

**IDENTIFIKASI GOLONGAN DARAH MANUSIA DENGAN
TEKNIK PENGOLAHAN CITRA MENGGUNAKAN
METODE JARINGAN SYARAF TIRUAN**

SKRIPSI

Oleh

SANGKY ARYADHI

04 03 03 096 9



**SKRIPSI INI DIAJUKAN UNTUK MELENGKAPI
SEBAGIAN PERSYARATAN MENJADI SARJANA TEKNIK**

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

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul :

IDENTIFIKASI GOLONGAN DARAH MANUSIA DENGAN TEKNIK PENGOLAHAN CITRA MENGGUNAKAN METODE JARINGAN SYARAF TIRUAN

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 skripsi yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau Instansi manapun, kecuali yang secara tertulis diacu dalam naskah ini dicantumkan di dalam daftar pustaka.

Depok, 29 Juli 2008

Sangky Aryadhi

NPM 04 03 03 096 9

PENGESAHAN

Skripsi dengan judul:

**IDENTIFIKASI GOLONGAN DARAH MANUSIA DENGAN TEKNIK
PENGOLAHAN CITRA MENGGUNAKAN METODE JARINGAN SYARAF
TIRUAN**

Dibuat untuk melengkapi sebagian persyaratan sarjana menjadi Sarjana Teknik pada Program Studi Teknik Elektro Departemen Teknik Elektro Universitas Indonesia. Skripsi ini telah diujikan pada sidang ujian skripsi pada tanggal 7 Juli 2008 dan dinyatakan memenuhi syarat/sah pada Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia.

Depok, 29 Juli 2008

Dosen Pembimbing

Dr.Ir. Dodi Sudiana, M.Eng

NIP040 705 018 9

UCAPAN TERIMA KASIH

Puji syukur kehadirat Allah SWT atas segala rahmat yang diberikan sehingga dapat menyelesaikan skripsi ini. Saya mengucapkan terima kasih kepada Bapak Dr.Ir. Dodi Sudiana, M.Eng sebagai dosen pembimbing skripsi yang telah memberikan bimbingan, arahan, dan motivasi sehingga skripsi ini dapat diselesaikan dengan baik. Terima kasih kepada keluarga di rumah yang telah memberikan dukungan moral dan doa, serta seluruh teman-teman elektro 2003 yang membantu dalam penyelesaian skripsi ini.



ABSTRAK

Nama : Sangky Aryadhi

Program Studi: Teknik Elektro

Judul: Identifikasi Golongan Darah Manusia dengan Teknik Pengolahan Citra menggunakan Metode Jaringan Syaraf Tiruan

Dalam skripsi ini dibuat perangkat lunak yang dapat mengenali tipe ABO golongan darah manusia melalui pengolahan citra berbasis Jaringan Syaraf Tiruan (JST). Pola sampel darah yang direaksikan dengan reagen akan memiliki hasil pasangan penggumpalan yang berbeda. Jenis reagen yang direaksikan adalah antigen-A dan antigen-B. Pola pasangan darah dan reagen serta bentuk penggumpalan citra darah menjadi ciri dan kombinasi yang unik untuk membedakan tipe-tipe darah yang ada.

Data sampel darah berupa citra digital yang diperoleh menggunakan kamera digital. Sampel darah direaksikan dengan reagen di atas preparat dengan urutan reagen anti-A ditempatkan pada bagian kiri preparat dan reagen anti-B pada bagian kanan. Sebelum diolah dalam sistem JST, data melewati pra-proses, yaitu proses perbaikan citra digital yang terdiri dari proses pencuplikan, *gray-level quantization*, dan normalisasi dalam bentuk matriks. Hasil akhirnya berupa citra fitur hitam putih dalam matriks berukuran 5×5. Hasil pra-proses kemudian dihitung nilai parameter karakteristiknya, yaitu nilai rata-rata matriks tersebut. Nilai ini selanjutnya akan dijadikan input bagi proses pelatihan jaringan syaraf tiruan dengan metode *backpropagation*.

Jumlah data sampel sebanyak 120 set, dimana 80 di antaranya digunakan untuk melatih JST, dan sisanya digunakan untuk menguji JST. Hasil simulasi menunjukkan sistem mampu mengenali golongan darah dengan tingkat akurasi hingga 90%.

Kata kunci:

Pengolahan citra, jaringan syaraf tiruan, golongan darah

ABSTRACT

Name: Sangky Aryadhi

Study Program: Electrical Engineering

Title: Identification of Human Blood Type using Artificial Neural Network Method

In this research, an identification system of human blood type is designed using image processing techniques and the Artificial Neural Network (ANN) with backpropagation algorithms. The pattern of human blood type was formed using a chemical reaction between the blood and a reagent. The reagent that used in the reaction process are anti-A and anti-B reagent. Using a flat glass preparat as a media, the anti-A reagent is mixed to the blood sample on the left side and the anti-B reagent on the right side. Combination of blood coagulation pairs could distinguish the blood type.

The blood coagulation pair is converted into digital images after taken by a digital camera. The image is then pre-processed and normalized to 50x50 matrix size. The matrix is divided to different blocks and reduced to 5x5 grayscale image. The pre-processing involved sampling, gray-level quantization, and normalization. After pre-processing, the mean of 5x5 gray scale image will be calculated and used as the input for the ANN.

The total number of blood sample data is 140 pairs, 80 set of them are used for training process of the ANN and the rest are used for identification. The simulation result shows that the system is able to identify up to 90% level of accuracy.

Key words:

Image processing, artificial neural network, human blood type

DAFTAR ISI

	Halaman
PERNYATAAN KEASLIAN SKRIPSI.....	ii
PENGESAHAN.....	iii
UCAPAN TERIMA KASIH.....	iv
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	xi
DAFTAR LAMPIRAN.....	xii
DAFTAR SINGKATAN.....	xiii
DAFTAR ISTILAH.....	xiv
BAB I PENDAHULUAN.....	1
1.1 LATAR BELAKANG.....	1
1.2 TUJUAN.....	1
1.3 BATASAN MASALAH.....	2
1.4 SISTEMATIKA PENULISAN.....	2
BAB II DASAR TEORI.....	3
2.1 DARAH.....	3
2.1.1 Ciri golongan darah.....	4
2.2 PENGOLAHAN CITRA.....	5
2.2.1 Sampling dan <i>Gray-level</i> Quantization.....	6
2.2.2 Normalisasi Data.....	9
2.2.2.1 Intensity Adjustment.....	10
2.2.2.2 Cropping.....	10
2.2.2.3 Levelling dan Reshaping.....	10
2.2.2.4 Edge Detection.....	10
2.2.3 EKSTRAKSI.....	11
2.3 JARINGAN SYARAF TIRUAN.....	11
BAB III METODOLOGI PENELITIAN.....	17
3.1 DATA DAN PERALATAN PENELITIAN.....	17
3.2 LANGKAH Pengerjaan.....	17
3.3 PRA-PROSES.....	18
3.3.1 Pemotongan.....	19

3.3.2	Konversi Citra RGB Ke Citra Hitam-Putih Dan Kompresi Data	19
3.4	LABELISASI.....	20
3.5	PELATIHAN JST.....	20
3.6	SIMULASI IDENTIFIKASI	23
BAB IV HASIL UJI COBA DAN ANALISIS.....		25
4.1	HASIL UJI COBA.....	25
4.2	PENGARUH UKURAN MATRIKS MASUKAN PADA HASIL IDENTIFIKASI.....	26
4.3	PENGARUH METODE OPTIMASI.....	29
4.4	PENGARUH JUMLAH PELATIHAN	30
4.5	PENGARUH FUNGSI AKTIVASI DAN TARGET	39
BAB V KESIMPULAN.....		41
DAFTAR ACUAN		42
DAFTAR PUSTAKA		43
LAMPIRAN.....		44
Lampiran 1		44
Lampiran 2		44

DAFTAR GAMBAR

Halaman

Gambar 2.1 Jenis darah dilihat dari penggumpalannya setelah direaksikan dengan reagen [3]	5
Gambar 2.2 Diagram alir <i>image recognition</i>	6
Gambar 2.3 Sampling <i>array</i> 2 dimensi [4]	7
Gambar 2.4 <i>Image</i> disampling pada 256 x 256, 128 x128, 64 x 64 , 32 x 32 , 16 x 16 <i>rectangular</i> sampling grid [5]	8
Gambar 2.5 <i>Gray-level Quantization</i> [5]	9
Gambar 2.6 Diagram blok Jaringan Syaraf Tiruan	11
Gambar 2.7 Fungsi aktivasi sigmoid biner	13
Gambar 2.8 Fungsi aktivasi sigmoid bipolar	13
Gambar 2.9 Fungsi aktivasi linear	14
Gambar 3.1 Diagram alir pembuatan labelisasi (oval merah) dan training JST (kotak biru)	18
Gambar 3.2 Daerah pemotongan citra	19
Gambar 3.3 Hasil <i>grayscale</i> dengan intensitas 256 bit	19
Gambar 3.4 Topologi JST <i>backpropagation</i> yang digunakan pada simulasi	21
Gambar 3.5 Diagram alir proses pelatihan jaringan syaraf tiruan	23
Gambar 3.6 Tampilan perangkat lunak modul identifikasi	24
Gambar 4.1 Grafik pelatihan JST menggunakan matriks 10x10 dengan unjuk kerja 10^0	26
Gambar 4.2 Grafik pelatihan JST menggunakan matriks 5x5 dengan unjuk kerja 10^0	27
Gambar 4.3. Grafik pelatihan JST menggunakan matriks 5x5 dengan unjuk kerja 10^{-1}	28
Gambar 4.4. Grafik pelatihan JST menggunakan matriks 10x10 dengan unjuk kerja 10^{-1}	28
Gambar 4.5. data ke-6 dari golongan darah A	31
Gambar 4.6. Bentuk matriks dari golongan darah A data ke-6	32
Gambar 4.7. bentuk matriks dari golongan darah O	32
Gambar 4.8. data ke-9 dari golongan darah A	32

Gambar 4. 9. bentuk matriks dari golongan darah A data ke-9	32
Gambar 4. 10. data ke-1 dari golongan darah B	33
Gambar 4. 11. bentuk matriks dari golongan darah B data ke-1	33
Gambar 4. 12. data ke-3 dari golongan darah AB	34
Gambar 4. 13. bentuk matriks dari golongan darah AB data ke-3	34
Gambar 4. 14. data ke-5 dari golongan darah AB	35
Gambar 4. 15. bentuk matriks dari golongan darah AB data ke-5	35
Gambar 4. 16. data ke-6 dari golongan darah AB	35
Gambar 4. 17. bentuk matriks dari golongan darah AB data ke-6	35
Gambar 4. 18. data ke-7 dari golongan darah AB	36
Gambar 4. 19. bentuk matriks dari golongan darah AB data ke-7	36
Gambar 4. 20. data ke-8 dari golongan darah AB	36
Gambar 4. 21. bentuk matriks dari golongan darah AB data ke-8	36
Gambar 4. 22. data ke-10 dari golongan darah AB	37
Gambar 4. 23. bentuk matriks dari golongan darah AB data ke-10	37
Gambar 4. 24. data ke-8 dari golongan darah O	39
Gambar 4. 25. bentuk dari golongan darah O data ke-8	39
Gambar 4. 26. Fungsi sigmoid biner (logsig) dan sigmoid bipolar (tansig)	40

DAFTAR TABEL

	Halaman
Tabel II. 1 Hubungan resipien-donor golongan darah ABO	4
Tabel IV. 1 Tabel hasil uji coba identifikasi	25
Tabel IV. 2 Pengaruh besar matriks masukan	28
Tabel IV. 3 Tingkat akurasi dengan 10x10 pixel citra darah	29
Tabel IV. 4 Perbandingan Levenberg-Marquardt dan <i>Gradient descent</i> dengan input 5x5	29
Tabel IV. 5 Perbandingan jumlah pelatihan dengan akurasi tiap tipe golongan darah	30
Tabel IV. 6. Tabel identifikasi golongan darah A dengan input 5x5 dan metode optimasi LM	31
Tabel IV. 7. Tabel identifikasi golongan darah B dengan input 5x5 dan metode optimasi LM	33
Tabel IV. 8. Tabel identifikasi golongan darah AB dengan input 5x5 dan metode optimasi LM	34
Tabel IV. 9. Tabel identifikasi golongan darah O dengan input 5x5 dan metode optimasi LM	38

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Hasil uji coba identifikasi (5x5 pixel)	
optimasi <i>Levenberg-Marquadt</i>	44
Lampiran 2 Hasil uji coba identifikasi (10x10 pixel)	
optimasi <i>Levenberg-Marquadt</i>	44



