘ $y$ ’ merupakan koordinat piksel dalam matriks dan ‘ $f$ ’ merupakan derajat intensitas

tersebut. Citra digital berbentuk matriks dengan ukuran  $M \times N$ , susunannya sebagai berikut:

$$f(x,y) = \left\{ \begin{array}{cccccc} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,N-1) \\ f(2,0) & f(2,1) & f(2,2) & \dots & f(2,N-1) \\ \dots & \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \dots & f(M-1,N-1) \end{array} \right\} \dots\dots(2.1)$$

Suatu citra  $f(x,y)$  dalam fungsi matematis dapat dituliskan sebagai berikut:

Dimana:  $M$  = banyaknya baris pada array citra  
 $N$  = banyaknya kolom pada array citra  
 $G$  = banyaknya skala keabuan (*graylevel*)

Interval  $(0, G)$  disebut skala keabuan (*grayscale*). Besar  $G$  tergantung pada proses digitalisasinya. Biasanya keabuan 0 (nol) menyatakan intensitas hitam dan  $G$  menyatakan intensitas putih. Untuk cira 8 bit, nilai  $G$  sama dengan  $2^8 = 256$  warna (derajat keabuan).

Jika kita memperhatikan citra digital secara seksama, kita akan melihat titik-titik kecil berbentuk segiempat yang memebentuk citra tersebut. Titik-titik tersebut merupakan satuan terkecil dari suatu citra digital disebut sebagai "*picture element*", "*pixel*", piksel, atau "*pel*". Jumlah piksel per satuan panjang akan menentukan resolusi citra tersebut. Makin banyak piksel yang mewakili suatu citra, maka makin tinggi nilai resolusinya dan makin halus gambarnya. Pada sistem dengan tampilan citra digital yang dirancang dengan baik (beresolusi tinggi), titik - titik kecil tersebut tidak teramati oleh kita yang melihat secara normal.

Untuk aplikasi 3 dimensi dikenal datuan *voxel* yang memiliki dimensi ketebalan dan berbentuk kubus. Beberapa sistem peralatan *medical imaging* seperti *CT-Scan* atau *ultrasonography* menggunakan bentuk citra digital 3 dimensi sebagai keakuratan pengamatan. Citra bisa digolongkan berdasarkan terdefinisi atau tidaknya citra tersebut pada setiap titik spasial  $(x,y)$  dan terhingga atau tidak

terhingga-nya nilai kecerahan citra. Berdasarkan penggolongan ini, citra dibagi menjadi kontinu-kontinu, kontinu-diskrit, diskrit-kontinu, dan diskrit-diskrit [2].

Citra kontinu-kontinu adalah citra dengan presisi yang tak terhingga, terdefinisi pada setiap titik spasial. Citra ini sering disebut citra analog atau citra kontinu. Citra ini dapat di representasikan dengan presisi yang terhingga untuk menghasilkan citra kontinu diskrit. Citra jenis ini dihasilkan dari proses kuantisasi yang memetakan nilai kecerahan citra yang sebenarnya (tak berhingga) menuju nilai kecerahan yang berhingga yang mampu diakomodasi oleh proses komputer. Citra juga bisa tetap memiliki tingkat kecerahan yang tak berhingga namun hanya terdefinisi pada kumpulan titik spasial yang terbatas. Citra jenis ini dihasilkan dari proses sampling yang menghasilkan citra diskrit kontinu. Karena komputer digital beroperasi pada presisi yang terbatas, maka komputer digital cocok untuk citra diskrit-diskrit. Pada citra jenis ini koordinat spasial dan tingkat kecerahan dikuantisasi menuju presisi numerik yang mampu ditangani komputer digital. Citra jenis ini biasa disebut citra digital atau citra diskrit [2].

Citra monokrom atau citra hitam putih ialah citra yang memiliki nilai  $f$  untuk satu warna yang merepresentasikan *graylevel*. Selain itu ada pula citra berwarna dimana nilai  $f$  merepresentasikan tiga komponen warna pada setiap titik pada koordinat spasial. Tiga komponen warna tersebut adalah merah hijau dan biru (RGB). Sehingga untuk citra berwarna  $f$  menjadi [2] :

$$f(x, y) = \{f_{merah}(x, y), f_{hijau}(x, y), f_{biru}(x, y)\} \dots\dots\dots (2.2)$$

Citra berwarna tersebut bisa diinterpretasikan sebagai tumpukan (*stack*) dari tiga citra. Tiga komponen warna tersebut masing-masing disebut kanal (*channel*). Dengan demikian citra monokrom hanya memiliki satu kanal, sedangkan citra RGB memiliki tiga kanal.

### 2.1.1 SAMPLING DAN GRAYLEVEL QUANTIZATION

Untuk prosesi komputer, citra kontinu harus diubah menjadi citra diskrit dengan melakukan dua proses yaitu sampling dan *graylevel quantization*. Citra kontinu yang di-sampling dan dikuantisasi akan menghasilkan  $N \times M$  array

dimana setiap elemen array (disebut *picture element / pixel*) bersifat diskrit dan merupakan representasi dari tingkat kecerahan (*graylevel*) citra pada titik koordinat spasialnya.

Tingkat *graylevel* yang ada menentukan storage bits dengan perhitungan suatu citra yang disampling menjadi  $N \times M$  array [2] :

$$N = 2^n, M = 2^k, \text{Maka } G = 2^m \dots\dots\dots(2.3)$$

Dengan  $G$  merupakan nilai *graylevel*. Maka storage bit yang dibutuhkan adalah  $B = N \times M \times m$  Contohnya *storage bit* yang dibutuhkan untuk gambar dengan ukuran piksel  $256 \times 256$  dengan *graylevel* 64 membutuhkan 393.216 bit untuk penyimpanan.

## 2.2 IMAGE RECOGNITION

Pengolahan citra (*image processing*) merupakan proses mengolah piksel-piksel dalam citra digital untuk suatu tujuan tertentu. Beberapa alasan dilakukannya pengolahan citra pada citra digital antara lain yaitu:

1. Untuk mendapatkan citra asli dari suatu citra yang sudah buruk karena pengaruh derau. Proses pengolahan bertujuan mendapatkan citra yang diperkirakan mendekati citra sesungguhnya.
2. Untuk memperoleh citra dengan karakteristik tertentu dan cocok secara visual yang dibutuhkan untuk tahap yang lebih lanjut dalam pemrosesan analisis citra

Dalam proses akuisisi, citra yang akan diolah ditransformasikan dalam suatu representasi numerik. Pada proses selanjutnya representasi numerik tersebutlah yang akan diolah secara digital oleh komputer.

Pengolahan citra pada umumnya dapat dikelompokkan dalam dua jenis kegiatan, yaitu:

1. Memperbaiki kualitas citra sesuai kebutuhan
2. Mengolah informasi yang terdapat pada citra

Bidang aplikasi yang kedua ini sangat erat kaitannya dengan *computer aided analysis* yang umumnya bertujuan untuk mengolah suatu objek citra dengan cara

mengekstraksi informasi penting yang terdapat di dalamnya. Dari informasi tersebut dapat dilakukan proses analisis dan klasifikasi secara cepat memanfaatkan algoritma perhitungan komputer.

Dari pengolahan citra diharapkan terbentuk suatu sistem yang dapat memproses citra masukan hingga citra tersebut dapat dikenali cirinya. Pengenalan ciri inilah yang sering diaplikasikan dalam kehidupan sehari-hari. Aplikasi yang dibahas pada laporan ini adalah aplikasi di bidang kedokteran, yaitu untuk aplikasi analisis. Dalam pengolahan citra digital terdapat lima proses secara umum [1], yaitu:

1. image restoration
2. image enhancement
3. image data compaction
4. image analysis
5. image reconstruction

### **2.2.1 Pemisahan Objek Iris dari Citra**

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. Metode yang dikembangkan untuk pemisahan ROI iris mata ini dilakukan dengan 3 tahap : binerisasi, operasi morfologi dan rekonstruksi *look up*.

### **2.2.2 Binerisasi**

Dalam hal ini citra mata akan diubah dari citra warna menjadi citra grayscale lalu menjadi citra biner dengan threshold nilai derajat keabuan 128. Hasil dari proses ini adalah bentuk iris mata yang berwarna hitam (*pixel 0*) dan selainnya berwarna putih (*pixel 1*). Proses ini bisa disebut juga tahap segmentasi yang memisahkan ROI iris yang diinginkan dengan bagian lain mata yang tidak dibutuhkan secara global.

### **2.2.3 Operasi *Morphology***

Bentuk iris pada mata kadangkala terganggu oleh kehadiran kilatan blitz atau pantulan cahaya yang akan menimbulkan bercak putih (*glare*) di dalam bagian iris. Dalam citra biner hal ini dapat dilihat dari kehadiran lubang putih pada bagian iris yang berwarna hitam. Untuk menghilangkan noise seperti ini dapat dilakukan operasi morfologi untuk menutup lubang glare yang terjadi. Proses yang dilakukan adalah proses *closing*, yaitu dengan melakukan operasi dilasi dan erosi secara berurutan.

### **2.2.4 Rekonstruksi *Look Up***

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.

### **2.2.5 Transformasi Koordinat Polar**

Bentuk iris yang berupa lingkaran akan sangat menyulitkan untuk dianalisis dan diolah lebih lanjut. Pola susunan piksel yang dianalisa harus mengikuti algoritma tertentu yang memungkinkan pengambilan piksel dengan bentuk geometri lingkaran. Hal ini akan sangat merepotkan dan tidak efisien. Untuk mengatasi hal tersebut kita harus terlebih dahulu mengubah citra iris ke dalam bentuk antara yang sesuai. Pengubahan bentuk ini dapat dilakukan dengan melakukan transformasi koordinat polar dari citra iris. Untuk dapat melakukan transformasi tersebut, pertama harus dilakukan deteksi tepian, perhitungan parameter koordinat polar, baru kemudian dilakukan pembentukan citra koordinat polar itu sendiri.

### **2.2.6 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.

### 2.2.7 Perhitungan Titik Pusat

Kita dapat menggunakan rumus dasar lingkaran yang telah diketahui untuk menghitung titik pusat dari lingkaran.

$$R^2 = (x - a)^2 + (y - b)^2 \dots\dots\dots(2.4)$$

$$R^2 = (x^2 - a^2) - (2x)a - (2y)b + a^2 + b^2 \dots\dots\dots(2.5)$$

dengan  $(a, b)$  adalah titik pusat dan  $R$  merupakan jari-jari.

Dari persamaan (2.5) untuk mencari 2 buah variabel absis dan ordinat titik pusat dibutuhkan sedikitnya minimal 3 buah persamaan yang berasal dari 3 titik sampel. Oleh karena itu pemilihan sampel harus lebih dari 3 buah titik pada tepian lingkaran. Jumlah perhitungan yang akan dilakukan merupakan hasil kombinasi semua titik sampel pada tiap perhitungan 3 titik yang dibutuhkan. Jumlah perhitungan yang dilakukan adalah sebanyak  ${}_3C_N$  proses dengan  $N$  adalah banyaknya titik sampel yang diambil.

### 2.2.8 Perhitungan Jari-Jari Lingkaran

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

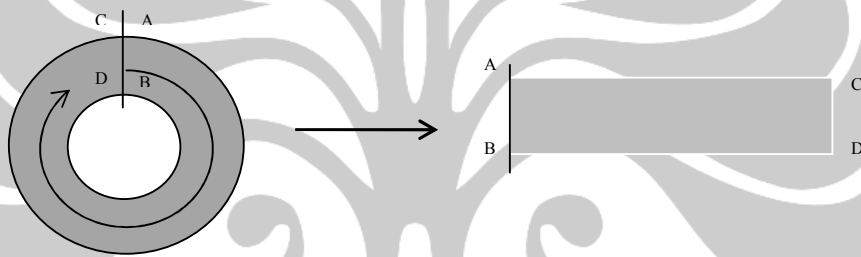
$$R_{nn}^2 = ((x_n^2 + y_n^2) - (2x_n)a - (2y_n)b + a^2 + b^2) \quad n = 0, 1, \dots, N \dots\dots\dots(2.6)$$



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

### 2.2.9 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. Proses transformasi dapat diilustrasikan sebagai berikut :



Gambar 2.1. Ilustrasi Transformasi Koordinat Polar.