DAFTAR SINGKATAN

ANN	Artificial Neural Network
GD	Gradient Descent
JPEG	Joint Picture Expert Group
JST	Jaringan Syaraf Tiruan
LM	<i>Levenberg-Marquadt</i>
RGB	Red, Green, Blue



DAFTAR ISTILAH

<i>Algoritma</i>	Sekumpulan langkah pengerjaan atau operasi dari suatu masalah
<i>Aliasing effect</i>	Hasil dari sampling citra, dimana frekuensi penyampling kurang dari dua kali frekuensi tertinggi pada citra. Efeknya seolah-olah elemen citra saling bertumpuk.
<i>Backpropagation</i>	Algoritma JST yang mengatur perubahan bobot jaringan melalui perhitungan keluaran dengan gradien negatif secara terbalik (<i>backward</i>)
<i>Descriptor</i>	Deskripsi objek berisi informasi yang diberikan oleh elemen citra
<i>Dijitalisasi</i>	Proses mengubah citra analog ke digital. Membutuhkan proses <i>sampling</i> dan <i>kuantisasi</i>
<i>Grayscale</i>	Tingkat keabu-abuan
<i>Hidden layer</i>	Lapisan tersembunyi pada ANN
<i>Hue</i>	Warna. Secara teknis <i>hue</i> berarti panjang gelombang dari warna.
<i>Image recognition</i>	Pengenalan objek citra
<i>JPEG</i>	<i>Joint Picture Expert Group</i> , format kompresi citra yang dikeluarkan oleh konsorsium JPEG tersebut
<i>Kompresi data</i>	Teknik mengurangi jumlah data yang diperlukan untuk merepresentasikan suatu citra.
<i>Kongruen</i>	Sama dan sebangun
<i>Kontras</i>	Variasi kecerahan antara terang-gelap pada area suatu citra
<i>Kuantisasi</i>	Mengkonversi data analog menjadi data digital
<i>Luminance</i>	Kecerahan warna, intensitas kecerahan dari suatu pixel
<i>Mean</i>	Nilai rata-rata
<i>Neural network</i>	Jaringan syaraf tiruan (JST)

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Teknologi yang serba cepat dan mudah selalu dikembangkan dan diaplikasikan pada kehidupan sehari-hari. Adanya perkembangan teknologi akan mempermudah manusia dalam melakukan berbagai hal secara mudah, cepat dan akurat. Kecepatan dan kemudahan untuk memproses data akan sangat penting dalam kehidupan sehari-hari, khususnya pada bidang medis. Dengan adanya perangkat lunak yang dapat mengidentifikasi tipe golongan darah, maka diharapkan proses identifikasi golongan darah dapat dilakukan secara otomatis dan tidak menggunakan tenaga manusia.

Pola dari golongan darah dapat diketahui dari pola penggumpalan kimiawi setelah ditetesi cairan reagen, untuk mengambil pola dari penggumpalan darah tersebut perlu dilakukan pengolahan citra. Teknik pengolahan citra dapat membantu memperbaiki citra dan menyederhanakan proses pelatihan dan pengidentifikasian.

Untuk mengidentifikasi suatu kumpulan data, terdapat beberapa macam teknik yang dapat digunakan, yaitu logika *fuzzy*, *Hidden Markov Model*, dan Jaringan Syaraf Tiruan (JST). Dengan adanya teknik pengidentifikasian tersebut, diharapkan proses pengidentifikasian menghasilkan tingkat keakurasian yang tinggi. Diantara ketiga metode tersebut, logika *fuzzy* merupakan teknik yang paling sederhana, tetapi memiliki hasil yang paling kurang akurat jika dibandingkan dengan dua metode lainnya. *Hidden Markov Model* pemrosesannya lebih cepat jika dibandingkan dengan JST, tapi masih memiliki tingkat keakurasian yang rendah dibandingkan dengan JST. Sedangkan JST memiliki tingkat akurasi yang lebih tinggi dari kedua teknik lainnya, JST menggunakan proses pelatihan dan iterasi dalam mengidentifikasi obyeknya.

1.2 TUJUAN

Tujuan skripsi ini adalah merancang bangun untuk mengidentifikasi golongan darah secara otomatis menggunakan JST.

1.3 BATASAN MASALAH

Masalah yang dibahas dalam skripsi ini adalah penggunaan pengolahan citra untuk pendeteksian objek pada citra digital sampel darah dengan 83 set data pelatihan dan 40 set data pelatihan yang sudah direaksikan dengan reagen. Hasil reaksi akan diproses menggunakan teknik pengolahan citra guna mendeteksi pola penggumpalannya serta menentukan tipe golongan darahnya menggunakan metode Jaringan Syaraf Tiruan (JST) menggunakan input sebesar 5x5 dan 10x10.

1.4 SISTEMATIKA PENULISAN

Bab satu, pendahuluan, menjelaskan latar belakang penulisan, tujuan penulisan, batasan masalah dan sistematika penulisan skripsi untuk memberikan gambaran umum mengenai penulisan skripsi.

Bab dua, dasar teori, menjelaskan dasar teori mengenai golongan darah, pengolahan citra dan metode Jaringan Syaraf Tiruan (JST).

Bab tiga, metodologi penelitian, menjelaskan tahap-tahap yang akan dikerjakan untuk membuat system identifikasi tipe golongan darah.

Bab empat, hasil uji coba dan analisis, bab ini berisi hasil uji coba dari pembuatan perangkat lunak beserta analisis mengenai pengaruh parameter-parameter yang digunakan dalam pelatihan JST terhadap kinerja sistem perangkat lunak.

Bab lima, kesimpulan, bab ini berisi kesimpulan dari hasil yang diperoleh pada penelitian ini.

BAB II

DASAR TEORI

2.1 DARAH

Darah adalah cairan yang terdapat pada semua hewan tingkat tinggi yang berfungsi mengirimkan zat-zat dan oksigen yang dibutuhkan oleh jaringan tubuh, mengangkut bahan-bahan kimia hasil metabolisme, dan juga sebagai pertahanan tubuh terhadap virus atau bakteri. Istilah medis yang berkaitan dengan darah diawali dengan kata *hemo-* atau *hemato-* yang berasal dari bahasa Yunani *haima* yang berarti darah.[1]

Darah manusia adalah cairan jaringan tubuh. Fungsi utamanya adalah mengangkut oksigen yang diperlukan oleh sel-sel di seluruh tubuh. Darah juga menyuplai jaringan tubuh dengan nutrisi, mengangkut zat-zat sisa metabolisme, dan mengandung berbagai bahan penyusun sistem imun yang bertujuan mempertahankan tubuh dari berbagai penyakit. Hormon-hormon dari sistem *endokrin* juga diedarkan melalui darah.

Tiap individu memiliki jenis-jenis ciri darah yang berbeda karena adanya perbedaan jenis karbohidrat dan protein pada permukaan membran sel darah merah. Oleh karena itu darah juga memiliki penggolongan untuk mengklasifikasikan jenis-jenis darah berdasarkan parameter-parameter yang merepresentasikan suatu karakteristik yang unik pada tiap golongannya, saat ini terdapat 29 macam cara untuk menggolongkan darah. Skripsi ini akan menggunakan cara ABO untuk menggolongkan jenis darah.

Golongan darah manusia ditentukan berdasarkan jenis antigen dan antibodi yang terkandung dalam darahnya [2], sebagai berikut:

- 1) Individu dengan golongan darah A memiliki sel darah merah dengan antigen A di permukaan membran selnya dan menghasilkan antibodi terhadap antigen B dalam serum darahnya. Sehingga, orang dengan golongan darah A hanya dapat menerima darah dari orang dengan golongan darah A atau O.
- 2) Individu dengan golongan darah B memiliki antigen B pada permukaan sel darah merahnya dan menghasilkan antibodi terhadap antigen A dalam serum

darahnya. Sehingga, orang dengan golongan darah B hanya dapat menerima darah dari orang dengan golongan darah B atau O

- 3) Individu dengan golongan darah AB memiliki sel darah merah dengan antigen A dan B serta tidak menghasilkan antibodi terhadap antigen A maupun B. Sehingga, orang dengan golongan darah AB dapat menerima darah dari orang dengan golongan darah ABO apapun dan disebut *resipien universal*.
- 4) Individu dengan golongan darah O memiliki sel darah tanpa antigen, tapi memproduksi antibody terhadap antigen A dan B. Sehingga, orang dengan golongan darah O dapat mendonorkan darahnya kepada orang dengan golongan darah ABO apapun dan disebut *donor universal*.

Secara umum hubungan resipien-donor pada penggolongan darah ABO dapat dilihat pada Tabel II.1:

Tabel II. 1 Hubungan resipien-donor golongan darah ABO

Recipient ^[1]	Donor ^[1]							
	O-	O+	A-	A+	B-	B+	AB-	AB+
O-	O-							
O+	O-	O+						
A-	O-		A-					
A+	O-	O+	A-	A+				
B-	O-				B-			
B+	O-	O+			B-	B+		
AB-	O-		A-		B-		AB-	
AB+	O-	O+	A-	A+	B-	B+	AB-	AB+

2.1.1 Ciri golongan darah

Untuk menentukan jenis golongan darah, darah harus direaksikan dengan cairan kimia yang biasa disebut reagen. Jenis reagen ada 3, yaitu:

- 1) Reagen antigen A, digunakan untuk mendeteksi golongan darah B. Apabila darah dengan golongan B ditetaskan dengan reagen antigen A, maka akan timbul gumpalan-gumpalan darah yang disebabkan oleh reaksi pembentukan antibodi terhadap antigen A
- 2) Reagen antigen B, digunakan untuk mendeteksi golongan darah A. Apabila darah dengan golongan A ditetaskan dengan reagen antigen B, maka akan

timbul gumpalan-gumpalan darah yang disebabkan oleh reaksi pembentukan antibodi terhadap antigen B

- 3) Reagen antigen AB, digunakan untuk mendeteksi golongan darah AB. Apabila darah dengan golongan AB ditetaskan dengan reagen antigen AB, maka akan timbul gumpalan-gumpalan darah yang disebabkan oleh reaksi pembentukan antibodi terhadap antigen A dan antigen B.

Bentuk penggumpalan darah yang terjadi setelah direaksikan dengan cairan reagen dapat diamati secara makroskopis sehingga dapat dilihat secara kasat mata, gambar dari penggumpalan darah hasil reaksi dengan menggunakan reagen dapat dilihat pada Gambar 2.1.

BLOOD TYPE	ANTI-A	ANTI-B	ANTI-D	CONTROL
O-POSITIVE				
O-NEGATIVE				
A-POSITIVE				
A-NEGATIVE				
B-POSITIVE				
B-NEGATIVE				
AB-POSITIVE				
AB-NEGATIVE				
INVALID				