### 2.3. LOGIKA FUZZY

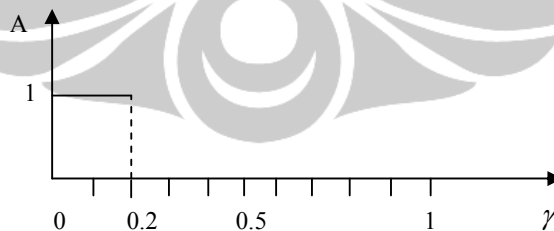
Logika Fuzzy pertama kali diusulkan oleh Lotfi A. Zadeh, profesor ilmu komputer dari University of California Berkeley pada tahun 1965. Pada dasarnya logika fuzzy adalah logika yang memberikan rentang nilai kebenaran yang kontinu dalam interval nol sampai satu, memungkinkan nilai yang intermediate (di tengah-tengah) didefinisikan diantara nilai kebenaran konvensional seperti benar atau salah, ya atau tidak, tinggi atau rendah dan lain-lain. Pernyataan seperti lebih tinggi atau sangat cepat bisa diformulasikan secara matematis dan diproses dengan komputer untuk mewujudkan cara berpikir yang lebih manusiawi pada pemrograman komputer [4]. Kelahiran logika fuzzy sendiri tidak terlepas dari semakin meningkatnya kompleksitas sistem sehingga menjadi sulit bahkan mustahil untuk membuat pernyataan yang pasti tentang *behaviour* dari suatu

sistem.

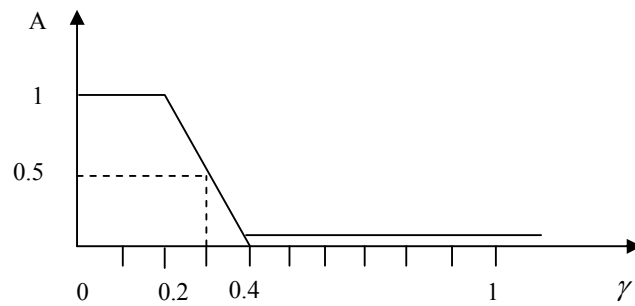
Secara formal logika fuzzy merupakan struktur pemodelan perkiraan yang bebas yang memperkirakan suatu fungsi melalui asosiasi pembahasan masukan dan keluaran. Sistem logika fuzzy melakukan pemetaan dari masukan menjadi hasil dalam aturan-aturan yang lebih mudah karena melakukan suatu aturan yang berdasarkan ada *natural language*, bahasa komunikasi sehari-hari yang pemahamannya dianggap *universal* [3].

### 2.3.1 Fuzzy Set

Konsep himpunan yang biasa (klasik atau *crisp*), keanggotaan untuk setiap elemen pada semesta himpunan didefinisikan dengan jelas antara anggota himpunan atau bukan anggota himpunan. Pada himpunan fuzzy (*fuzzy sets*) transisi keanggotaan bisa bersifat gradual. Hal ini terjadi karena pada himpunan fuzzy batas dari himpunan-himpunan fuzzy samar-samar dan ambigu. Jadi keanggotaan dari elemen-elemen pada semesta himpunan diukur dengan sebuah fungsi yang mendeskripsikan kesamar-samaran dan ambiguitas [4]. *Fuzzy sets* adalah himpunan yang beranggotakan elemen-elemen yang memiliki derajat keanggotaan yang bervariasi. *Fuzzy set* memungkinkan kondisi keanggotaan yang tidak sepenuhnya atau parsial karena keanggotaan tidak perlu penuh untuk bisa menjadi anggota *fuzzy set* yang lain pada semesta yang sama. Perbedaan antara *fuzzy set* dengan himpunan biasa dalam hal keanggotaannya bisa dilihat pada Gambar 2.4 dan Gambar 2.5 (untuk tujuan penyederhanaan semua kurva fungsi keanggotaan digambarkan kontinu) [4].



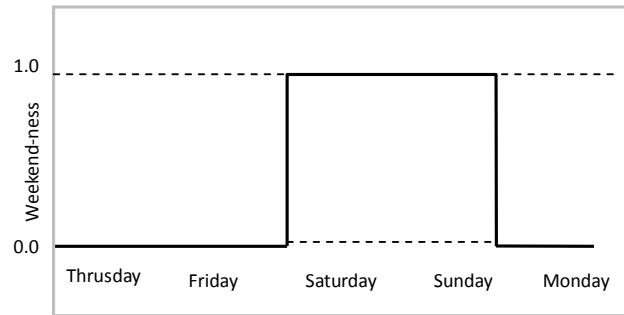
Gambar 2.4 Karakteristik dari fungsi keanggotaan pada himpunan biasa [4]



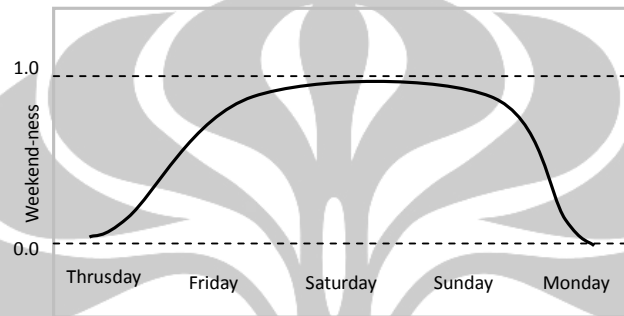
Gambar 2.3 Karakteristik dari fungsi keanggotaan pada *fuzzy sets*[4]

Dari Gambar 2.3 terlihat bahwa pada himpunan biasa (*crisp set*), elemen-elemen yang merupakan anggota dari himpunan  $A$  yaitu 0, 0,1 dan 0,2 selalu memiliki nilai keanggotaan 1. Sedangkan pada *fuzzy set* (Gambar 2.4), selain 0, 0,1 dan 0,2 yang memiliki nilai keanggotaan yang penuh (bernilai 1), himpunan  $A$  juga memiliki anggota yang nilai keanggotaannya tidak penuh yaitu 0,3.

Contoh dari *fuzzy sets* adalah hari-hari akhir pekan dan bukan akhir pekan. Gambar 2.5 adalah fungsi keanggotaan hari-hari akhir pekan untuk *crisp set* dan *fuzzy set*. Pada kurva kiri hari-hari akhir pekan digambarkan dengan fungsi keanggotaan dua nilai yaitu 0 dan 1. Dari kurva terlihat pada hari Jum'at tengah malam nilai kebenaran kurva beralih secara tidak kontinu dari nol menuju satu (bukan akhir pekan menjadi akhir pekan) [3]. Hal ini kurang sesuai dengan logika manusia. Kurva kanan menggambarkan hari-hari akhir pekan menggunakan kurva yang halus. Kurva kanan memperlihatkan bahwa hari Jum'at dan hari Kamis termasuk golongan akhir pekan dengan keanggotaan yang tidak penuh. Hal ini lebih masuk akal dari penggambaran sebelumnya.



Days of the weekend two-valued membership



Days of the weekend multivalued membership

Gambar 2.6. Fungsi keanggotaan hari-hari akhir pekan [3]

### 2.3.2 Fuzzy Inference System

*Fuzzy Inference System* adalah proses pemetaan dari suatu masukan menjadi keluaran dengan menggunakan logika fuzzy [3]. Hasil pemetaan kemudian dijadikan dasar untuk pengambilan keputusan atau untuk membedakan pola. Ada dua tipe *Fuzzy Inference System* yang bisa diimplementasikan pada *Fuzzy Logic Toolbox* yaitu tipe Mamdani dan tipe Sugeno. *Fuzzy Inference System* metode Sugeno pada dasarnya hampir sama dengan metode Mamdani. Dua bagian pertama dari proses *Fuzzy Inference* Sugeno yaitu fuzzifikasi *input* dan penggunaan operator fuzzy tepat sama dengan Mamdani. Perbedaan utama antara Mamdani dengan Sugeno adalah output fungsi keanggotaan Sugeno bersifat linier atau konstan. Rule yang sering dipakai pada model Fuzzy Sugeno berbentuk [3] :

$$\text{If Input 1} = x \text{ and Input 2} = y, \text{ then Output is } z = ax+by+c \dots \dots \dots (2.7)$$

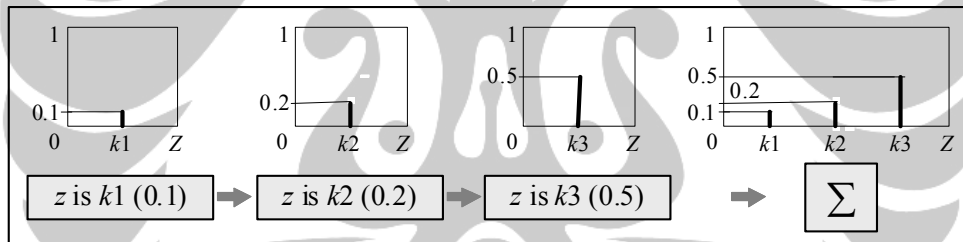
Untuk model Sugeno orde nol, level *output*  $z$  adalah konstan ( $a=b=0$ ). Level *output*  $z_i$  untuk setiap *rule* diboboti dengan *firing strength*  $w_i$  dari *rule*. Contohnya untuk sebuah *AND rule* dengan *input* 1 =  $x$  dan *input* 2 =  $y$ , nilai *firing strength* adalah [3]:

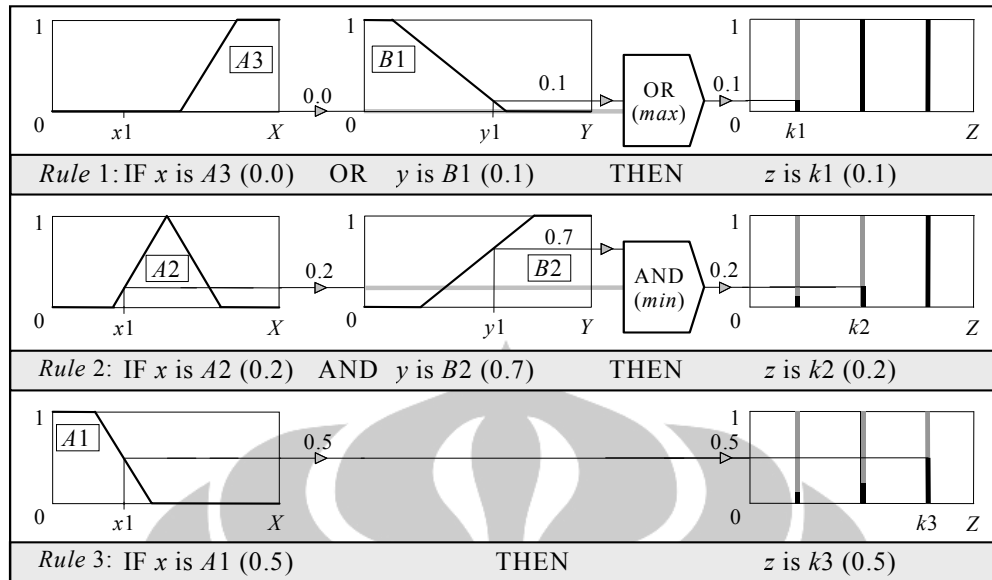
$$w_i = \text{AndMethod}(F_1(x), F_2(y)) \dots \dots \dots (2.8)$$

dimana  $F_1$  dan  $F_2$  adalah fungsi keanggotaan untuk masukan 1 dan masukan 2. Keluaran terakhir sistem adalah rata-rata terboboti dari semua aturan keluaran yang dihitung dengan persamaan (2.5) [3].

$$\text{Keluaran terakhir} = \frac{\sum_{i=1}^N w_i z_i}{\sum_{i=1}^N w_i} \dots \dots \dots (2.9)$$

Rule Sugeno bekerja seperti terlihat pada Gambar 2.7.