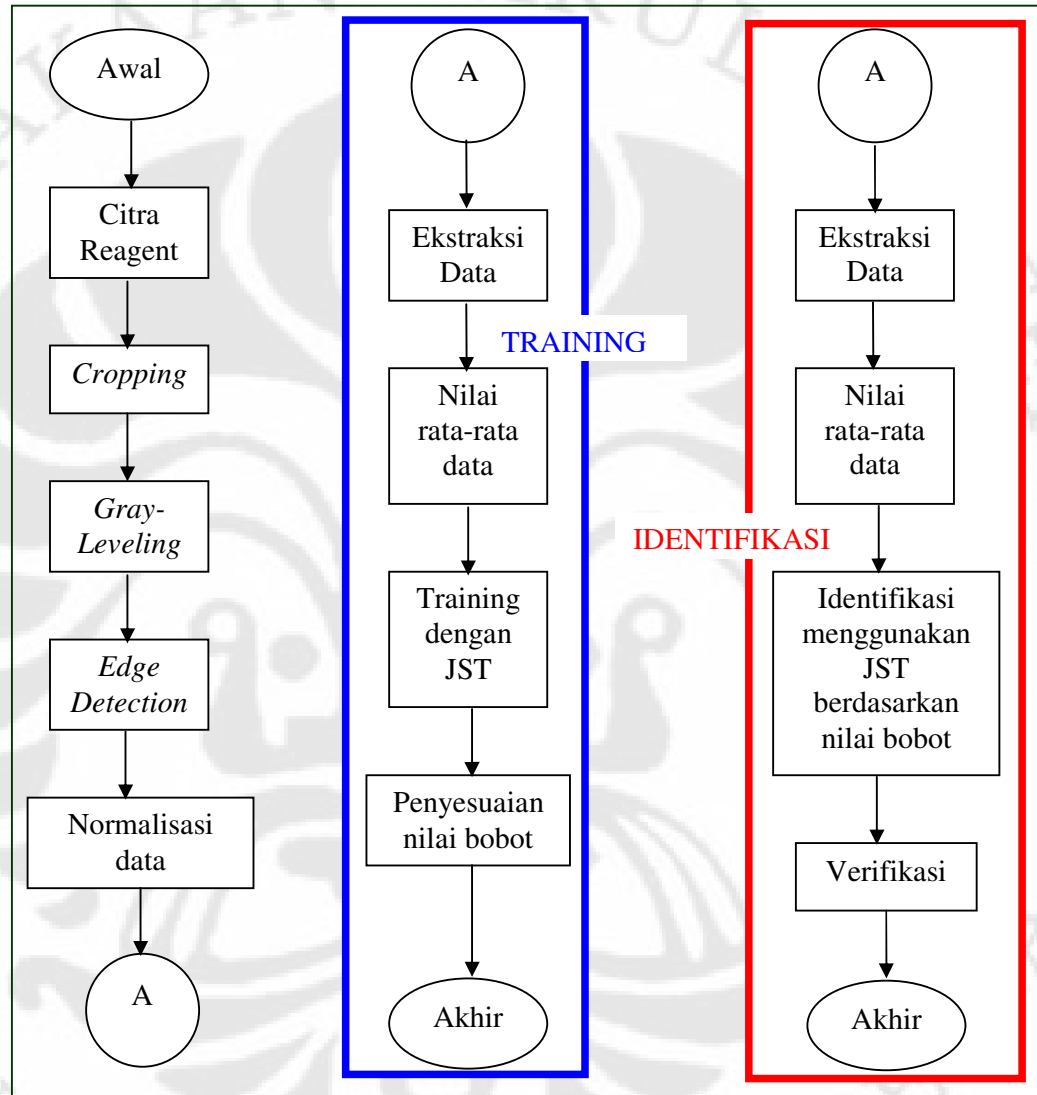
Gambar 2.1 Jenis darah dilihat dari penggumpalannya setelah direaksikan dengan reagen [3]

2.2 PENGOLAHAN CITRA

Pengolahan citra adalah suatu proses manipulasi data, dalam hal ini suatu citra digital (*image*) agar dapat lebih bermanfaat untuk hasil yang diinginkan. Proses pengolahan citra tidak terbatas pada *image* statis saja, pengolahan citra juga dapat dilakukan pada *image* bergerak (*video*). Dalam skripsi ini hanya akan menggunakan *image* statis sebagai objek penelitian dalam hal *object recognition*.

Recognition adalah suatu proses pemberian label kepada suatu objek berdasarkan informasi yang diberikan atau disebut *descriptor* [4].

Proses pengenalan *image* yang dilakukan dalam penulisan skripsi ini adalah dapat dilihat pada Gambar 2.2



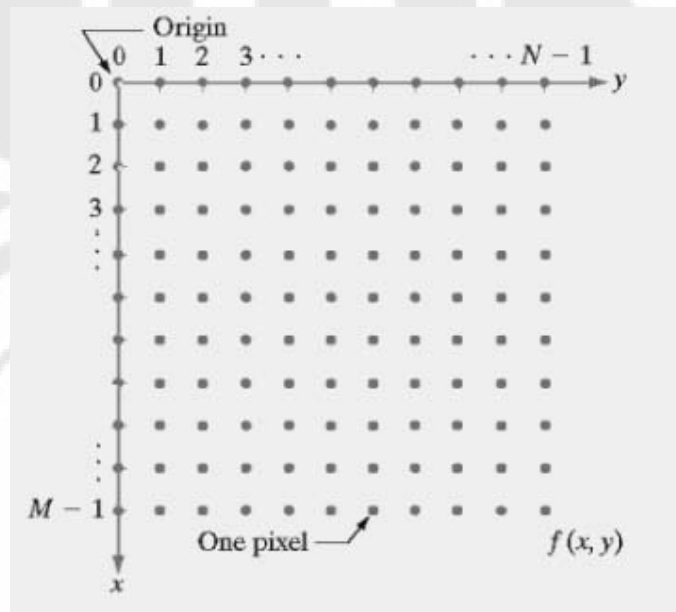
Gambar 2.2 Diagram alir *image recognition*

2.2.1 Sampling dan *Gray-level Quantization*

Pada sebuah *image* terdapat dua jenis perbaikan citra, yaitu perbaikan dalam domain spasial dan perbaikan citra dalam domain frekuensi. Dalam skripsi ini, perbaikan citra yang digunakan adalah perbaikan citra dalam domain spasial, dalam

domain spasial sebuah *image* direpresentasikan oleh koordinat x dan y sebagai suatu fungsi dua dimensi yang lebih sering disebut dengan istilah piksel, nilai pada koordinat (x,y) merepresentasikan tingkat warna dari suatu *image*, pada *image grayscale* nilai tersebut akan merepresentasikan intensitas keabuan pada piksel yang dimaksud.

Suatu *image* memiliki jumlah piksel yang berhingga pada dengan koordinat (x,y) sebagai penanda lokasi piksel yang dituju dan tiap piksel akan memiliki karakteristik yang berbeda tergantung dari nilainya. Untuk lebih jelasnya dapat dilihat Gambar 2.3 berikut ini:



Gambar 2.3 Sampling array 2 dimensi [4]

Suatu *image* $f(x,y)$ di-sampling menjadi $N \times M$ array matriks dimana setiap elemen dari array memiliki kuantitas diskrit seperti persamaan berikut:

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

Jika $f(x,y)$ merupakan suatu fungsi dari gambar yang bernilai kontinu dan $s(x,y)$ adalah fungsi sampling 2-D dapat dituliskan dalam persamaan:

$$s(x,y) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \delta(x - j\Delta x, y - k\Delta y) \quad (2.1)$$

Persamaan 2.1 merupakan suatu fungsi *array* tak-hingga yang tersusun dari Δx dan Δy seperti ditunjukkan Gambar 2.4, Fungsi sampling ini disusun dalam bentuk *grid* berjarak Δx dan Δy sepanjang sumbu-x dan sumbu-y. Gambar hasil sampling dapat dipresentasikan dalam bentuk persamaan berikut ini [4]:

$$f_s(x, y) = f(x, y)s(x, y) \\ = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} f(j\Delta x, k\Delta y) \cdot \delta(x - j\Delta x, y - k\Delta y) \quad (2.2)$$

Fungsi $f_s(x, y)$ adalah gambar hasil sampling berupa *array* dari intensitas suatu gambar pada titik sampel $(j\Delta x, k\Delta y)$. Syarat dalam melakukan proses sampling berdasarkan *Nyquist Criterion* dimana besar frekuensi penyampling (sepanjang sumbu-y) harus dua kali lebih besar dari besar frekuensi yang terdapat pada sinyal yang di-sampling (sepanjang sumbu-x). Besarnya nilai Δx dan Δy harus dipilih sedemikian rupa agar gambar yang disampling memenuhi frekuensi *Nyquist Criterion*. Apabila nilai Δx dan Δy terlalu kecil maka gambar disebut *oversampled* sedangkan jika terlalu besar maka disebut *undersampled*. Bila gambar mengalami *undersampled* maka akan terjadi *over-lapping* yang menyebabkan *aliasing effect* yang meningkat seiring pengurangan resolusi gambar (Gambar 2.4) [5].



Gambar 2.4 *Image* disampling pada 256 x 256, 128 x 128, 64 x 64, 32 x 32, 16 x 16 *rectangular* sampling grid [5]

Proses pengubahan dari suatu sampel pixel analog menjadi suatu bilangan diskrit yang integer disebut sebagai proses *quantization*. *Quantization* meliputi pemberian sebuah nilai tertentu kepada setiap sampel [5]. *Gray-level quantization* merupakan perubahan citra berdasarkan tingkat amplitudo-nya yakni tingkat intensitas keabuan (*gray-level*) pada posisi xy . Untuk lebih jelasnya sapat dilihat pada Gambar 2.5:



Gambar 2.5 *Gray-level Quantization* [5]

Suatu proses digitalisasi memerlukan suatu keputusan tentang nilai N , M dan jumlah nilai diskrit dari *gray-level* yang diperbolehkan pada setiap piksel. Biasanya digunakan perhitungan suatu citra yang di-sampling menjadi $N \times M$ array sebagai berikut [4]:

$$N = 2^n, \quad M = 2^k \quad (2.3)$$

$$G = 2^m \quad (2.4)$$

G menunjukkan jumlah *gray-level*. Tingkat *gray-level* yang ada menentukan *storage* bit yang dibutuhkan untuk menyimpan citra digital. Menggunakan persamaan 2.4 dan 2.5, maka jumlah bit yang dibutuhkan, b , untuk menyimpan citra adalah:

$$b = N \times M \times m \quad (2.5)$$

jika $N=M$ maka

$$b = N^2 m \quad (2.6)$$

Sebagai contoh, 512×512 image dengan 256 *gray-level* ($m=8$) akan membutuhkan 262,144 8-bits penyimpanan data.

2.2.2 Normalisasi Data

Data yang merupakan citra foto darah yang berasal dari kamera digital harus dinormalisasi terlebih dahulu. Hal ini berguna untuk menyesuaikan data agar dapat diproses melalui berbagai proses selanjutnya dan untuk mendapatkan data yang baik.

Berikut ini adalah beberapa proses yang terjadi dalam normalisasi data:

2.2.2.1 *Intensity Adjustment*

Data berupa tetesan darah yang telah direaksikan dengan cairan reagen memiliki tingkat gradasi warna yang berbeda. Hal ini disebabkan komposisi banyaknya darah dan reagen yang tercampur berbeda-beda, walaupun hasil dari reaksi kimia tetap, akan tetapi data yang berupa *image* digital akan lebih mudah untuk diproses apabila dilakukan *intensity adjustment* terlebih dahulu.

Maka untuk mendapatkan data yang baik dibutuhkan suatu pengaturan intensitas sampai level tertentu agar data dapat diolah dengan baik sehingga meningkatkan hasil proses identifikasi.

Pengaturan nilai intensitas ini disebut dengan *intensity adjustment*. Nilai intensitas yang diatur merupakan intensitas *gray-level* dari *image* yang telah diubah dari RGB ke *gray-level*

2.2.2.2 *Cropping*

Istilah *Cropping* dalam hal ini dapat diartikan sebagai memotong suatu *image* pada daerah tertentu untuk diambil dan diolah. Tidak semua daerah dari *image* foto yang diolah sebagai data akan tetapi hanya daerah tertentu yang memiliki *image* darah. Hal ini dilakukan untuk mendapatkan data yang tepat dan berukuran kecil sehingga memudahkan dalam proses komputasi data.

2.2.2.3 *Levelling dan Reshaping*

Leveling merupakan proses pemberian level nilai *gray-level* *image* dari suatu *array* atau matriks ke suatu nilai diskrit tertentu. Sedangkan *reshaping* adalah membentuk matriks dengan ukuran tertentu menjadi matriks dengan ukuran lain yang memiliki jumlah *array* yang sama, misalnya matriks $N \times M$ menjadi matriks $1 \times NM$.

2.2.2.4 *Edge Detection*

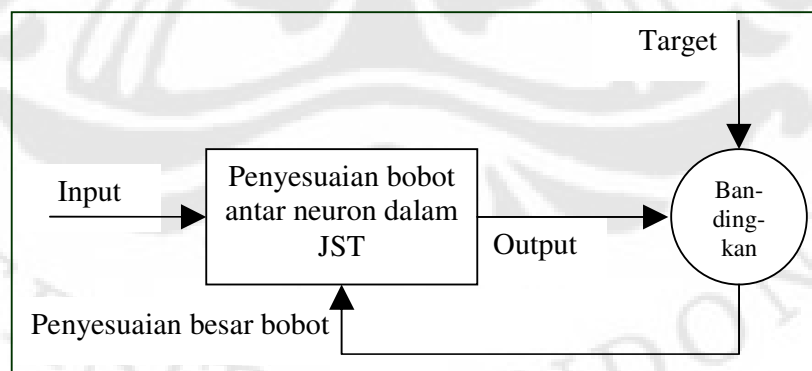
Edge detection adalah proses untuk memperbaiki citra yang berguna untuk melihat perbedaan intensitas warna dalam *image* yang dikenali sebagai tepian, proses ini berfungsi untuk melihat ada atau tidaknya penggumpalan yang terjadi pada *image* darah

2.2.3 EKSTRAKSI

Data yang telah dinormalisasi selanjutnya akan di-ekstraksi untuk mengambil bagian yang akan diolah lebih lanjut dengan mengambil parameter karakteristiknya berdasarkan matriks baru dengan bentuk $M.N \times 1$ dengan fungsi *reshaping*. Selanjutnya rata-rata dari matriks didapatkan dari matriks yang telah diubah bentuknya menjadi 1 baris. Untuk mencari rata-rata dari matriks sampel digunakan fungsi *mean*.