Gambar 2.7. Mekanisme Rule Sugeno [3]

Pada contoh *tipping* model ini, pertama-tama input yaitu *service* dan *food* dilakukan proses fuzzifikasi terlebih dahulu sehingga masukan digambarkan dalam fungsi keanggotaan. Kemudian menerapkan fuzzy operator dalam hal ini adalah operator OR. Operator OR akan memilih fungsi keanggotaan terbesar (max) untuk setiap rule. Terakhir implication method diterapkan, sehingga didapatkan nilai  $z$  untuk setiap rule [3]. Keluaran terakhir sistem adalah rata-rata terboboti dari semua rule keluaran yang dihitung dengan persamaan (2.5).

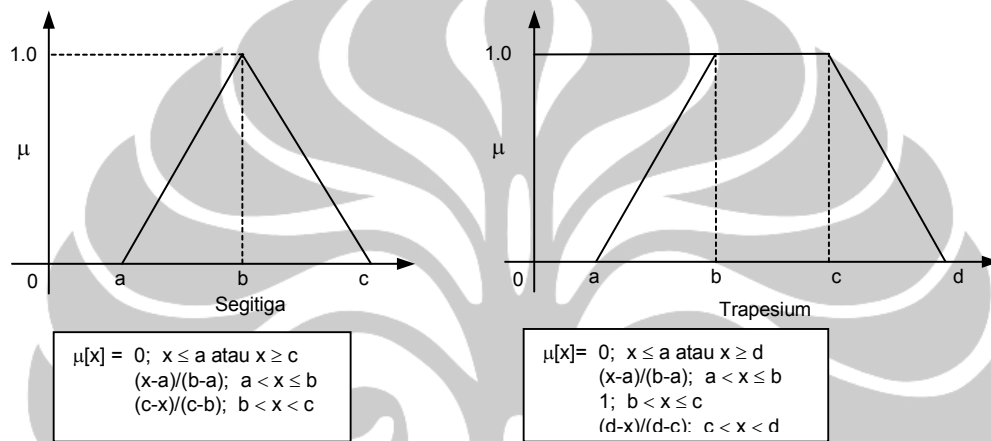
### 2.3.3 Fungsi Keanggotaan

*Fuzzy sets* selalu merupakan fungsi-fungsi yang memetakan semesta himpunan, misalnya  $X$ , menuju unit interval  $[0,1]$  dimana fuzzy set  $H$  adalah fungsi  $\mu_H$  yang memetakan  $X$  menuju  $[0,1]$ . Fungsi keanggotaan dalam fuzzy sets digambarkan dengan kurva yang mendefinisikan tiap titik pada bagian masukan yang dipetakan menjadi nilai keanggotaan atau tingkat keanggotaan antara nol dan satu.

Fungsi keanggotaan dari *fuzzy sets* memiliki tiga bagian yaitu *core*, *support*, dan *boundaries*. *Core* adalah daerah pada kurva fungsi keanggotaan yang memiliki nilai keanggotaan penuh pada *fuzzy sets*. *Support* adalah daerah pada kurva fungsi keanggotaan yang memiliki nilai keanggotaan tidak nol ( $\mu(x) > 0$ ).

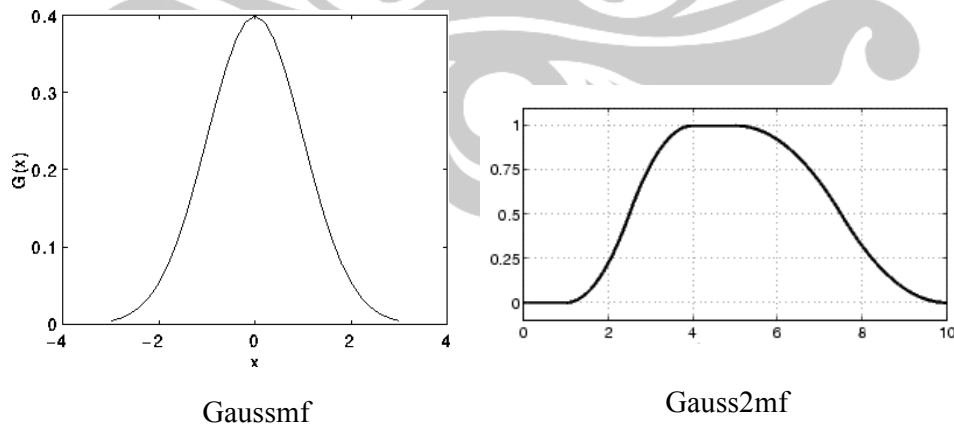
*Boundaries* adalah daerah pada kurva fungsi keanggotaan yang memiliki nilai keanggotaan antara nol dan satu ( $0 < \mu(x) < 1$ ) [5].

Ada beragam cara untuk menggambarkan fungsi keanggotaan. Cara yang paling mudah adalah dengan menggambarkan fungsi keanggotaan sebagai fungsi keanggotaan bentuk segitiga seperti pada Gambar 2.7 bagian kiri. Cara yang lain adalah dengan menggambarkannya dalam bentuk trapesium yang memiliki atap yang datar seperti pada Gambar 2.7 bagian kanan. Penggambaran fungsi keanggotaan dengan garis lurus ini memiliki keuntungan yaitu kesederhanaannya.



Gambar 2.8 Fungsi Keanggotaan Segitiga dan Trapezium

Cara lain yang lebih sulit untuk menggambarkan fungsi keanggotaan adalah dengan menggambarkannya sebagai kurva distribusi Gaussian yang memiliki bentuk kurva yang halus yang terlihat pada Gambar 2.9



Gambar 2.9 Fungsi Keanggotaan Gaussian [3]

### 2.3.4 Fuzzifikasi

Fuzzifikasi adalah proses pengubahan nilai masukan sistem yang bersifat crisp menjadi fuzzy, memiliki nilai keanggotaan yang bernilai antara nol dan satu. Disebut pula sebagai proses pembentukan fungsi keanggotaan dari masukan. Fuzzifikasi untuk mendapatkan fungsi keanggotaan dari masukan dapat dilakukan dengan berbagai cara atau metode, seperti [5] :

- a) Intuisi
- b) Inferensi
- c) *Rank ordering*
- d) *Angular fuzzy sets*
- e) Jaringan syaraf tiruan
- f) Algoritma genetika
- g) *Inductive reasoning*

### 2.3.5 If-Then Rules

Suatu masukan dari sistem fuzzy dapat dievaluasi menggunakan *if-then rules* untuk menghasilkan *output* yang diinginkan. *If-then rules* dapat merumuskan kondisi

If  $x$  is  $A$  then  $y$  is  $B$ .....(2.10)

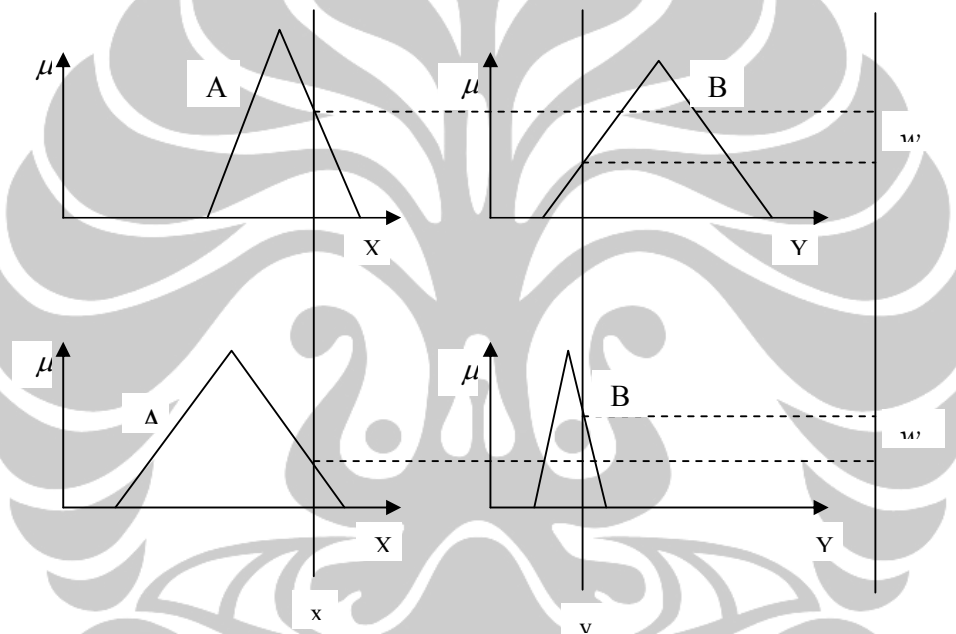
Dengan  $A$  dan  $B$  merupakan bahasa nilai yang didefinisikan menggunakan *fuzzy sets* dalam rentang (semesta pembicaraan)  $x$  dan  $y$ . Bagian *if* dalam *rule* ini ( $x$  is  $A$ ) merupakan *antecedant* atau *premise*, sedangkan bagian *then* ( $y$  is  $B$ ) disebut *consequent* atau kesimpulan [6].

### 2.3.6 Adaptive Neuro-Fuzzy inference System (ANFIS)

*Adaptive Neuro-Fuzzy Inference System* (ANFIS) adalah suatu jaringan kerja antara jaringan saraf tiruan (*neural network*) dg sistem inferensi fuzzy (FIS) model sugeno. Sistem ini disebut juga suatu kelas jaringan adaptif (*adaptive network*) dengan kemampuan pembelajaran *supervised*. sesuai dengan namanya,



jaringan adaptif adalah suatu struktur jaringan dari node-node yang berhubungan sebab akibat dalam satu jaringan (*link*) membentuk suatu unit pemroses. Seluruh atau sebagian node-node tersebut dapat menyesuaikan diri (*adaptive*), yaitu node-node keluarannya bergantung pada parameter yang dimodifikasi di tiap-tiap node. Aturan pembelajaran menunjukkan bahwa parameter-parameter tersebut dapat diperbaharui (*update*) untuk meminimalkan kesalahan (*error*) pengukuran, yang secara matematik berupa perbedaan antara keluaran jaringan yang aktual dan perkiraan keluaran. Adapun mekanisme sistem inferensi model sugeno dan arsitektur ANFIS ditunjukkan dalam gambar berikut:



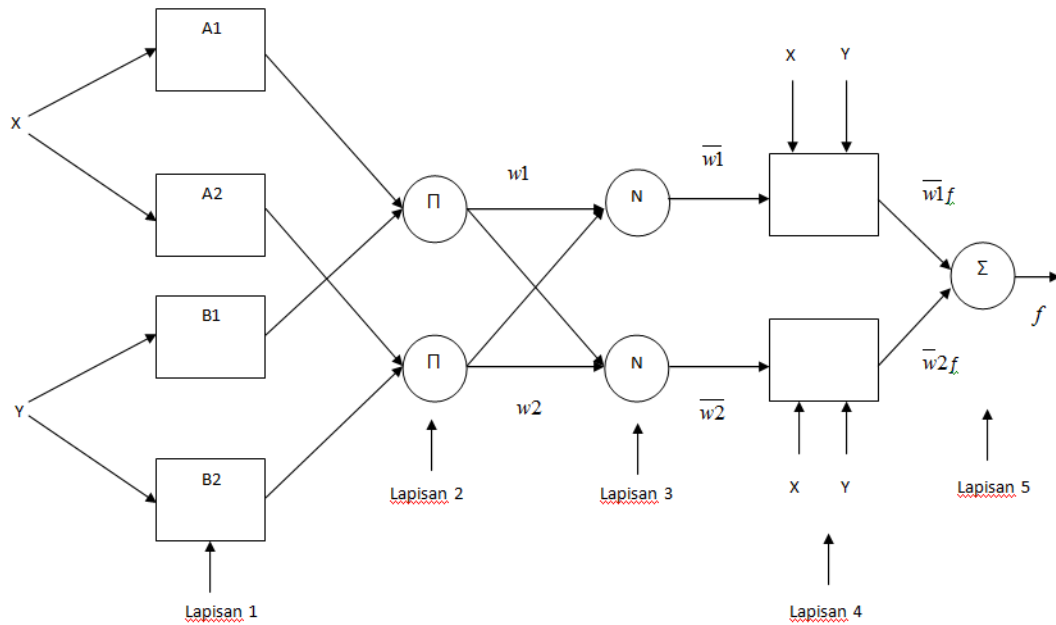
Gambar 2.10 Model Fuzzy Sugeno orde satu dengan dua buah input

Pada model fuzzy Sugeno (pada gambar 2.10) fungsi keluarannya adalah

$$\begin{aligned}
 f &= \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} \dots\dots\dots(2.11) \\
 &= \overline{w_1} f_1 + \overline{w_2} f_2
 \end{aligned}$$

dengan

$$\begin{aligned} f_1 &= p_1x + q_1y + r_1 \\ f_2 &= p_2x + q_2y + r_2 \end{aligned} \dots\dots\dots(2.12)$$



Gambar 2.11 Arsitektur ANFIS

Secara sederhana, diasumsikan bahwa sistem inferensi fuzzy model Sugeno dalam gambar 2.9 mempunyai dua masukan (*input*) x dan y satu keluaran (*output*) z. Untuk model fuzzy sugeno orde satu tersebut mempunyai aturan sebagai berikut:

- Aturan 1: *If x is A<sub>1</sub> and y is B<sub>1</sub> then f<sub>1</sub>=p<sub>1</sub>x+q<sub>1</sub>y+r<sub>1</sub>*
- Aturan 2: *If x is A<sub>2</sub> and y is B<sub>2</sub> then f<sub>2</sub>=p<sub>2</sub>x+q<sub>2</sub>y+r<sub>2</sub>*

Sedangkan pada arsitektur ANFIS yang ditunjukkan dalam gambar 2.11, node-node dari lapisan yang sama mempunyai fungsi yang sama dan keluaran (*output*) dari ode ke-i dalam lapisan l dinotasikan oleh *O<sub>l,i</sub>*. Sedangkan dalam jaringan ANFIS tersebut mempunyai lima lapisan (*layer*) yaitu:

### Layer 1

Semua simpul pada lapisan ini adalah simpul adaptif (parameter dapat berubah) dengan fungsi simpul :

$$\begin{aligned} O_{1,i} &= \mu_A(x) \text{ untuk } i = 1, 2, \text{ atau} \\ O_{2,i} &= \mu_A(x) \text{ untuk } i = 3, 4 \end{aligned} \dots\dots\dots(2.13)$$

dengan  $x$  dan  $y$  adalah masukan pada simpul  $i$ ,  $A_i$  (atau  $B_{i-2}$ ) adalah fungsi keanggotaan masing-masing simpul. Simpul  $O_{1,i}$  berfungsi untuk menyatakan derajat keanggotaan tiap masukan terhadap himpunan fuzzy  $A$  dan  $B$ . Fungsi keanggotaan yang dipakai adalah jenis *generalized bell* (*gbell*). Parameter  $a, b, c$ , pada fungsi keanggotaan *gbell* dinamakan parameter premis yang adaptif.

### Lapisan 2

Tiap keluaran simpul menyatakan derajat pengaktifan (*firing strength*) tiap aturan fuzzy. Fungsi ini dapat diperluas apabila bagian premis memiliki lebih dari dua himpunan fuzzy. Banyaknya simpul pada lapisan ini menunjukkan banyaknya aturan yang dibentuk. Fungsi perkalian yang digunakan adalah interpretasi kata *hubung and* dengan menggunakan operator *t-norm*

### Lapisan 3

Lapisan Setiap simpul pada lapisan ini adalah simpul nonadaptif yang menampilkan fungsi derajat pengaktifan ternormalisasi (*normalized firing strength*) yaitu rasio keluaran simpul ke- $i$  pada lapisan sebelumnya terhadap seluruh keluaran lapisan sebelumnya, dengan bentuk fungsi simpul:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, i = 1, 2 \dots\dots\dots(2.14)$$

### Lapisan 4

Setiap simpul pada lapisan ini adalah simpul adaptif dengan fungsi simpul :

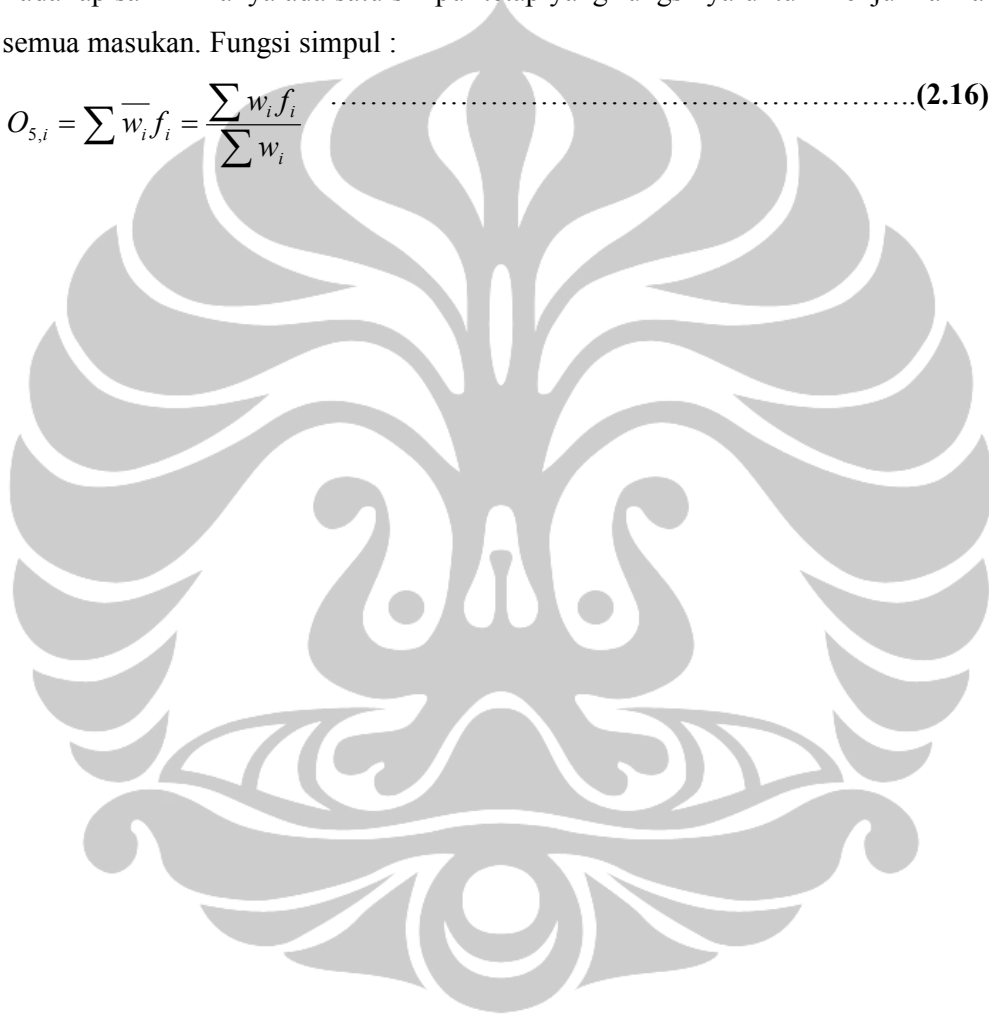
$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \dots\dots\dots(2.15)$$

Dengan  $O_{4,l}$  adalah derajat pengaktifan ternormalisasi dari lapisan 3 dan parameter  $p, q, r$  menyatakan parameter konsekuen yang adaptif.

Lapisan 5

Pada lapisan ini hanya ada satu simpul tetap yang fungsinya untuk menjumlahkan semua masukan. Fungsi simpul :

$$O_{5,i} = \sum \overline{w_i} f_i = \frac{\sum w_i f_i}{\sum w_i} \dots\dots\dots(2.16)$$



# BAB III

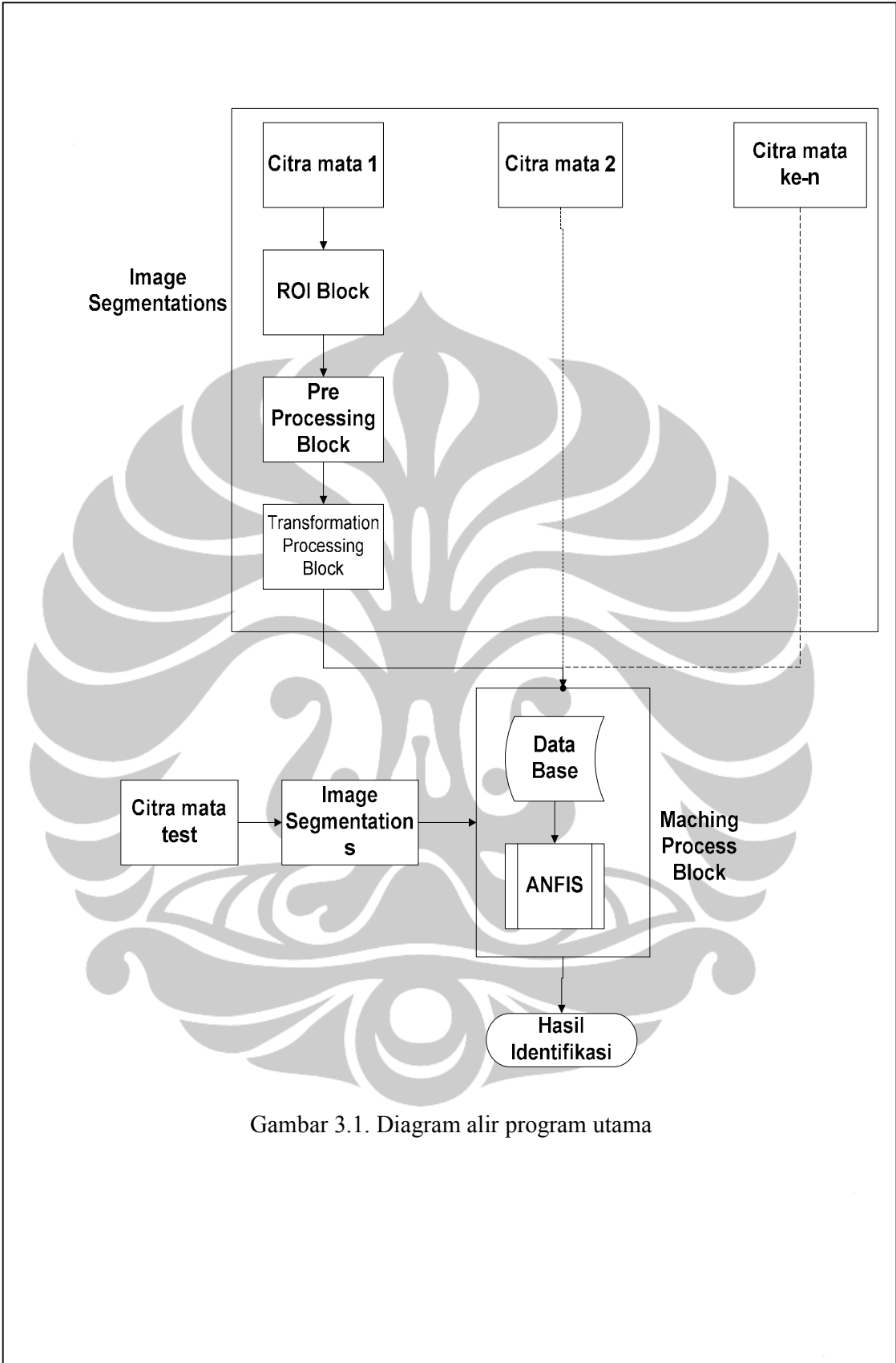
## ALGORITMA PERANCANGAN SISTEM IDENTIFIKASI IRIS DENGAN *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM*