2.3 JARINGAN SYARAF TIRUAN

Jaringan Syaraf Tiruan atau sering disebut *neural network* adalah sistem komputasi dimana arsitektur dan operasi diilhami dari pengetahuan tentang sel syaraf biologis di dalam otak, yang merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba menstimulasi proses pembelajaran dari otak manusia tersebut [6]. Oleh karena itu, JST memiliki kemampuan untuk melakukan proses pembelajaran, menyimpan memori, dan melakukan generalisir dari pola pelatihan maupun dari data. Pada umumnya jaringan syaraf tiruan dilatih (*trained*) agar input mengarah ke output target yang spesifik. Jadi jaringan dilatih terus menerus hingga mencapai kondisi dimana input sesuai dengan target yang telah ditentukan. Pelatihan dimana setiap input diasosiasikan dengan target yang telah ditentukan disebut pelatihan terarah (*Supervised learning*) seperti pada Gambar 2.6:



Gambar 2.6 Diagram blok Jaringan Syaraf Tiruan

Secara umum arsitektur Jaringan Syaraf Tiruan memiliki empat komponen utama:

1. Neuron

Neuron merupakan tempat pemrosesan informasi.

2. Layer

Layer merupakan lapisan dalam aringan. Lapisan pertama adalah input dan lapisan yang terakhir adalah output.

3. Bobot

Setiap sambungan penghubung antar layer memiliki bobot yang bersesuaian.

Bobot ini akan digunakan untuk mengandakan/mengalikan isyarat yang dikirim melaluinya. [7].

4. Fungsi aktivasi.

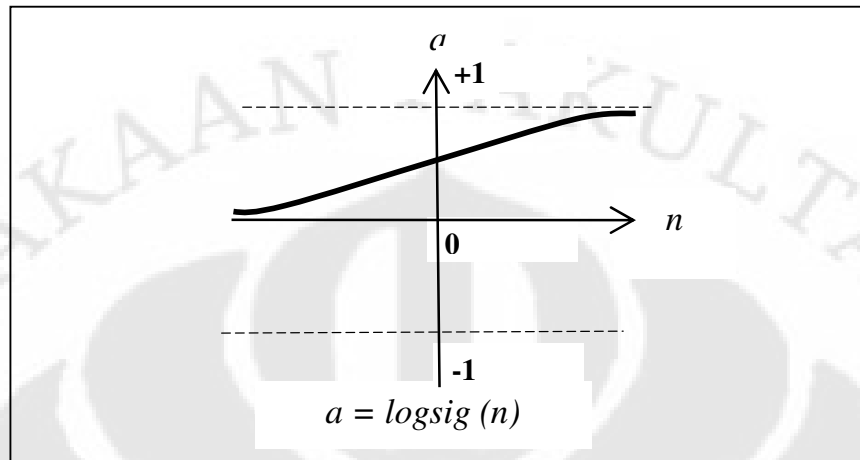
Setiap neuron akan menerapkan fungsi aktivasi terhadap isyarat hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan isyarat keluarannya. [7].

Saat ini yang sering digunakan pada jaringan syaraf tiruan adalah pelatihan terarah (*Supervised learning*). Prinsip dasar pelatihan ini adalah membandingkan antara input dengan target untuk menyesuaikan nilai bobot pada setiap kali *training*, hal ini dilakukan agar pada pelatihan selanjutnya nilai bobot yang didapat akan semakin mendekati dan akhirnya mencapai kondisi dimana input telah sesuai dengan target output. Selain pelatihan terarah, ada juga pelatihan tidak terarah dimana tidak ditentukan target outputnya. Metode pelatihan tidak terarah ini sekarang jarang digunakan.

Backpropagation merupakan salah satu algoritma pelatihan terarah. Algoritma *backpropagation* biasa digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada lapisan tersembunyinya. Algoritma *Backpropagation* menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* tersebut, tahap perambatan maju (*forward propagation*) harus dilakukan terlebih dahulu. Pada perambatan maju *neuron-neuron* akan diaktifkan dengan menggunakan fungsi aktivasi yang dapat didiferensiasikan [7], seperti:

1. Sigmoid

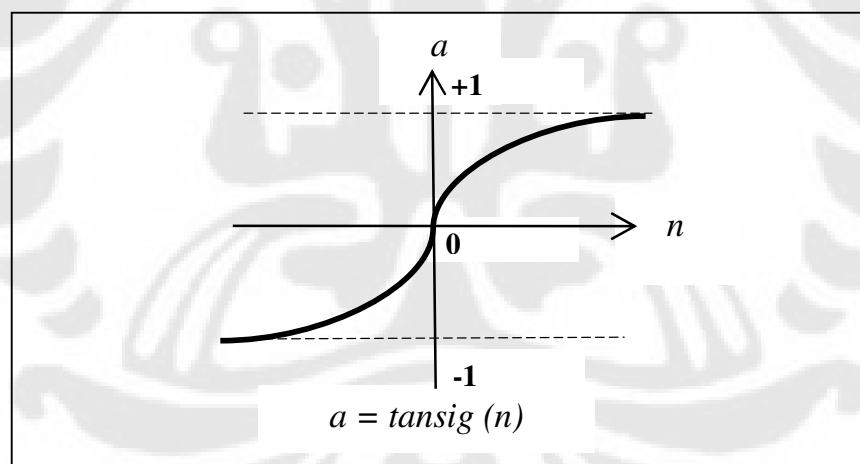
$$y = f(x) = \frac{1}{1 + e^{-\sigma x}} \quad (2.7)$$



Gambar 2.7 Fungsi aktivasi sigmoid biner

2. Tansig

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



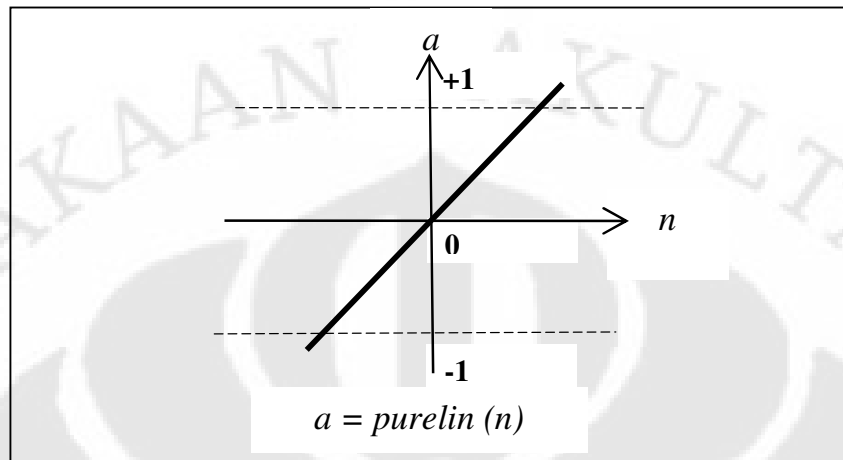
Gambar 2.8 Fungsi aktivasi sigmoid bipolar

dengan

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

3. Purelin

$$y = f(x) = x \quad (2.10)$$



Gambar 2.9 Fungsi aktivasi linear

Secara umum metode pembelajaran backpropagation ini dapat dituliskan sebagai berikut:

1. Menginiasialisasi bobot dan bias.
2. Menetapkan maksimum epoch, *target error*, dan *learning rate* (α).
3. Selama (epoch < maksimum epoch) dan (error > target error) maka proses akan diteruskan.
4. Untuk setiap pasangan data training dilakukan langkah berikut (epoch = epoch + 1):

Feedforward :

- a) Setiap unit masukan menerima data x_i dan mengirim sinyal masukan ke semua unit pada lapisan berikutnya (lapisan tersembunyi).
- b) Tiap – tiap unit pada suatu lapisan tersembunyi h_q menjumlahkan sinyal masukan terbobot dan menggunakan fungsi aktivasi untuk menghitung sinyal keluarannya.

$$z_in_j = \left(\sum_{i=1}^n x_i v_{ij} \right) + v_{0j} \quad (2.11)$$

$$z_j = f(z_in_j) \quad (2.12)$$

Kemudian sinyal keluaran h_q akan dikirim ke semua unit pada lapisan berikutnya

- c) Setiap keluaran unit y_j menjumlahkan sinyal masukan terbobot dan menggunakan fungsi aktivasi untuk menghitung sinyal keluarannya.

$$y_{in_k} = \left(\sum_{j=1}^n z_j w_{jk} \right) + w_{ok} \quad (2.13)$$

$$y_k = f(y_{in_k}) \quad (2.14)$$

Backpropagation :

- d) Setiap keluaran unit y_j menerima target pola masukan pembelajaran kemudian dihitung informasi kesalahannya ($e=(t_j - y_j)$):

$$\delta_k = (t_k - y_k) f'(y_{in_k}) \quad (2.15)$$

Hitung koreksi bobot untuk memperbaiki nilai w_{jk} :

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (2.16)$$

Hitung koreksi bobot bias w_{ok}

$$\Delta w_{ok} = \alpha \delta_k \quad (2.17)$$

- e) Setiap unit pada lapisan tersembunyi h_q menjumlahkan delta masukannya (dari unit – unit yang berada di lapisan berikutnya) :

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.18)$$

Kemudian dihitung informasi kesalahannya :

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (2.19)$$

Hitung koreksi bobot untuk memperbaiki nilai v_{ij} :

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (2.20)$$

Hitung koreksi bias untuk memperbaiki nilai v_{oj} :

$$\Delta v_{oj} = \alpha \delta_j \quad (2.36)$$

- f) Setiap keluaran unit memperbaiki bias dan bobotnya.

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (2.21)$$

$$w_{ok}(\text{baru}) = w_{ok}(\text{lama}) + \Delta w_{ok} \quad (2.22)$$

- g) Setiap unit pada lapisan tersembunyi memperbaiki bias dan bobotnya.

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (2.23)$$

$$v_{oj}(\text{baru}) = v_{oj}(\text{lama}) + \Delta v_{oj} \quad (2.24)$$

5. Apabila kondisi (iterasi < iterasi maksimum) atau (kesalahan \geq target kesalahan), tercapai maka pelatihan dihentikan. (Proses iterasinya lebih sedikit) maka dapat digunakan metode momentum pada perhitungan *backpropagation*.

Untuk memperoleh algoritma pembelajaran *backpropagation* yang lebih cepat (proses iterasinya lebih sedikit) maka pelatihan dapat dipilih metode optimasi yang disesuaikan dengan kemampuan sistem yang ada. Hal ini dikarenakan beberapa metode optimasi memiliki fungsi komputasi yang cukup kompleks. Terdapat beberapa alternatif optimasi yang dapat digunakan antara lain, *Gradient Descent*, *Conjugate Gradient*, *Golden Section*, *Quasi newton*, dan *Levenberg-Marquadt*.

BAB III

METODOLOGI PENELITIAN

Pada bab ini akan dibahas mengenai tahap-tahap yang dilakukan dalam melakukan proses identifikasi golongan darah. Ada tiga macam tahap besar dalam pembuatan skripsi ini, yaitu: Sampling, pelatihan JST dan proses pengenalan obyek

3.1 DATA DAN PERALATAN PENELITIAN

Data yang digunakan dalam penelitian terdiri dari sekumpulan citra untuk pelatihan data sejumlah 83 citra yang terdiri atas 20 citra untuk masing-masing objek. Sedangkan untuk pengujian citra golongan darah sejumlah 10 citra untuk masing-masing golongan darah.

Pengambilan data dilakukan dengan mengambil citra darah yang telah direaksikan dengan reagen menggunakan kamera digital.