### 3.1 PROSEDUR IDENTIFIKASI

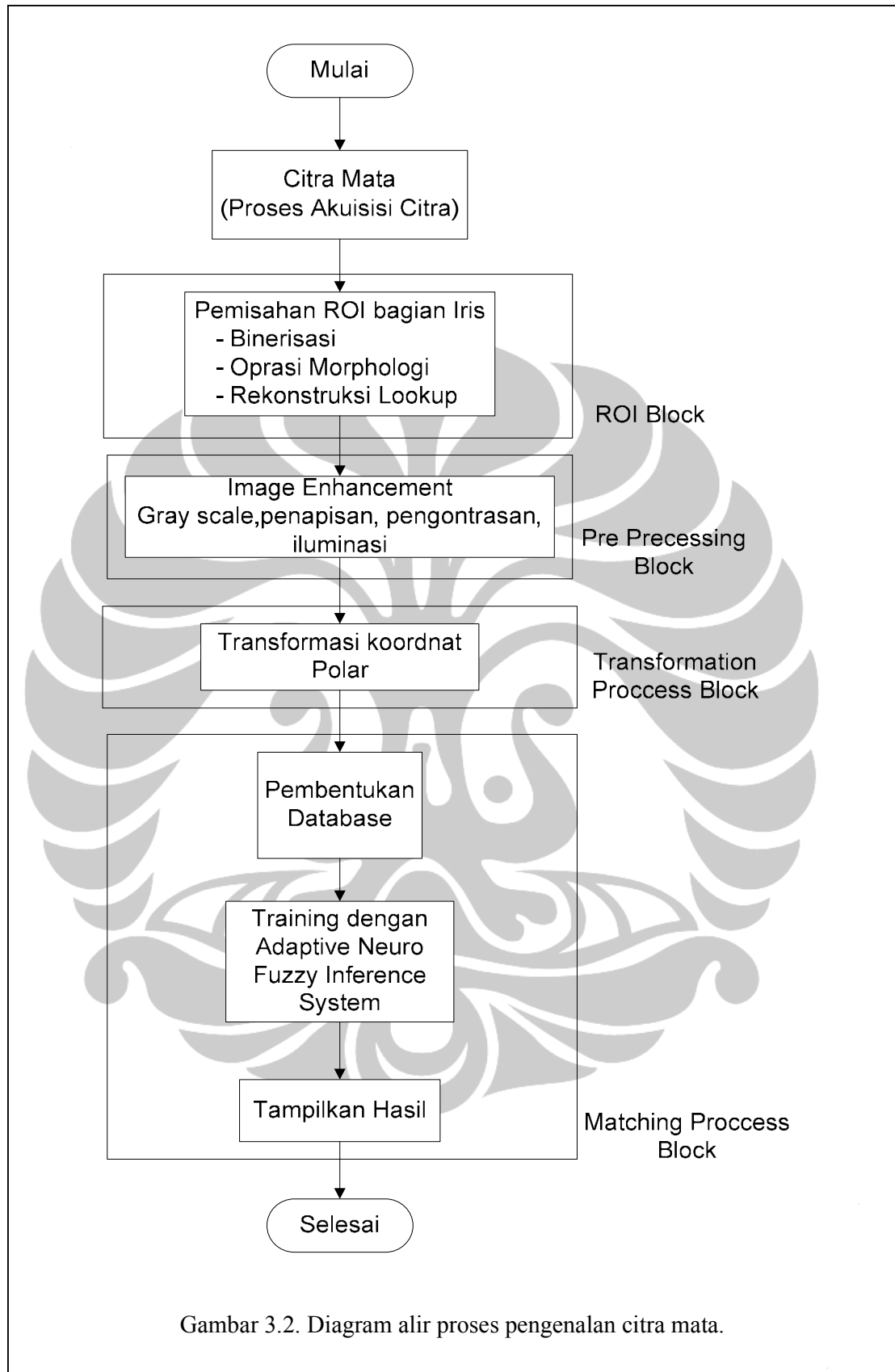
Bagian ini akan membahas mengenai perancangan algoritma yang digunakan untuk identifikasi iris mata dengan menerapkan metode *adaptive neuro-fuzzy inference system*. Data berupa sampel iris mata yang diambil dari database <http://pesona.mmu.edu.my/~ccteo/>. Data diambil dengan proses scanning yang dilakukan pada kedua mata (kanan dan kiri) dan diambil sampel masing – masing sebanyak 5 kali.

Program pada tugas akhir ini dirancang menggunakan perangkat lunak komputasi numerik dan banyak menggunakan fungsi terapan yang terdapat pada *Image Processing Toolbox*. Secara garis besar program ini terdiri dari tiga bagian utama yaitu proses segmentasi, normal encoding dan proses *matching*. Segmentasi/pemisahan ROI (*region of interest*) iris yaitu suatu teknik untuk memisahkan antara bagian iris mata dengan bagian mata yang lain yang tidak dibutuhkan. Dari proses ini akan didapat iris dan pupil. Normal encoding adalah merubah bentuk pola iris dari berbentuk lingkaran menjadi persegi panjang. Proses *matching* adalah proses mencocokkan data dari masukan dengan data yang ada pada database, dalam tugas akhir ini proses *matching* menggunakan algoritma dari *adaptive neuro-fuzzy inference system (ANFIS)*.

Program menerima masukan berupa citra mata dalam format bmp. Pengambilan citra mata ini dilakukan dengan menggunakan kamera infra merah sehingga citra yang dihasilkan sudah dalam format *grayscale* dengan *graylevel* 256 (8bit). Pada level ini pengolahan citra akan menjadi lebih sederhana dan lebih menghemat waktu jika dibandingkan dengan pengolahan citra dalam level rgb.



Gambar 3.1. Diagram alir program utama



### 3.1 ALGORITMA PEMISAHAN BAGIAN IRIS MATA/ROI BLOCK

Bagian pemisahan ROI (*region of interest*) iris dilakukan untuk memisahkan bagian iris mata yang penting, dari bagian-bagian lain mata yang tidak dibutuhkan atau bahkan mengganggu. Bagian dari mata yang tidak dibutuhkan akan dihilangkan sehingga hanya menyisakan bagian lingkaran iris dan pupil.

#### 3.1.1 Binerisasi

Citra mata terlebih dahulu diubah menjadi bentuk citra biner / hitam-putih melalui proses *threshold*. Fungsi perangkat lunak komputasi numerik yang digunakan dapat ditulis secara matematis:

```
citra=baca citra('gambar_mata.bmp');  
transform2bw = tranform to black and white(citra);
```

#### 3.1.2 Perataan Iluminasi

Kurang sempurnanya teknik pengambilan gambar kadangkala menyebabkan kurang meratanya penyebaran cahaya pada citra iris mata akibat intensitas iluminasi yang tidak sama pada daerah-daerah citra. Hal ini akan berpengaruh pada perubahan derajat keabuan yang terbentuk pada citra. *Noise* seperti ini harus diantisipasi dengan terlebih dahulu melakukan penyeragaman iluminasi pada keseluruhan citra masukan sebelum dilakukan pembentukan citra *grayscale*. Penyeragaman ini dapat dilakukan dengan mengatur distribusi histogram pada level 0-255 pada skala derajat keabuan. Pada pemrograman perangkat lunak komputasi numerik yang digunakan perataan ini telah dilakukan bersama-sama dalam proses pembentukan citra *grayscale*. Pada gambar mata yang diambil dari <http://pesona.mmu.edu.my/~ccteo/>, gambar diambil menggunakan kamera infra merah sehingga citra yang dihasilkan sudah dalam bentuk *grayscale*, sehingga proses iluminasi dapat dihilangkan.

#### 3.1.3 Operasi *Morphology*

Saat citra mata diambil terkadang ada cahaya yang dipantulkan ke mata pantulan tersebut akan menimbulkan bercak putih pada citra mata. Akibatnya saat



citra tersebut diubah ke bentuk biner bagian pantulan cahaya tersebut akan berwarna putih sehingga akan mengganggu untuk proses analisa. Untuk mengatasi masalah tersebut dapat dilakukan dengan operasi morfologi. Operasi ini terdiri dari operasi dilasi dan erosi secara berturut-turut pada data citra. Gambar 3.3 (b) menunjukkan hasil morphology dari citra iris mata. Dalam perangkat lunak komputasi numerik dapat digunakan fungsi sebagai berikut:

```

%Proses Morphologi Closing
morphologi = structure element('circle', 5);
% proses close morphologi, smoothing outer edge
Morphologismoothing = smooting(transform2bw, morphologi);

%Proses Morphologi Dilasi
Morphologi2 = structure element('circle', 5);
ImageDilasi = dilasi(Morphologismoothing, Morphologi2);
ImRekon=imagereconstruct(imagecomplement(ImageDilasi),
imagecomplement(Morphologismoothing));
ImRekon = imagecomplement(ImRekon);

%Proses Morphologi Erosi
Morphologi3 = structure element('circle', 5);
ImErosi = image erosi(ImRekon, morphologi);
ImRekon2 = reconstruct(ImErosi, ImRekon);

```



Gambar 3.3. (a) Citra mata asli; (b) Citra mata hasil morphology

### 3.1.4 Pencarian Titik Batas

Adakalanya dalam pengambilan citra terdapat bagian dari mata yang tidak terpakai seperti alis, kelopak mata. 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. Hal ini dilakukan dengan menetapkan titik pusat sementara, kemudian diikuti dengan proses scanning dari dalam ke luar sampai

menemukan titik batas yang dimaksud. Hasil pengukuran titik batas ini akan digunakan untuk rekonstruksi look up berikutnya. Program yang dapat digunakan adalah sebagai berikut.

```

% Penentuan titik pusat sementara
[M,N]=ukuran matriks(ImRekon2);

% Menentukan ukuran baris dan kolom
esty=round to integer (M/2);
estx=round to integer (N/2);

% Kanan
Kn = estx;
while(ImRekon2(esty,Kn)==0)
    Kn = Kn + 1;
end

% Kiri
Kr = estx;
while(ImRekon2(esty,Kr)==0)
    Kr = Kr - 1;
end

% Atas
At = esty;
while(ImRekon2(At,estx)==0)
    At = At + 1;
end

% Bawah
Bw = esty;
while(ImRekon2(Bw,estx)==0)
    Bw = Bw - 1;
end

% Tentukan urutan matriknya
M = (Bw-5) sampai (At+5);
N = (Kr-5) sampai (Kn+5);

```

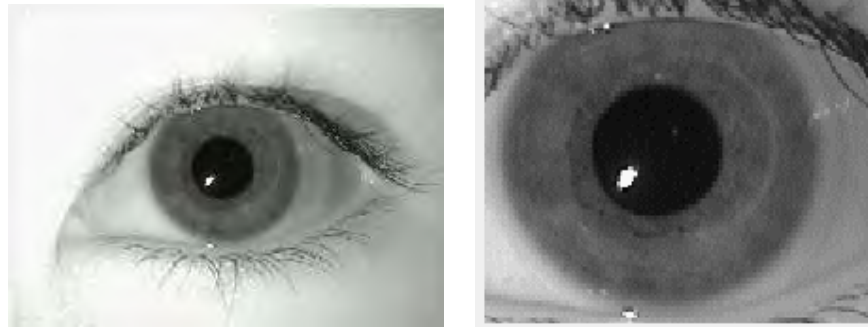
### 3.1.5 Rekonstruksi *Look Up*

Citra biner mata yang mengandung *glare* pada bagian iris perlu diperbaiki dengan melakukan operasi *morphology*, gambar 3.6 (b) menunjukkan citra hasil rekonstruksi. Algoritma yang digunakan adalah:

```

ImIris=BW(batas_bawah-5:batas_atas+5,batas_kiri-5:batas_kanan+5);

```



(a)

(b)

Gambar 3.4. (a) Citra mata asli; (b) Citra mata hasil rekonstruksi

### 3.2 ALGORITMA PEMROSESAN AWAL (*PREPROCESSING*)

Bagian *preprocessing* melakukan pengolahan awal citra mata agar dapat diolah lebih lanjut untuk diambil cirinya. Pada proses ini juga diharapkan agar derau yang muncul pada saat pengambilan gambar atau pada saat akuisisi citra dapat dihilangkan. Peningkatan kualitas gambar melalui perbaikan kontras, dan iluminasi dapat dilakukan untuk mempermudah proses pengolahan citra nantinya.

### 3.3 ALGORITMA TRANSFORMASI POLAR

Seperti yang telah dibahas pada bab II, bentuk geometris iris yang berupa lingkaran amat sulit untuk diolah lebih lanjut. Solusi bentuk antara yang harus diambil adalah dengan terlebih dahulu mengubahnya ke dalam bentuk citra koordinat polar. Pada bagian *Transformation Process* ingin diperoleh bentuk antara berupa citra transformasi polar dari iris mata yang berbentuk lingkaran. Dengan proses transformasi polar ini akan dihasilkan sebuah citra polar yang memiliki bentuk geometri persegi. Dengan bentuk seperti ini maka analisis dan klasifikasi dapat dilakukan dengan lebih mudah.

#### 3.3.1 Pembentukan Citra Iris ke Koordinat Polar

Pada dasarnya iris mata mempunyai bentuk geometri polar, yang merupakan sistem koordinat yang alami. Sistem Daugman's menggunakan *radial scaling* untuk kompensasi setiap ukuran sesuai dengan model variasi pupil berdasarkan *linear stretching*. Model *homogeneous rubber sheet* digunakan pada

setiap titik pada iris mata, tanpa melihat ukuran dan dilasi pupil dalam sebuah koordinat  $(r, \theta)$ , dimana  $r$  adalah unit interval  $(0,1)$  dan  $\theta$  adalah sudut  $(0,2\pi)$ . Pemetaan iris image  $I(x,y)$  dari koordinat kartesian  $(x,y)$ , kedalam koordinat polar  $(r, \theta)$ , dapat direpresentasikan sebagai berikut :

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \dots \dots \dots (3.1)$$

dimana  $x(r, \theta)$  dan  $y(r, \theta)$  didefinisikan sebagai kombinasi linear dari kedua titik batas pupil  $(xp(\theta), yp(\theta))$  dan titik batas limbus pada iris  $(xs(\theta), ys(\theta))$ , keduanya dapat dideteksi dengan menentukan operator maksimum dari persamaan sebagai berikut :

$$\begin{aligned} x(r, \theta) &= (1-r) xp(\theta) + r xs(\theta) \\ y(r, \theta) &= (1-r) yp(\theta) + r ys(\theta) \dots \dots \dots (3.2) \end{aligned}$$

Dari parameter-paramter ini dipakai untuk mengubah bentuk citra iris yang berbentuk lingkaran ke dalam koordinat polar. Citra ditransformasikan ke dalam bentuk polar dengan titik pusat sebagai acuannya.

Dalam pemrograman dengan piranti pemrograman perangkat lunak komputasi numerik, pembentukan polar ini di desain sebagai berikut :

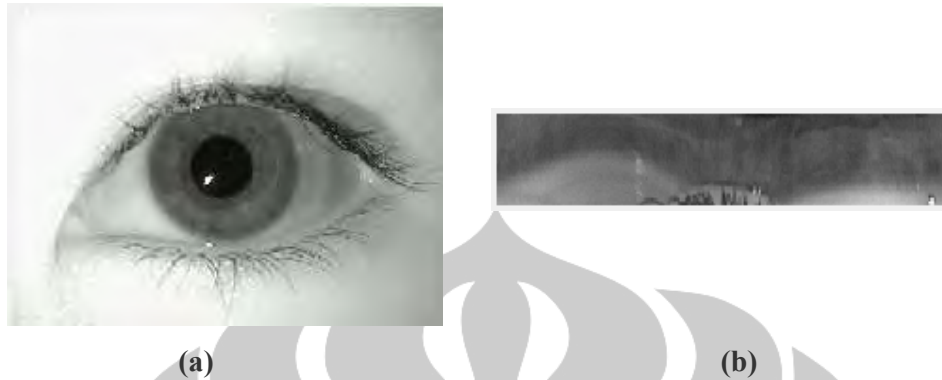
```
[Mr Nr] = ukuran matriks(imR); % ukuran citra persegi
Om = (Mr+1)/2; % koordinat pusat citra
On = (Nr+1)/2;

imP = zeros(M, N);

delR = (rMax - rMin)/(M-1);
delT = 2*pi/N;

% loop pada radius
for ri = 1:M
for ti = 1:N
r = rMin + (ri - 1)*delR;
t = (ti - 1)*delT;
x = r*cos(t);
y = r*sin(t);
xR = x*sx + Om;
yR = y*sy + On;
imP(ri, ti) = interpolate(imR, xR, yR);
end
end
```

Hasil dari transformasi polar seperti terlihat pada gambar 3.5 (b):



Gambar 3.5. (a) citra mata asli (b) citra mata dalam koordinat polar

### 3.4 PEMBENTUKAN *IRIS CODE* DENGAN *DISCRETE WAVELET TRANSFORM*

*Iris code* adalah metoda untuk mengekstrak suatu citra dengan tujuan untuk membedakan antara citra yang satu dengan citra yang lainnya. Tujuan dari pembentukan *iris code* ini adalah untuk menentukan bagian yang unik dari sebuah citra iris mata. Sebelum dikodekan terlebih dahulu citra mata polar di perbaiki kualitasnya dengan menggunakan fungsi histogram equalization, hal ini bertujuan untuk meningkatkan akurasi dalam proses identifikasi. Kemudian citra polar yang telah diolah, citra polar dipartisi menjadi tiga dimensi matriks kemudian masing – masing bagian dari matriks tersebut dibuat blok-blok dengan *distinct block processing*, kemudian ditransformasi dengan *discrete wavelet transform 2D* dengan *wavewlet name* adalah *daubechies4*. Dari transformasi ini didapat 3 parameter yaitu matriks citra diagonal, matriks citra vertikal dan matriks citra horisontal. Dari ketiga parameter ini digunakan untuk masukan (*input*) untuk pemrosesan selanjutnya, algoritma adalah sebagai berikut:

```
ImPolar1 = ImPolar(:,1); %membagi array menjadi 2 dimensional matriks  
ImPolar2 = ImPolar(:,2); %membagi array menjadi 2 dimensional matriks  
ImPolar3 = ImPolar(:,3); %membagi array menjadi 2 dimensional matriks
```

```

% partisi citra ke matriks 4-by-4
ImPolar1block= distinct block processing (ImPolar1, [4 4],f);
ImPolar2block= distinct block processing (ImPolar2, [4 4],f);
ImPolar3block= distinct block processing (ImPolar3, [4 4],f);

[cA1,cH1,cV1,cD1] = discrete wavelet transform 2D(ImPolarblock, daubechies4);
% mengkonstruksi level 1 approximation dan detil dari
% (A1, H1, V1, and D1) dari koefisien cA1, cH1, cV1, dan cD1
A1 = reconstruct dari 2D wavelet coef ('a',cA1, 'daubechies4',1);
H1 = reconstruct dari 2D wavelet coef ('h',cH1, 'daubechies4',1);
V1 = reconstruct dari 2D wavelet coef ('v',cV1,' daubechies4',1);
D1 = reconstruct dari 2D wavelet coef ('d',cD1,' daubechies4',1);

% transformasi kembali ke image coding
% mengkonstruksi level 1 Approximation
colormap(gray);
image(extended pseudocolor matrix scaling (A1,256));
image(extended pseudocolor matrix scaling (H1,256));
image(extended pseudocolor matrix scaling (V1,256));
image(extended pseudocolor matrix scaling (D1,256));

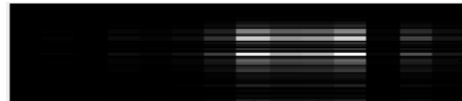
% resize koefisien matiks ke original size
cA= resize citra (cA1,4)
cD= resize citra (cD1,4)
cH= resize citra (cH1,4)
cV= resize citra (cV1,4)

```

gambar 3.6 (b) merupakan hasil dari pembentukan iris code dengan transformasi wavelet.



(a)

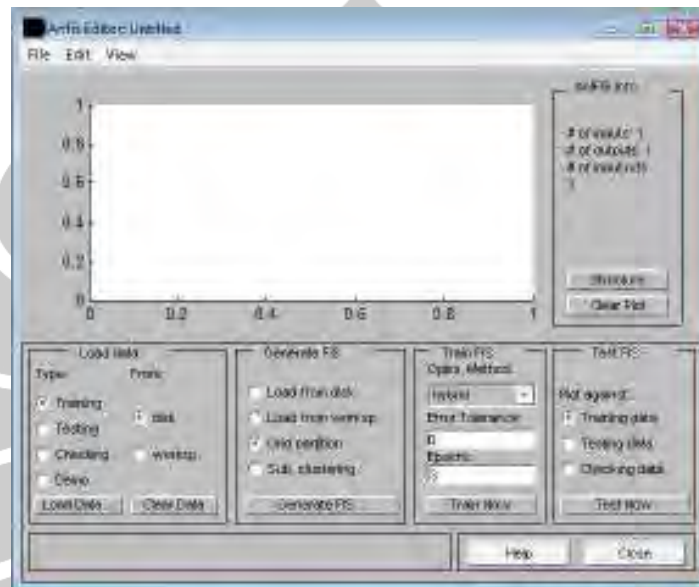


(b)

Gambar 3.6. (a) Citra polar sebelum di kodekan; (b) Citra polar hasil pengkodean.

### 3.5 ALGORITMA *MATCHING* DENGAN *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* (ANFIS)

Perancangan *fuzzy inference system* pada tugas akhir ini menggunakan *FIS Editor*, *Membership Function Editor* dan *rule editor* yang tersedia pada *fuzzy logic toolbox* yang sudah terintegrasi dengan perangkat lunak komputasi numerik. Gambar 3.7 menunjukkan *tools* untuk pembentukan file.fis dengan menggunakan GUI.



Gambar 3.7. ANFIS Editor

*Fuzzy inference system* adalah proses pemetaan dari suatu *input* ke *output* dengan menggunakan logika *fuzzy*. Pemetaan tersebut dapat menyediakan dasar pengambilan keputusan atau pola yang diperoleh. Proses *fuzzy inference system* melibatkan beberapa proses:

1. *Membership function*
2. *Fuzzy Logic Operator*
3. *If then Rules*

Terdapat dua jenis sistem inferensi *fuzzy* yang dapat diimplementasikan, tipe mamdani dan tipe sugeno. Sistem mamdani dan sugeno berbeda pada cara penentuan *output*. Pada penulisan tugas akhir ini diimplementasikan sistem inferensi *fuzzy* tipe sugeno. Pada metode mamdani, baik *input* (*anteseden*) maupun *output* (*konsekuen*) sistem berupa himpunan *fuzzy*. Penalaran dengan

metode sugeno, *output* sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linier.

Untuk melakukan perancangan sistem *fuzzy* perlu dilakukan beberapa tahap sebagai berikut:

1. mengidentifikasi karakteristik model secara fungsional dan operasional.
2. melakukan dekomposisi variabel model menjadi himpunan *fuzzy*.
3. membuat aturan *fuzzy*.
4. menentukan metode defuzzifikasi untuk tiap-tiap variabel solusi.

#### 1. Menentukan karakteristik model secara fungsional dan operasional

Pada bagian ini perlu diperhatikan karakteristik apa saja yang dimiliki oleh sistem yang ada, kemudian dirumuskan karakteristik operasi-operasi yang akan digunakan dalam model *fuzzy*. Ditentukan pula batasan-batasan model berupa perbandingan himpunan *fuzzy*, *hedge*, dan definisi dari beberapa variabel. Pada sistem identifikasi iris ini terdapat 3 *input mf* dan 1 *mf output*, *input* diperoleh dari nilai rata-rata diagonal, nilai rata-rata vertikal dan nilai rata-rata horisontal. *Output* dari sistem *fuzzy* adalah iris mata hasil identifikasi.