Perangkat yang digunakan pada proses pelatihan dan simulasi identifikasi tipe golongan darah kertas ini adalah sebagai berikut:

Software	: Perangkat lunak pengolah data numeris
Kamera digital	: 8 megapixels
Reagen	: Anti-A dan Anti-B
Prosesor	: Intel™ Pentium 4 1.8 GHz
Memory	: 1024 MB

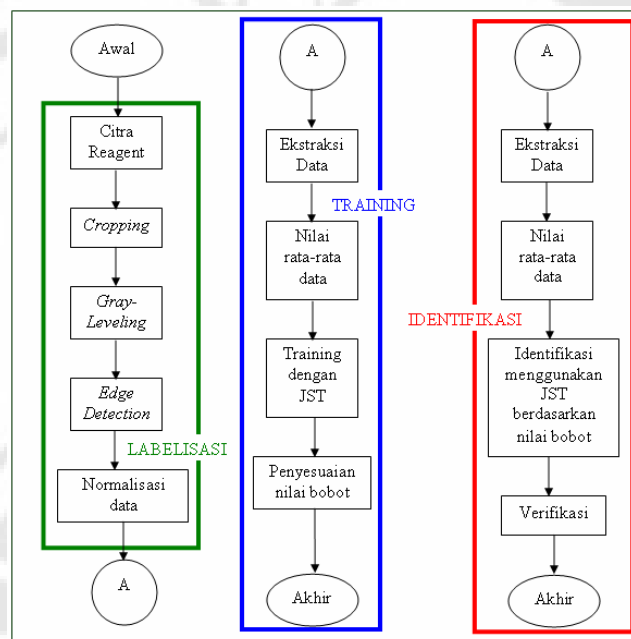
3.2 LANGKAH Pengerjaan

Terdapat tiga tahap pengerjaan utama dalam perangkat lunak, yaitu:

1. Tahap pembentukan basis data sebagai masukan JST
2. Tahap pelatihan set data yang juga menghasilkan hasil akhir berupa bobot lapisan dan bobot bias untuk tiap unit lapisan tersembunyi dan keluaran, yang nantinya disimpan *file* parameter untuk digunakan pada simulasi identifikasi
3. Simulasi identifikasi, mendeteksi citra tipe golongan darah setelah pelatihan selesai dilaksanakan.

Dalam proses pembentukan *database* terdiri dari dua tahap, terdiri atas: labelisasi (tahapan mencuplik dan kuantisasi data, normalisasi data, ekstraksi fitur citra, nilai rata-rata fitur citra). Sedangkan pada pelatihan digunakan metode jaringan syaraf tiruan dengan algoritma *backpropagation*.

Diagram aliran keseluruhan proses, mulai dari labelisasi, pelatihan JST, dan simulasi identifikasi dapat dilihat pada Gambar 3.1 berikut ini. Pada proses pelatihan JST dan simulasi, keduanya akan melalui tahapan pra-proses. Perbedaannya adalah pada pelatihan JST seluruh data citra akan diproses secara simultan (LABELISASI) untuk dilatih pada JST yang akan menghasilkan nilai bobot. Sedangkan pada simulasi identifikasi hanya dilakukan tahapan pra-proses citra yang bersangkutan diteruskan dengan proses *feed-forward* dari JST menggunakan bobot hasil akhir pelatihan. Blok diagram alir proses ini dapat dilihat pada Gambar 3.1 berikut ini.



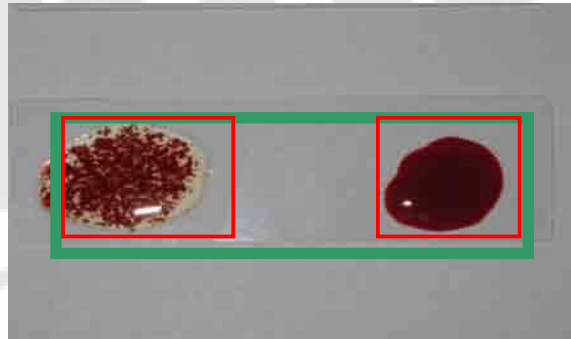
Gambar 3. 1 Diagram alir pembuatan labelisasi (kotak hijau), training JST (kotak biru) dan identifikasi (kotak merah)

3.3 PRA-PROSES

Sebelum data dimasukkan ke dalam *database* maka data mentah berupa citra darah kertas harus mengalami proses awal (pra proses) atau persiapan basis data untuk digunakan secara simultan dalam pelatihan JST. Tahapan proses ini dapat disebut sebagai LABELISASI yang terdiri dari beberapa tahapan, yaitu:

3.3.1 Pemotongan

Proses pemotongan data adalah proses dimana data yang memiliki ciri yang unik akan diisolasi dari citra mentah yang diperoleh dari kamera digital, citra mentah akan dipotong sebanyak dua kali, pemotongan pertama bertujuan untuk memisahkan citra yang akan diteliti dari latar belakan citra tersebut, sedangkan pemotongan kedua bertujuan untuk meneliti penggumpalan citra darah pada sebelah kiri dan kanan adapun daerah dari citra yang akan diproses dengan pemotongan adalah (Gambar3.2)):



Gambar 3. 2 Daerah pemotongan citra

Warna hijau menunjukkan pemotongan pertama, sedangkan warna merah menunjukkan pemotongan kedua.

3.3.2 Konversi Citra RGB Ke Citra Hitam-Putih Dan Kompresi Data

Mengubah citra dari RGB menjadi hitam-putih dengan 256 tingkat keabu-abuan. Hal ini dapat dilakukan dengan cara mengeliminasi informasi *hue* (gradasi warna) dan *saturation* (kejenuhan warna) dengan tetap mempertahankan tingkat kecerahan (*luminance*). Sehingga dihasilkan matrik $M \times N$ warna hitam-putih dalam 256 tingkat keabu-abuan. Hal ini dilakukan karena dengan hanya citra hitam-putih, proses perhitungan menjadi jauh lebih sederhana.

Hasil citra hitam-putih adalah sebagai berikut (Gambar 3.3):



Gambar 3.3 Hasil *grayscale* dengan intensitas 256 bit

Setelah itu dilakukan ekstraksi ciri melalui fungsi pengaturan resolusi citra, dimana dengan fungsi ini rata-rata nilai citra dapat diambil sekaligus mengubah rasio skala panjang dan lebar (ukuran) dari citra menjadi 50x50 pixel. Citra hasil proses ini akan diekstraksi cirinya berupa nilai rata-rata matriks berukuran 5x5 *pixel* (M x N) yang akan dibentuk ulang menjadi matriks kolom berukuran 25x1 (M.N x 1) yang menjadi masukan bagi pelatihan JST.

3.4 LABELISASI

Tahap pertama dari pembentukan basis data adalah labelisasi. Pembentukan basis data terbagi menjadi dua bagian, terdiri atas data citra darah yang menggumpal dan data citra darah yang tidak menggumpal.

Untuk data darah yang menggumpal dan tidak menggumpal dikelompokkan menjadi dua bagian yang berbeda sebelum dimasukkan dalam pelatihan JST, kombinasi dari citra menggumpal dan tidak menggumpal akan menghasilkan empat jenis tipe darah yang berbeda, berikut ini tabel III.1 yaitu kombinasi dari jenis golongan darah pada citra darah:

Tabel III. 1 Kombinasi citra darah

Bagian sebelah kanan citra darah	Bagian sebelah kiri citra darah	Jenis golongan darah
Menggumpal	Tidak Menggumpal	A
Tidak Menggumpal	Menggumpal	B
Menggumpal	Menggumpal	AB
Tidak Menggumpal	Tidak Menggumpal	O

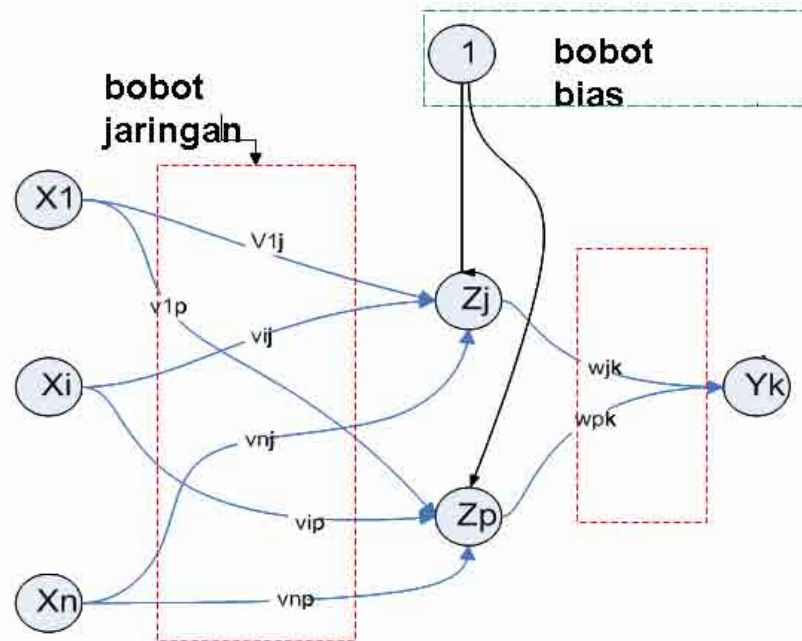
3.5 PELATIHAN JST

Pada bagian ini akan dilakukan proses pelatihan JST untuk mengubah bobot-bobot koneksi pada JST karakteristik dari jaringan syaraf tiruan. Hal ini dilakukan dengan cara memasang masukan dan target yang dituju (*supervised learning*) menggunakan 25 nilai rata-rata sampel dari masing-masing citra.

Algoritma JST yang digunakan adalah *backpropagation*. Model *backpropagation* merupakan metode pelatihan yang terbimbing. Jaringan ini terdiri dari satu lapisan masukan dengan 25 unit neuron, satu lapisan tersembunyi, dan satu lapisan keluaran dengan 4 unit keluaran. Fungsi aktivasi yang digunakan pada lapisan

tersembunyi adalah *logsig*, sedangkan fungsi aktivasi pada lapisan keluaran adalah *purelin* (Gambar 3.4).

Untuk lebih jelasnya dapat dilihat pada gambar 3.10 berikut ini yang menunjukkan topologi JST *backpropagation* yang digunakan beserta keterangan parameter-parameter JST yang dipakai.



Gambar 3. 4 Topologi JST *backpropagation* yang digunakan pada simulasi

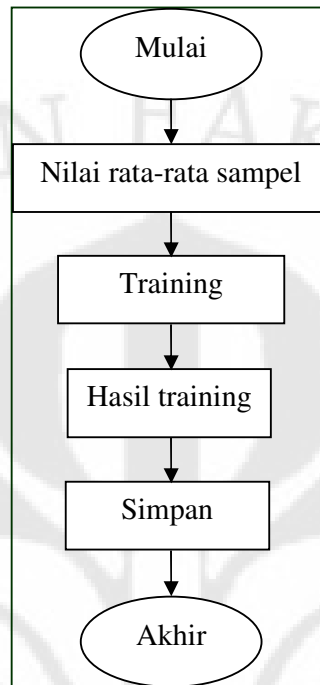
Parameter-parameter dari topologi JST *backpropagation* yang digunakan adalah sebagai berikut:

1. Lapisan masukan : 25 unit neuron (matriks 25x1)
2. Lapisan tersembunyi : 1 lapisan tersembunyi dengan 10 unit neuron
3. Bobot jaringan : (matriks 10 x 25) bobot jaringan pada 10 unit neuron lapisan tersembunyi
4. Bobot bias : terdapat 1 bobot bias, yaitu pada lapisan tersembunyi (matriks 10 x 1)
5. Lapisan keluaran : terdapat 1 lapisan keluaran dengan 1 unit neuron
6. Fungsi aktivasi lapisan tersembunyi tansig, fungsi aktivasi lapisan keluaran purelin.
7. Pelatihan dilakukan sebanyak 83 kali dengan metode optimasi yang digunakan *Levenberg Marquadt*.

Berikut tahapan proses yang terjadi pada pelatihan JST *backpropagation* :

1. Memasukkan 25×1 nilai rata-rata matriks dari tiap citra dengan target yang dituju.
2. Menentukan target jaringan. Untuk jenis citra darah yang menggumpal bernilai nol sedangkan untuk citra darah yang tidak menggumpal bernilai satu.
3. Membangun jaringan dengan menentukan beberapa parameter yakni rentang nilai masukan, jenis fungsi aktivasi, menentukan metode optimasi, target error, serta jumlah pelatihan.
4. Menentukan tingkat keluaran jaringan yang diperbolehkan.
5. Menentukan jumlah iterasi maksimal.
6. Melakukan pelatihan JST, melalui propagasi balik untuk mengubah bobot-bobot koneksi pada jaringan.
7. Pembelajaran berhenti apabila kesalahan pelatihan telah mencapai nilai minimum yang diinginkan atau jumlah iterasi telah melewati batas maksimal.
8. Menyimpan data jaringan (bobot jaringan dan bobot bias) ke dalam suatu *file* untuk digunakan pada saat identifikasi.
9. Pada tahap simulasi, aliran data arah maju (*feed-forward*) dilakukan berdasarkan bobot akhir hasil pelatihan.

Pada Gambar 3.5 berikut ini diagram alir yang menggambarkan proses training JST:



Gambar 3.5 Diagram alir proses pelatihan jaringan syaraf tiruan

Sebagai bahan perbandingan, akan diuji tingkat akurasi yang didapatkan bilamana :

1. Masukan dilakukan dengan data berukuran besar (10 x 10 pixel)
2. Jumlah pelatihan yang divariasikan.
3. Fungsi aktivasi divariasikan.
4. Metode optimasi yang digunakan (gradient descent dibandingkan *Levenberg-Marquadt*).

3.6 SIMULASI IDENTIFIKASI

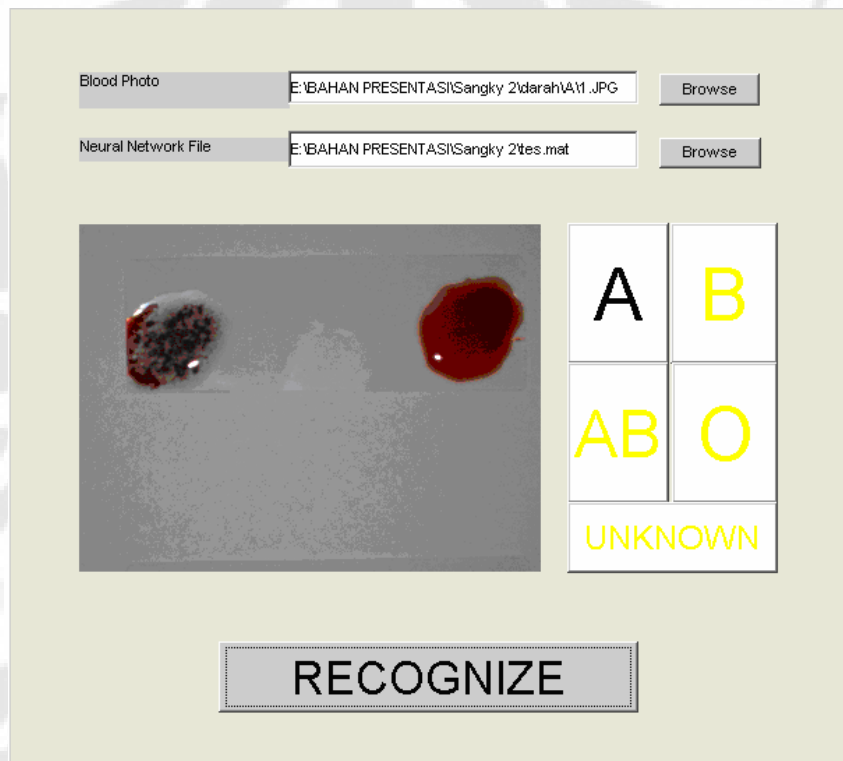
Proses identifikasi merupakan tahap dimana sistem akan berusaha mengenali pola-pola unik dari citra darah yang menjadi masukan, dan akan mengklasifikasikan citra tersebut ke dalam golongan darah yang dimaksud. Dalam hal ini sistem akan berfungsi sebagai alat identifikasi citra golongan darah.

Proses identifikasi pada jaringan syaraf tiruan dilakukan melalui proses matematis yang sama dengan operasi arah maju (*feed-forward*), yaitu mengalikan 25 neuron masukan terhadap bobot jaringan dan ditambah dengan bobot bias untuk

masing-masing unit neuron tersembunyi dan keluaran. Nilai bobot diperoleh pada proses pelatihan sebelumnya. Jadi pada proses klasifikasi tidak terjadi perubahan atau penyesuaian bobot.

Proses klasifikasi inilah yang dijadikan dasar dalam menentukan tipe golongan darah. Nilai output yang benar akan bernilai 1 selepas dari fungsi aktivasi tansig pada lapisan tersembunyi dan output salah akan bernilai 0. Nilai 1 dan 0 inilah yang menentukan pemilihan target keluaran diantara empat klasifikasi target yang ada.

Adapun tampilan dari proses identifikasi adalah sebagai berikut (Gambar 3.6):



Gambar 3. 6 Tampilan perangkat lunak modul identifikasi

BAB IV

HASIL UJI COBA DAN ANALISIS

4.1 HASIL UJI COBA

Untuk mengukur nilai evaluasi dari sistem perangkat lunak pengenalan tipe golongan darah manusia dengan menggunakan JST, maka sistem akan diuji menggunakan citra darah yang tidak termasuk dalam proses pelatihan JST sebelumnya. Jumlah pengujian tiap objek akan dilakukan minimal sepuluh kali variasi untuk masing-masing jenis darah. Total set data yang disediakan dalam proses pengujian ini adalah sebanyak 40 citra.

Percobaan simulasi identifikasi tipe golongan darah dilakukan dengan menggunakan perangkat lunak pengolah data numeris

Beberapa simulasi dilakukan untuk mencari konfigurasi terbaik dengan tingkat akurasi tertinggi dan waktu operasi tersingkat dengan memvariasikan struktur pembentuk JST antara lain:

1. Besar masukan JST
2. jenis *hidden layer*
3. Nilai keluaran JST
4. Jumlah pelatihan JST

Tingkat akurasi dihitung dengan membandingkan jumlah identifikasi yang benar dari masing-masing objek terhadap total objek masing-masing. Pengujian tidak mengikutkan data pelatihan.

Hasil akhir yang didapatkan oleh perangkat lunak dengan konfigurasi optimum dapat dilihat pada Tabel IV.1, menunjukkan persentase hasil uji coba simulasi untuk masing-masing tipe golongan darah dengan variasi akurasi yang berbeda-beda.

Tabel IV. 1 Tabel hasil uji coba identifikasi

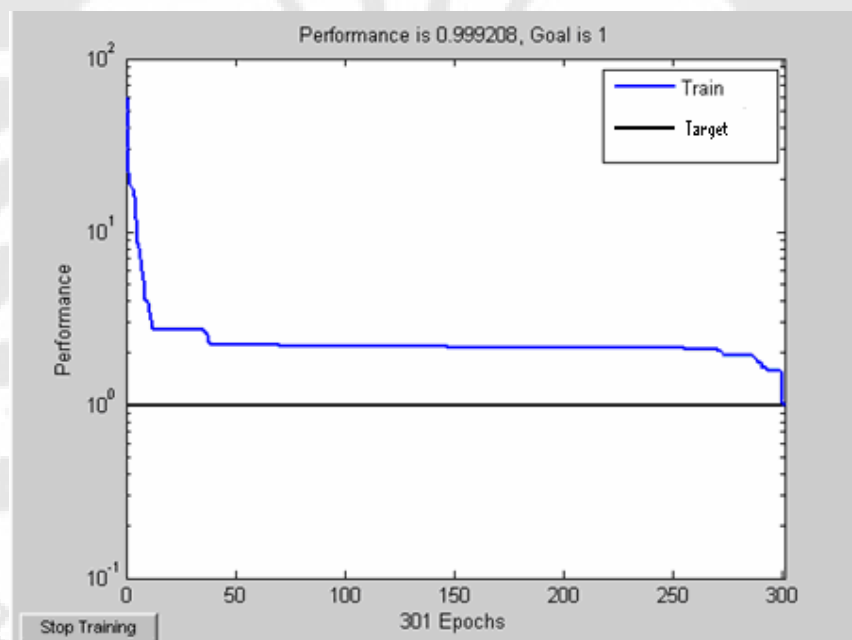
Tipe golongan darah	Akurasi
A	80%
B	90%
AB	40%
O	90%

Pada tabel di atas dapat kita lihat konfigurasi optimum dari parameter-parameter JST menghasilkan tingkat akurasi sebesar 80% untuk golongan darah A, 90% untuk golongan darah B, 40% untuk golongan darah AB dan 90% untuk golongan darah O.

Pada golongan darah AB, akurasi yang didapat sangat kecil jika dibandingkan dengan golongan darah lainnya, hal ini disebabkan karena tingkat keabuan yang hanya terbagi menjadi dua level, dan sebagian besar data gambar darah yang didapat untuk identifikasi golongan darah AB menunjukkan bahwa darah telah kering sebelum difoto

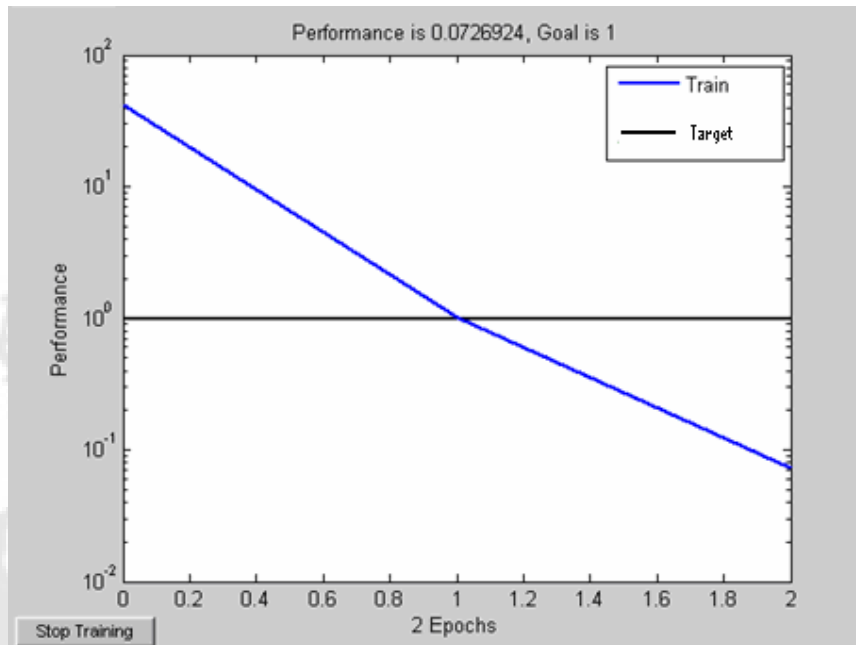
4.2 PENGARUH UKURAN MATRIKS MASUKAN PADA HASIL IDENTIFIKASI

Pada hasil simulasi yang menggunakan matriks masukan sebesar 10x10, hasil identifikasi menjadi lebih buruk jika dibandingkan dengan masukan sebesar 5x5 (Gambar 4.1), berikut ini merupakan hasil grafik dari pelatihan JST dengan masukan sebesar 10x10:



Gambar 4.1 Grafik pelatihan JST menggunakan matriks 10x10 dengan unjuk kerja 10^0

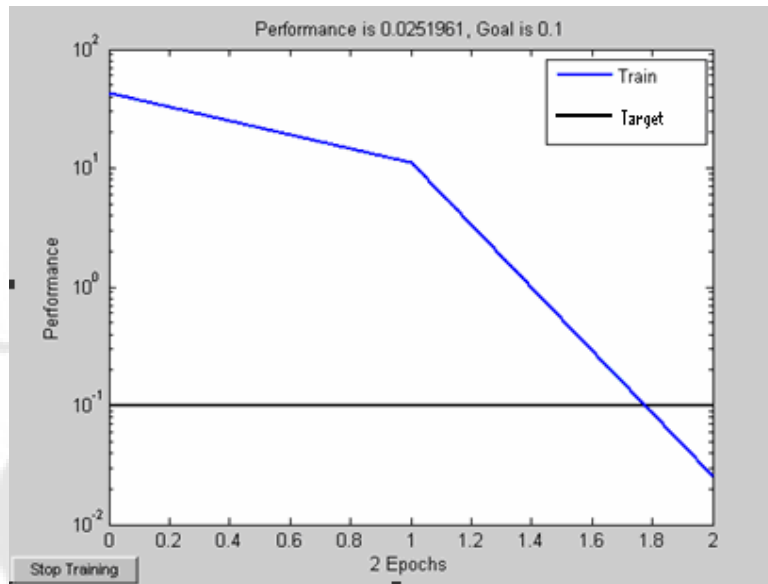
Jika dibandingkan dengan hasil training pada masukan matriks 5x5, maka besarnya jumlah iterasi dapat dibandingkan bahwa masukan matriks 5x5 memiliki jumlah iterasi yang lebih sedikit, yaitu sebesar 2 epoch jika dibandingkan dengan masukan 10x10, yaitu sebesar 301 epoch. Gambar 4.2 berikut ini adalah hasil simulasi dari pelatihan masukan matriks 5x5:



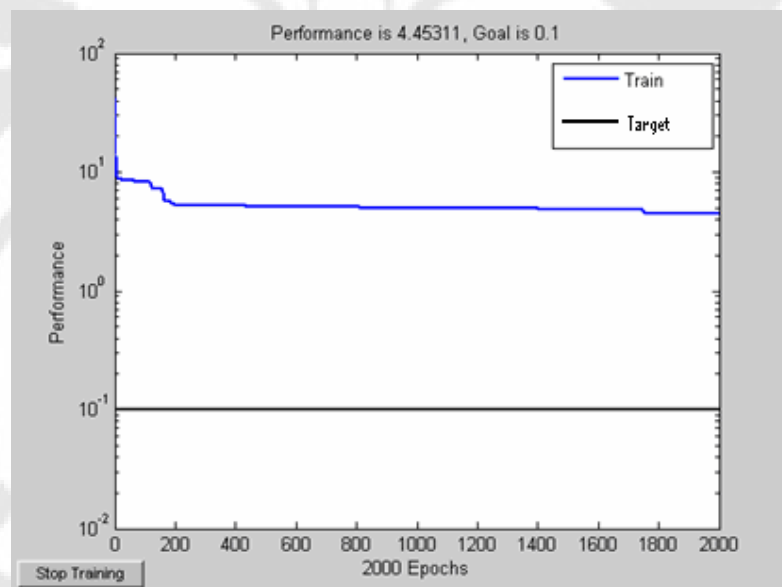
Gambar 4.2 Grafik pelatihan JST menggunakan matriks 5x5 dengan unjuk kerja 10^0

Pada dua gambar diatas dapat disimpulkan bahwa ukuran matriks masukan akan mempengaruhi jumlah iterasi terhadap unjuk kerja pelatihan JST, semakin sedikit jumlah masukan, maka jumlah iterasi akan semakin cepat, hal ini dikarenakan metode JST memiliki ciri dimana semakin detail ciri citra maka akan semakin buruk unjuk kerja JST yang ada.