#### 2. Membentuk *Membership Function*

Fuzzifikasi adalah proses membuat suatu nilai *crisp* menjadi bersifat *fuzzy*. Hal ini dengan menganggap kuantitas yang selama ini *crisp* dan deterministik sebenarnya bersifat ambigu, impresisi dan mengandung ketidakpastian. Representasi data yang bersifat ambigu sebagai himpunan *fuzzy* dilakukan dengan menetapkan fungsi keanggotaan dapat dilakukan dengan beberapa cara. Pada penulisan ini digunakan untuk membangun nilai keanggotaan dari variabel *fuzzy* adalah *inductive reasoning*.

Penentuan fungsi keanggotaan dari *input* (masukan) dilakukan berdasarkan hasil percobaan dari tiga buah iris mata dari lima buah iris mata tiap-tiap sample.

Tabel 3.1 merupakan hasil dari nilai rata-rata diagonal, rata-rata vertikal dan nilai rata-rata horisontal dari tiap-tiap sample mata



Tabel III.1 Parameter dari Tiap – Tiap Iris Mata

Nama	Rata-rata Diagonal	Rata-rata Vertikal	Rata-rata Horisontal
Aeva Left	0.0021	0.0091	0.0061
	0.0011	0.0094	0.0038
	0.0013	0.0135	0.0043
Aeva Right	0.0024	0.0123	0.0115
	0.0017	0.0099	0.0058
	0.0019	0.0087	0.0064
Bryan Left	0.0012	0.0177	0.0036
	0.0012	0.0064	0.0036
	0.0013	0.0112	0.0042
Bryan Right	0.0012	0.0071	0.005
	0.0019	0.0201	0.0062
	0.0013	0.0091	0.0042
Lowyf Left	0.0043	0.0121	0.0103
	0.0021	0.0109	0.0071
	0.0012	0.0103	0.0058
Lowyf Right	0.0022	0.0134	0.0057
	0.0026	0.0104	0.0098
	0.0015	0.0117	0.0048
Ngkokwhy Left	0.0025	0.0234	0.0174
	0.0022	0.0216	0.0083
	0.0021	0.0204	0.0058
Ngkokwhy Right	0.0025	0.019	0.0161
	0.0049	0.0151	0.0113
	0.0044	0.0096	0.0186
Liujuw Left	0.0027	0.0202	0.0088
	0.0016	0.0089	0.0061
	0.0022	0.0145	0.008
Liujuw Right	0.0021	0.012	0.0074
	0.0025	0.0087	0.0083
	0.0027	0.0204	0.0108

### 3. Pembentukan Aturan If Then Rules

Aturan pada suatu model *fuzzy* menunjukkan bagaimana suatu sistem beroperasi. Sebagai contoh pembentukan aturan pada sistem inferensi *fuzzy*, akan dijelaskan cara pembentukan *rule* untuk iris mata Aeva untuk mata sebelah kiri, dalam program ditulis sebagai AevaL. Mempunyai tiga buah masukan (input) yaitu nilai rata-rata diagonal (meanD), nilai rata-rata vertikal (meanV) dan nilai rata-rata horisontal (meanH). Bila fungsi keanggotaan AevaL berada pada himpunan yang sama maka *rule* yang digunakan untuk membentuk daerah *fuzzy*

yang berhubungan dengan variable solusi AevaL adalah:

**IF** (meanD is AevaL) **AND** (meanV is AevaL) **AND** (meanH)  
**THEN** (identify is AevaL)

dengan aturan yang sama dibentuk untuk aturan fungsi keanggotaan yang lainnya.

#### 4. Menentukan Metode Defuzzyfikasi

Input dari defuzzyfikasi adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Metode yang digunakan dalam penulisan ini adalah metode *centroid*.

### 3.6 PEMBENTUKAN DATABASE UNTUK MENENTUKAN MEMBERSHIP FUNCTION PADA ANFIS

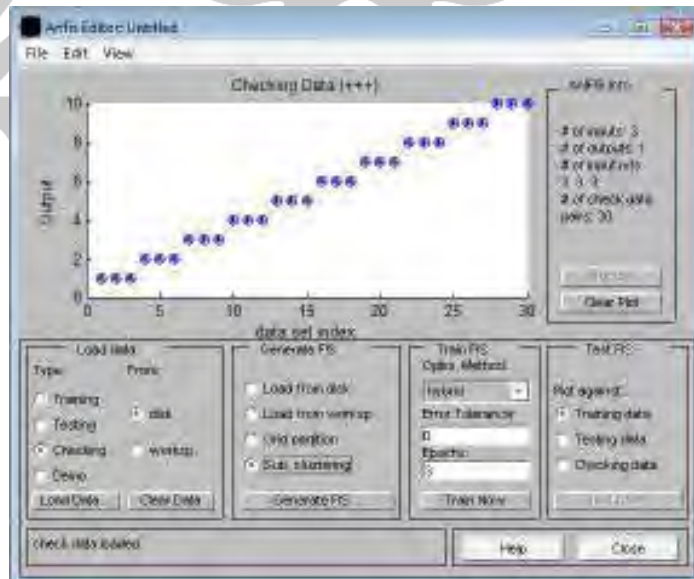
Pembentukan *membership function* dalam jaringan adaptif *fuzzy* inferensi sistem adalah dengan membentuk file .dat dengan memasukkan nilai dari rata-rata diagonal, rata-rata vertikal, rata-rata horisontal, dan nilai dari bobot atau *weight* dari citra yang disimpan dalam *database*. Gambar 3.8 merupakan contoh pembentukan database menggunakan *notepad* mengubah ekstensi menjadi .dat

rata-rata diagonal	rata-rata horisontal	rata-rata vertikal	wght
0.000393	0.0039	0.0012	1
0.000333	0.0037	0.0015	1
0.000345	0.004	0.000801	1
0.000862	0.0047	0.0025	2
0.000364	0.0041	0.0014	2
0.000405	0.0030	0.0012	2
0.000295	0.0061	0.001	3
0.000308	0.0015	0.0011	3
0.000481	0.0043	0.0015	3
0.000475	0.0044	0.0016	3
0.000567	0.007	0.002	4
0.000355	0.0041	0.0013	4
0.000399	0.0038	0.0023	5
0.000345	0.0035	0.0018	5
0.000261	0.0034	0.000972	5
0.000430	0.0038	0.0015	6
0.000326	0.0035	0.0014	6
0.000290	0.0037	0.000535	6

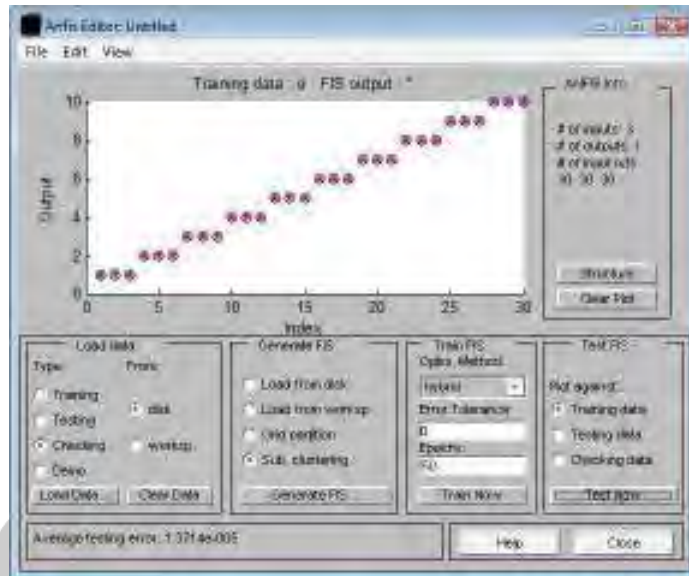
Gambar 3.8. Pembentukan *database*

### 3.6.1 Proses *Training Adaptive Neuro-Fuzzy Inference System*

Dari *command window* ketikkan *anfisedit* lalu tekan *enter*, maka akan muncul GUI *dialog box* seperti gambar 3.7 diatas. Kemudian data ditraining dengan memilih tipe fungsi keanggotaan *gaussian* dan dilakukan proses *training* sebanyak 50 kali. Proses *training* ini menggunakan metode *hybrid* yaitu proses pembelajaran yang terdiri atas dua bagian yaitu arah maju (*forward pass*) dan arah mundur (*backward pass*) Pada arah maju, parameter premis dibuat tetap. Dengan menggunakan metode *Recursive Least Square Estimator (RLSE)*, parameter konsekuen diperbaiki berdasarkan pasangan data masukan-keluaran. Metode RLSE dapat diterapkan karena parameter konsekuen yang diperbaiki adalah parameter linier. Metode RLSE akan mempercepat proses belajar hibrid. Kemudian setelah parameter konsekuen didapatkan, data masukan dilewatkan jaringan adaptif kembali dan hasil keluaran jaringan adaptif ini dibandingkan dengan keluaran yang sebenarnya. Pada arah mundur, parameter konsekuen dibuat tetap. Kesalahan yang terjadi antara keluaran jaringan adaptif dan keluaran sebenarnya dipropagasikan balik dengan menggunakan *gradient descent* untuk memperbaiki parameter premis. Pembelajaran ini dikenal sebagai Algoritma *Backpropagation-error*. Dari *database* yang telah dibentuk kemudian di-*training*, *testing* dan *dicheck*. Gambar 3.9 menunjukkan data masukan dari data yang akan di-*training*, *testing* dan *checking*:

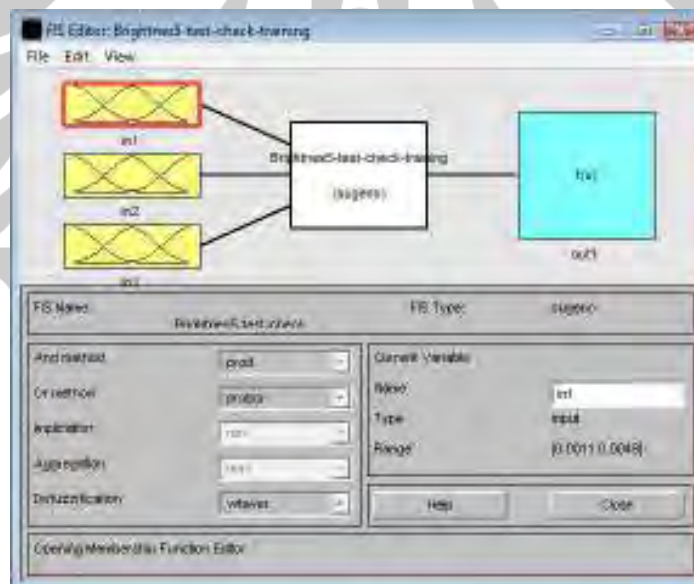


Gambar 3.9. Database yang di-*training*, *testing* dan *checking*



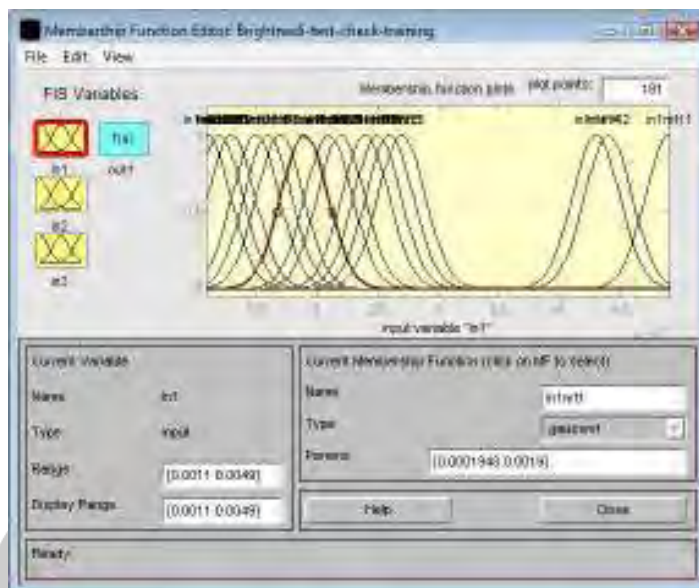
Gambar 3.10. Data setelah di-training

Gambar 3.10 menunjukkan data yang telah di-training dengan *epoch* sebanyak 50. Tanda o berwarna biru adalah data training sedangkan tanda \* berwarna merah adalah hasil dari proses *training* yang telah dilakukan.