Dua gambar grafik diatas merepresentasikan unjuk kerja sebesar 10^0 , jika kita mencoba meningkatkan unjuk kerja menjadi 10^{-1} , maka matriks 10x10 tidak dapat memenuhi performansi tersebut, sedangkan matriks 5x5 masih dapat memenuhi unjuk kerja tersebut. Dapat dilihat pada Gambar 4.3 dan Gambar 4.4 berikut ini:



Gambar 4. 3. Grafik pelatihan JST menggunakan matriks 5x5 dengan unjuk kerja 10^{-1}



Gambar 4. 4. Grafik pelatihan JST menggunakan matriks 10x10 dengan unjuk kerja 10^{-1}

Jika ditampilkan dalam bentuk tabel maka, hasil yang didapat dari hasil pelatihan simulasi adalah sebagai berikut pada Tabel IV.2:

Tabel IV. 2 Pengaruh besar matriks masukan

Jumlah Data pelatihan	\sum Iterasi 100	\sum Iterasi 10-1
5x5 pixel	2	2
10x10 pixel	301	-

Selain jumlah iterasi yang berbeda, tingkat akurasi pada citra juga berpengaruh terhadap ukuran matriks, akurasi matriks akan menurun dengan bertambahnya jumlah masukan matriks seperti terlihat pada Tabel IV.3:

Tabel IV. 3 Tingkat akurasi dengan 10x10 pixel citra darah

Type golongan darah	Akurasi
A	40%
B	50%
AB	30%
O	90%

Tingkat akurasi yang buruk disebabkan ciri citra yang diproses tidak unik, apabila data terlalu besar dengan jumlah pelatihan data yang sedikit maka terjadi perulangan data yang sama pada objek yang berbeda. Karena dengan data yang detail, satu objek dapat memiliki nilai ciri yang sama dengan objek lainnya.

4.3 PENGARUH METODE OPTIMASI

Pada simulasi, struktur JST dioptimasi dengan metode *Levenberg-Marquardt*. Keistimewaan dari metode ini adalah mampu meminimalkan fungsi error yang didapat sekaligus menambah nilai bobot sedikit demi sedikit dan tetap mempertahankan aproksimasi yang linear. Sebagai bahan perbandingan dilakukan pula optimasi menggunakan *gradient descent*. Seperti diketahui basis matematika dari operasi *backpropagation* pada JST adalah optimasi *gradient descent / steepest descent*.

Tabel IV.4 menunjukkan hasil perbandingan antara optimasi *Levenberg-Marquardt* dan *Gradient descent* (GD) dari segi jumlah iterasi.

Tabel IV. 4 Perbandingan *Levenberg-Marquardt* dan *Gradient descent* dengan input 5x5

Metode	\sum Iterasi
<i>Levenberg-Marquardt</i>	2
<i>Gradient descent</i>	7782

Terlihat dengan *Levenberg-Marquardt* (LM), jumlah iterasi LM memerlukan lebih sedikit iterasi yaitu 2 epoh jika dibandingkan dengan GD yaitu 7782 epoh. Hal ini disebabkan kenaikan bobot pada GD sedikit demi sedikit (*steepest*) dibandingkan

LM yang lebih cepat dalam kenaikan bobotnya dengan tetap menjaga penurunan tingkat kesalahan yang linear konvergen dibandingkan dengan GD.

4.4 PENGARUH JUMLAH PELATIHAN

Jumlah pelatihan dalam sistem JST merupakan suatu aspek yang penting untuk diperhitungkan. Dalam proses pelatihan, data yang dimasukkan dalam proses ini harus bervariasi dan dengan jumlah yang banyak dalam variasinya, hal ini bertujuan untuk memperkaya sistem dengan berbagai macam kondisi keadaan data, semakin banyak variasinya maka akan semakin handal sistem JST tersebut dalam mengenali ciri suatu data, sehingga sistem dapat mengenali data dalam segala jenis keadaan yang berada dalam kondisi yang sebenarnya.

JST memiliki kemampuan dalam men-generalisasi suatu ciri yang telah dilatih, sehingga semakin banyak variasi ciri yang dilatih, maka akurasi sistem akan menjadi semakin baik. Tetapi apabila pola pelatihan JST diperbanyak dengan masukan yang berulang-ulang akan menyebabkan JST menjadi spesifik terhadap pola-pola berulang yang terakhir. Hal ini akan menyebabkan akurasi untuk data pelatihan baik tetapi buruk untuk data pengujian. Oleh karena itu pelatihan JST terus dilakukan dalam batasan akurasi sistem masih membaik. Apabila sebaliknya pelatihan harus dihentikan.

Dari hasil pelatihan pada Tabel IV.5 dengan pelatihan 10 set hingga 83 set dengan kenaikan 10 set dapat dilihat akurasinya untuk tiap golongan darah pada tabel IV.5 berikut:

Tabel IV. 5 Perbandingan jumlah pelatihan dengan akurasi tiap tipe golongan darah

Jumlah Pelatihan	Akurasi tipe golongan darah			
	A	B	AB	O
10	70%	60%	30%	90%
20	60%	60%	30%	100%
30	60%	70%	40%	100%
40	50%	90%	40%	100%
50	80%	90%	30%	90%
60	60%	80%	40%	90%
70	80%	90%	40%	90%
83	80%	90%	40%	90%

Jika pelatihan dilanjutkan untuk input lebih dari 70, maka akan terlihat akurasi sistem cenderung tetap disebabkan adanya batas jenuh JST sehingga JST terlalu spesifik terhadap pola-pola terakhir yang dilatih. Jadi dapat disimpulkan JST memiliki

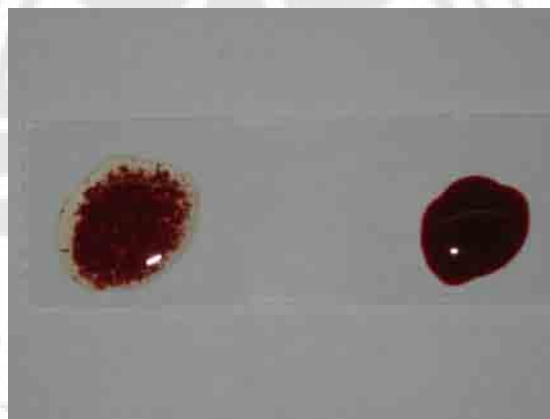
batasan jenuh pelatihan dimana dengan semakin banyak jumlah pelatihan, maka akurasi yang didapatkan juga semakin baik apabila JST masih mampu men-generalisir pola pelatihan masukan. Apabila pelatihan terus berulang-ulang JST akan menjadi spesifik terhadap pola masukan pelatihan terakhir. Hal ini berakibat memburuknya akurasi sistem identifikasi yang dibuat. Sedangkan pelatihan yang sedikit tidak menjamin performa JST yang baik walaupun kesalahan pengenalan pada pelatihan sangatlah kecil.

Pada proses identifikasi, terjadi kesalahan identifikasi pada data-data tertentu, untuk lebih jelasnya dapat dilihat pada tabel berikut:

Tabel IV. 6. Tabel identifikasi golongan darah A dengan input 5x5 dan metode optimasi LM

Data ke	Data dikenal sebagai tipe golongan			
	A	B	AB	O
1	ok			
2	ok			
3	ok			
4	ok			
5	ok			
6				salah
7	ok			
8	ok			
9				salah
10	ok			

Dapat dilihat pada Tabel IV.6 bahwa terjadi kesalahan pada data ke-6 dan data ke-10, gambar darah pada data ke-6 dan ke-9 dapat dilihat pada Gambar 4.5 dan 4.6 sebagai berikut:



Gambar 4. 5. data ke-6 dari golongan darah A

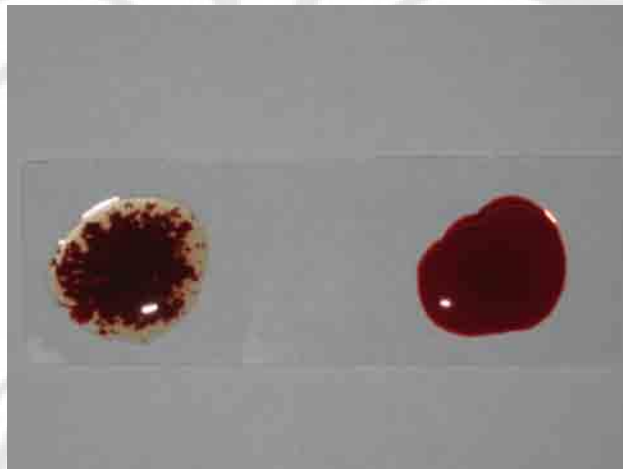
1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
0	1	1	1	1	0	1	1	1	1

Gambar 4. 6. Bentuk matriks dari golongan darah A data ke-6

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

Gambar 4. 7. bentuk matriks dari golongan darah O

Berdasarkan hasil pengolahan citra, maka dapat kita lihat bahwa nilai matriks untuk golongan darah A data ke-6, memiliki nilai yang mirip dengan nilai matriks untuk golongan darah O



Gambar 4. 8. data ke-9 dari golongan darah A

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

Gambar 4. 9. bentuk matriks dari golongan darah A data ke-9

Berdasarkan pengolahan citra pada data ke-9 kita dapat lihat bahwa matriks golongan darah A mirip dengan matriks golongan darah O.

Dari kedua gambar di atas dapat kita simpulkan, bahwa untuk data ke-6 dan data ke-9, gambar golongan darah A mirip dengan gambar golongan darah O, hal ini menyebabkan kesalahan pembacaan pada perangkat lunak yang mengakibatkan hasil dari pengidentifikasian golongan darah A terbaca menjadi golongan darah O.

Selanjutnya dapat kita lihat pada Tabel IV.7 untuk golongan darah B, Tabel IV.8 untuk golongan darah AB, dan Tabel IV.9 untuk golongan darah O.