Gambar 3.11. hasil pembentukan *rule base* dari *database*

Gambar 3.11 merupakan pembentukan *rule base* dari *database* dengan tiga masukan dan satu keluaran.



Gambar 3.12. Himpunan keanggotaan masukan pertama

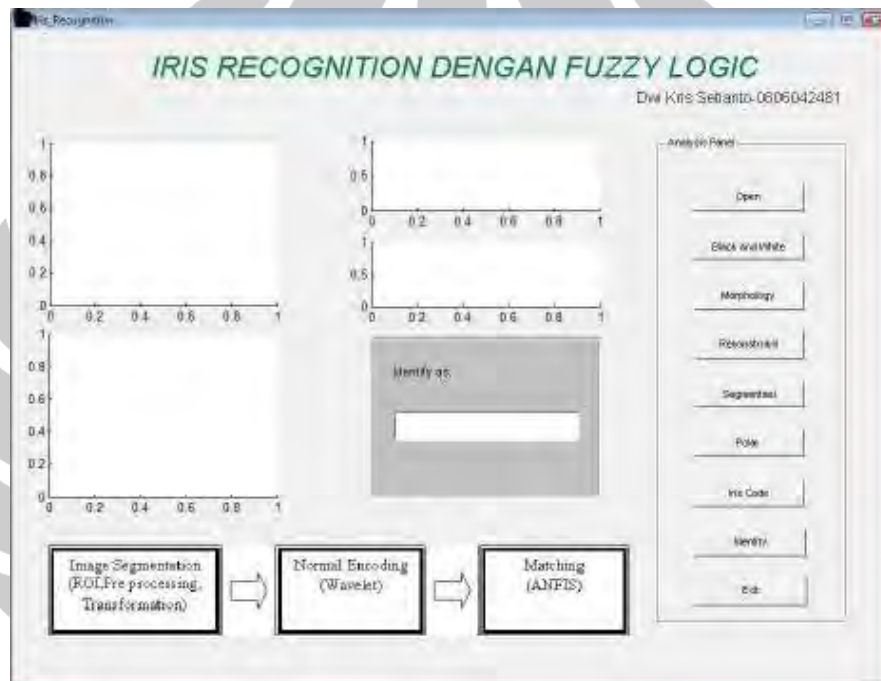
Gambar 3.12 merupakan fungsi keanggotaan dari masukan pertama hasil dari pembentukan *database* yang telah dibentuk pada sistem inferensi fuzzy.

## BAB IV

### HASIL SIMULASI DAN ANALISA

#### 4.1 PROSEDUR SIMULASI

Simulasi dilakukan untuk menguji tingkat akurasi sistem pengenalan iris mata manusia menggunakan logika fuzzy untuk membedakan iris yang satu dengan yang lainnya. Untuk melakukan simulasi pertama-tama jalankan M file dengan nama file `iris_recognition.m`. M file tersebut akan menampilkan figure seperti yang terlihat pada Gambar 4.1.



Gambar 4.1. Tampilan *Figure* program

Pada *figure* tersebut terdapat Sembilan tombol yang mempunyai fungsi sebagai berikut:

1. Fungsi *Open*, untuk membuka file citra iris yang diinginkan.
2. Fungsi *Black and White*, untuk merubah dari citra grayscale menjadi citra hitam putih.
3. Fungsi *Morphology*, untuk melakukan proses-proses morphology seperti dilasi, erosi dan lain-lain.

4. Fungsi *Reconstruct*, untuk merekonstruksi citra yang telah diolah sebelumnya.
5. Fungsi *Segmentasi*, untuk memisahkan bagian iris dengan bagian mata yang lainnya yang tidak dibutuhkan dalam pengolahan selanjutnya.
6. Fungsi *Polar*, untuk transformasi citra mata dari koordinat rectangular ke koordinat polar.
7. Fungsi *Iris Code*, untuk mengkodekan citra polar.
8. Fungsi *Identify*, untuk melakukan proses identifikasi.
9. Fungsi *Exit*, untuk keluar dari program.

Dari hasil pengujian yang telah dilakukan, data yang diolah terdiri dari 50 buah citra iris mata dengan 30 citra yang telah dilatih dan 20 citra yang belum dilatih. Bahan uji yang dimasukkan adalah semua citra yang telah dilatih dan yang belum dilatih. Dari hasil training yang telah dilakukan dengan menggunakan tiga masukkan (input) dengan jumlah data sebanyak 30 citra dengan pengaturan epoch 50, metode pembelajaran yang dilakukan adalah hybrid, dan *mf* yang digunakan adalah tipe gaussian didapat kesalahan rata-rata sebesar  $2.4995 \times 10^{-5}$ .

#### **4.2 KARAKTERISTIK CITRA MATA**

Citra Mata yang digunakan adalah citra mata yang sudah melalui proses dijitasi dan dalam bentuk grayscale. Selanjutnya citra mata diakuisisikan kedalam program untuk dilakukan pre-processing yang kemudian dilanjutkan untuk process pelatihan maupun proses identifikasi.

Citra yang sudah diakuisisi harus mengikuti langkah – langkah yang sudah dipersiapkan agar didapat hasil yang lebih akurat. Dalam proses perlangkah tersebut terdapat proses morfologi yang digunakan untuk mendapatkan ruang citra yang diinginkan, kemudian dilakukan perbaikan kualitas citra dengan histogram equalization yang fungsinya dapat meningkatkan akurasi dari proses identifikasi, diikuti pembentukan kekoordinat polar yang gunanya membentuk citra iris menjadi suatu bentuk persegi panjang. Untuk mendapatkan nilai unik

dari setiap citra dilakukan ekstraksi tekstur yang kemudian citra tersebut diambil karakteristiknya dengan menggunakan fungsi *discrete wavelet transform 2D* membentuk menjadi matrik menjadi tiga bagian, diagonal, vertikal dan bagian horisontal. Dari ketiga parameter ini kemudian dimasukkan sebagai masukan *recognition tools* (ANFIS) yang kemudian digunakan sebagai proses pembelajaran.

### 4.3 ANALISA DATA

Dari hasil uji coba yang telah dilakukan didapat : citra sebanyak 31 buah teridentifikasi (62%), 19 citra salah identifikasi (38%).

Berikut data hasil uji coba yang telah dilakukan:

Tabel IV.1. Data hasil uji coba untuk semua data training dan data uji

<b>Nama</b>	<b>Teridentifikasi</b>	<b>Salah identifikasi</b>
Aeva Left	4	1
Aeva Right	1	4
Bryan Left	3	2
Bryan Right	3	2
Lowyf Left	2	3
Lowyf Right	4	1
Ngkokwhy Left	4	1
Ngkokwhy Right	3	2
Liuwj Left	4	1
Liuwj Right	3	2
jumlah	31	19

Tabel IV.2. Hasil uji untuk citra yang telah ditraining

<b>Nama</b>	<b>Teridentifikasi</b>	<b>Salah identifikasi</b>
Aeva Left	3	0
Aeva Right	1	2
Bryan Left	3	0
Bryan Right	3	0
Lowyf Left	2	1
Lowyf Right	3	0
Ngkokwhy Left	3	0
Ngkokwhy Right	3	0
Liuwj Left	3	0
Liuwj Right	3	0
jumlah	27	3



Tabel IV.3. Hasil uji untuk citra yang tidak ditraining

Nama	Teridentifikasi	Salah identifikasi
Aeva Left	1	1
Aeva Right	0	2
Bryan Left	0	2
Bryan Right	0	2
Lowyf Left	0	2
Lowyf Right	1	1
Ngkokwhy Left	1	1
Ngkokwhy Right	0	2
Liujuw Left	1	1
Liujuw Right	0	2
jumlah	4	16

Dari tabel 4.2 terlihat hasil akurasi untuk data yang telah dilatih sebanyak 27 citra dari 30 citra (90% teridentifikasi) sedangkan dari tabel 4.3 untuk data yang tidak ditraining memiliki akurasi sebanyak 4 citra dari 16 citra (25% teridentifikasi).

Faktor yang mempengaruhi dari akurasi pengenalan citra ini adalah fungsi keanggotaan yang digunakan sebagai pembanding mempunyai bentuk dan parameter yang belum mewakili iris mata dengan perbedaan posisi pengambilan citra tersebut. Disamping itu nilai rata-rata antara iris yang satu dengan yang lainnya tidak terlalu berbeda jauh. Diperlukan data sampling yang lebih banyak lagi dalam berbagai posisi pengambilan citra sehingga kurva fungsi keanggotaan yang terbentuk dapat mewakili setiap iris mata. Faktor lainnya adalah karena nilai dari ketiga parameter input hasil data sampling mempunyai kurva yang beririsan sehingga kesalahan pengenalan akan mudah terjadi. Selain itu kurangnya akurasi juga dipengaruhi oleh pengambilan citra yang terletak tidak di tengah sehingga akan mempengaruhi dalam proses pengambilan *region of interest* iris.

## **BAB V**

### **KESIMPULAN**

1. Perbaikan kualitas citra dengan menggunakan histogram equalization dapat meningkatkan akurasi identifikasi sebesar 22%.
2. Pembentukan file.fis dengan menggunakan sub. clustering mempunyai nilai akurasi sebesar 62% sedangkan jika menggunakan grid partition hanya mempunyai akurasi 46%. Karena dalam grid partition hanya memperbanyak jumlah rule sedangkan pada grid partition dengan memperbanyak jumlah fungsi keanggotaan.
3. Data yang ditraining mempunyai akurasi sebesar 90%, karena setiap data yang ditraining mempunyai fungsi keanggotaan masing-masing sehingga nilai akurasinya besar.
4. Data yang tidak ditraining hanya mempunyai akurasi sebesar 25%, karena tidak setiap fungsi keanggotaan data yang ditraining dapat mewakili setiap data uji.

## DAFTAR ACUAN

- [1] "Badan anda adalah password," *Pikiran Rakyat*, 14 Desember 2006.
- [2] R.C. Gonzalez, R.E. Woods, S.L. Eddins, *Digital Image Processing*, 2nd Ed., (New Jersey : Prentice-Hall, 2004)
- [3] M Ch. Wijaya, Agus Prijono, *Pengolahan Citra Dijital Menggunakan MATLAB* (Bandung : Informatika, 2007)
- [4] Fahmi, "Perancangan Algoritma Pengolahan Citra Mata Menjadi Citra Polar Iris Sebagai Bentuk Antara Sistem Biometrik." Karya Ilmiah, Fakultas Teknik Universitas Sumatra Utara, Medan, 2007
- [5] J.F Faaborg, "Using Neural Network to Create an Adaptive Character Recognition System" (Ithaca NY: Cornell University, 2002)
- [6] Artificial Intellegence Technologies Tutorial (2007). *Artificial Neural Network*. Diakses 6 Mei 2008 dari Artificial Neural Network. <http://www.learnartificialneuralnetworks.com>
- [7] Hendra. "Identifikasi Uang Kertas Rupiah dengan Logika Fuzzy". Skripsi , Program Sarjana Fakultas Teknik UI, Depok, 2008.

## DAFTAR PUSTAKA

R.C. Gonzalez, R.E. Woods, S.L. Eddins, *Digital Image Processing*, 2nd Ed., (New Jersey : Prentice-Hall, 2004)

Jong Jek Siang, Msc, *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*, (Yogyakarta: Andi, 2004)

M Ch. Wijaya, Agus Prijono, *Pengolahan Citra Dijital Menggunakan MATLAB* (Bandung : Informatika, 2007)