Tabel IV. 7. Tabel identifikasi golongan darah B dengan input 5x5 dan metode optimasi LM

Data ke	Data dikenal sebagai tipe golongan			
	A	B	AB	O
1				salah
2	ok			
3	ok			
4	ok			
5	ok			
6	ok			
7	ok			
8	ok			
9	ok			
10	ok			



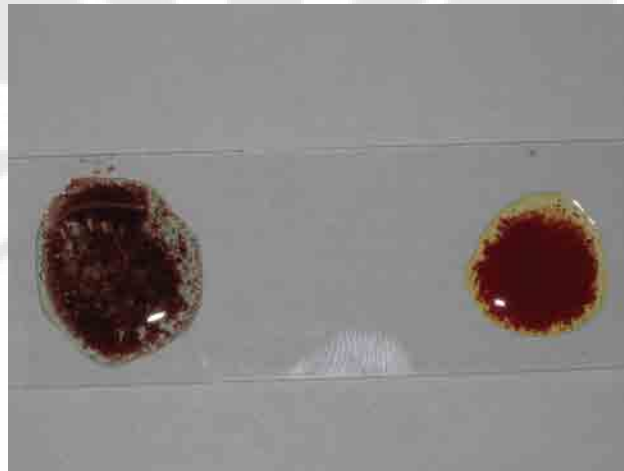
Gambar 4. 10. data ke-1 dari golongan darah B

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

Gambar 4. 11. bentuk matriks dari golongan darah B data ke-1

Tabel IV. 8. Tabel identifikasi golongan darah AB dengan input 5x5 dan metode optimasi LM

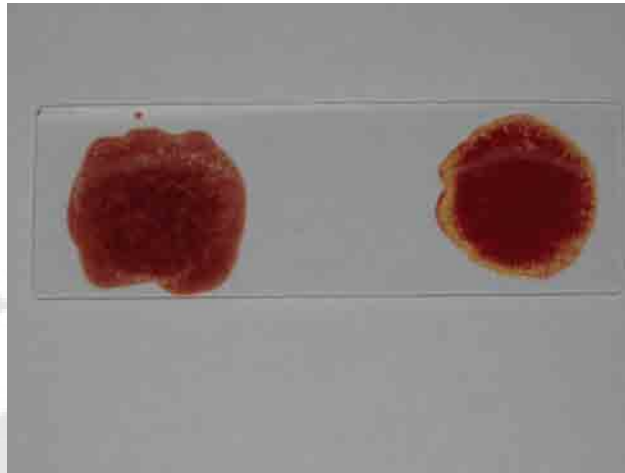
Data ke	Data dikenal sebagai tipe golongan			
	A	B	AB	O
1			ok	
2			ok	
3				salah
4			ok	
5				salah
6				salah
7				salah
8				salah
9			ok	
10	salah			



Gambar 4. 12. data ke-3 dari golongan darah AB

1	2	3	4	5	1	2	3	4	5
0	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

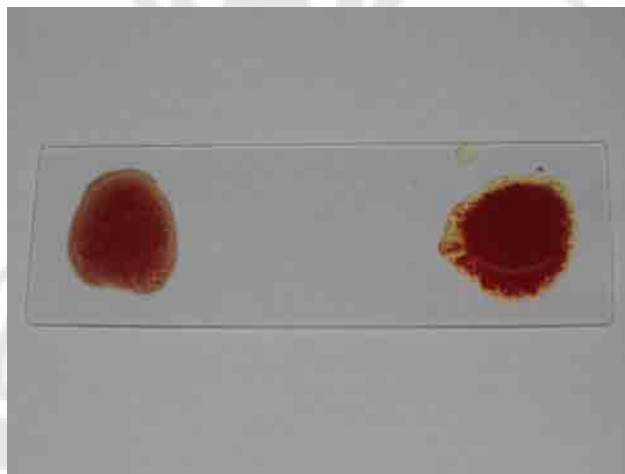
Gambar 4. 13. bentuk matriks dari golongan darah AB data ke-3



Gambar 4. 14. data ke-5 dari golongan darah AB

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

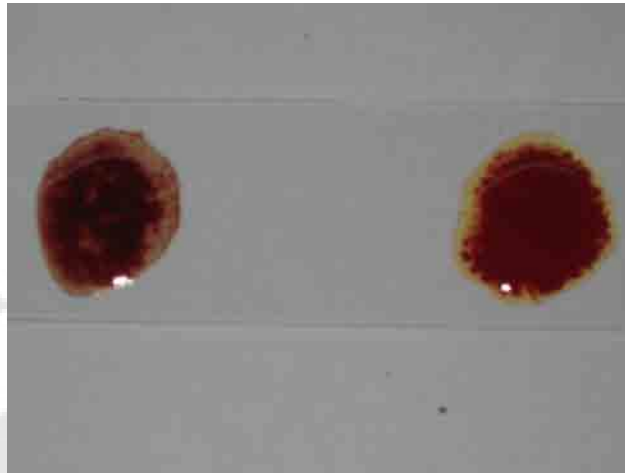
Gambar 4. 15. bentuk matriks dari golongan darah AB data ke-5



Gambar 4. 16. data ke-6 dari golongan darah AB

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

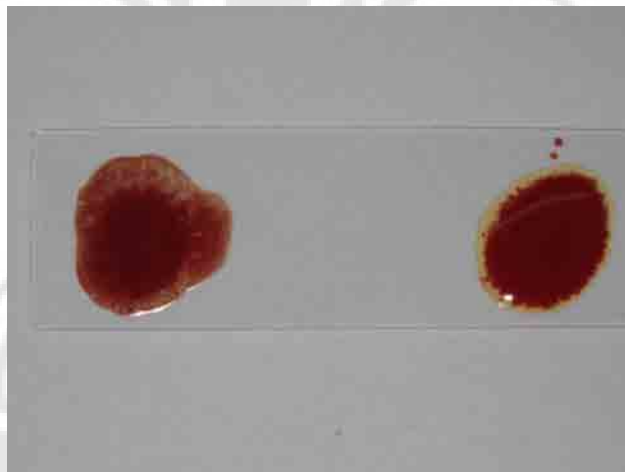
Gambar 4. 17. bentuk matriks dari golongan darah AB data ke-6



Gambar 4. 18. data ke-7 dari golongan darah AB

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

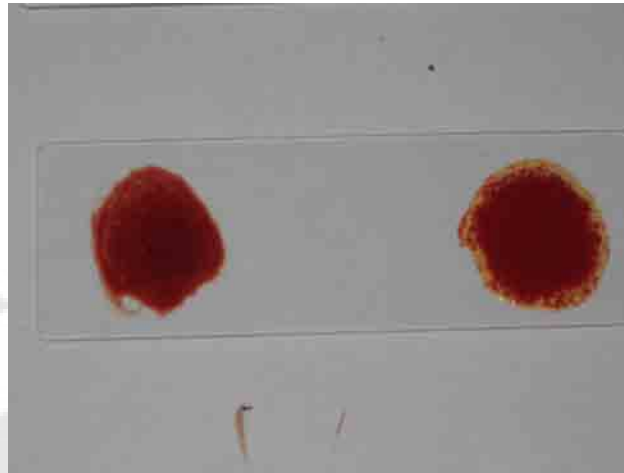
Gambar 4. 19. bentuk matriks dari golongan darah AB data ke-7



Gambar 4. 20. data ke-8 dari golongan darah AB

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

Gambar 4. 21. bentuk matriks dari golongan darah AB data ke-8



Gambar 4. 22. data ke-10 dari golongan darah AB

1	2	3	4	5	1	2	3	4	5
0	0	1	1	0	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1
1	1	1	1	0	0	1	1	1	1

Gambar 4. 23. bentuk matriks dari golongan darah AB data ke-10

Dapat kita perhatikan pada Gambar 4.12 hingga Gambar 4.23, perbandingan antara gambar asli dengan matriks yang dihasilkan masih memungkinkan terjadinya kesalahan pada proses identifikasi, hasil matriks yang digunakan untuk proses pengidentifikasian memiliki kemiripan dengan Gambar 4.7, yaitu matriks golongan darah O, hal ini dapat disebabkan oleh dua hal, yaitu tingkat keabuan yang digunakan untuk mengidentifikasi program hanya terdiri dari dua kelas, 1 dan 0, dan tingkat kecerahan RGB yang digunakan tidak dioptimalkan dengan warna merah saja.

Jika tingkat keabuan dibagi menjadi kelas yang lebih banyak, maka sistem dapat mengidentifikasi gambar lebih baik. Hal ini dikarenakan gradasi warna merah yang terjadi ketika darah menggumpal tidak dapat diklasifikasikan dengan baik oleh sistem hanya dengan menggunakan dua kelas tingkat keabuan.

Begitupula dengan tingkat kecerahan warna merah pada RGB, jika warna merah saja yang digunakan untuk mendeteksi citra darah, maka tingkat gradasi warna merah pada sistem dapat mengidentifikasi adanya penggumpalan pada citra darah.

Pada Gambar 4.22 dapat kita lihat bahwa gambar darah pada sebelah kiri dapat diidentifikasi oleh sistem sebagai gambar darah yang menggumpal, sedangkan

gambar di sebelah kanan diidentifikasi tidak menggumpal, hal ini menyebabkan sistem mengidentifikasi Gambar 4.22 sebagai golongan darah A.

Penyebab kegagalan pengidentifikasian golongan darah AB pada data-data di atas, jika kita lihat pada gambar asli, gambar penggumpalan darah pada kedua sisi tidak memiliki kerenggangan antara masing-masing penggumpalan darah, sehingga pembacaan pada matriks menghasilkan darah tersebut tidak menggumpal dan tidak menghasilkan warna terang selain merah. Agar sistem dapat membaca dengan jelas gambar penggumpalan darah, maka darah yang mengalami penggumpalan sebaiknya memiliki kerenggangan antara masing-masing penggumpalan. Hal ini berhubungan dengan teknik pengadukan darah dengan reagen secara fisik.

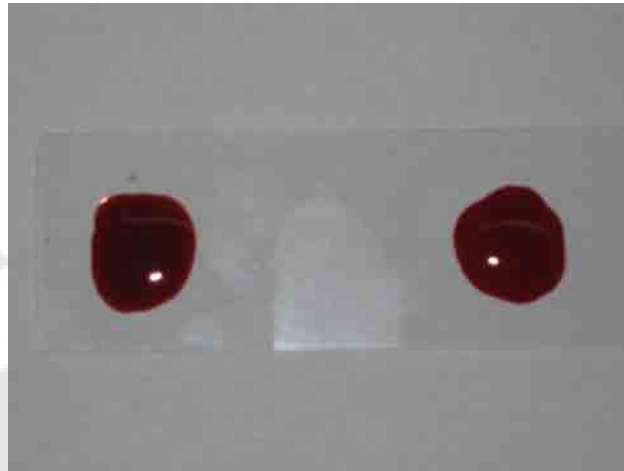
Pada Gambar 4.24, sistem mengidentifikasi bahwa gambar tersebut merupakan gambar dari golongan darah B, karena pada matriks gambar darah sebelah kanan didapatkan warna putih, dapat kita lihat pada gambar asli bahwa warna putih tersebut ternyata merupakan pantulan dari cahaya kamera digital. Dari hasil pengidentifikasian kali ini dapat kita simpulkan bahwa cahaya kilat dari kamera digital dapat mengganggu proses identifikasi.

Sebaiknya dalam pengambilan data, data diambil pada keadaan lingkungan yang terang sehingga kamera tidak perlu mengeluarkan cahaya kilat.

Tabel IV.9 menunjukkan hasil identifikasi yang dihasilkan golongan darah tipe O, pada tabel dapat kita lihat bahwa sebagian besar data yang digunakan untuk mengidentifikasi golongan darah O bisa mengenali pola penggumpalan darah dengan baik, hal ini terlihat pada tingkat akurasi yang dihasilkan, yaitu sebesar 90% dari sepuluh data yang digunakan untuk mengevaluasi perangkat lunak identifikasi golongan darah manusia.

Tabel IV. 9. Tabel identifikasi golongan darah O dengan input 5x5 dan metode optimasi LM

Data ke	Data dikenal sebagai tipe golongan			
	A	B	AB	O
1				ok
2				ok
3				ok
4				ok
5				ok
6				ok
7				ok
8		salah		
9				ok
10				ok



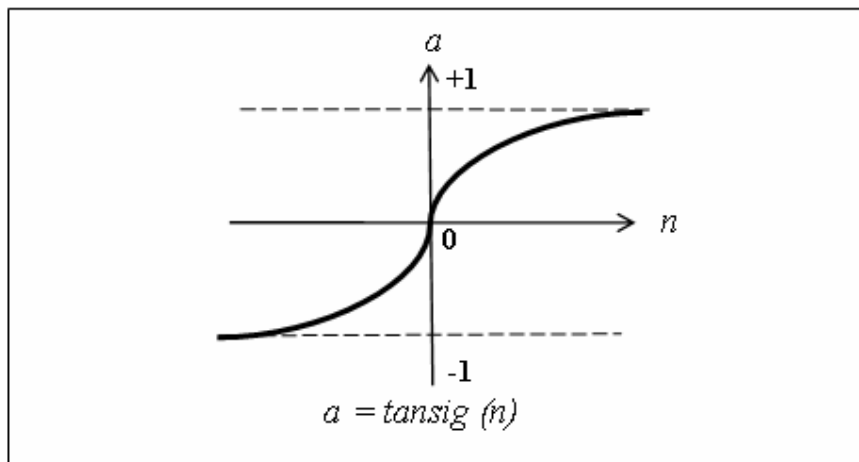
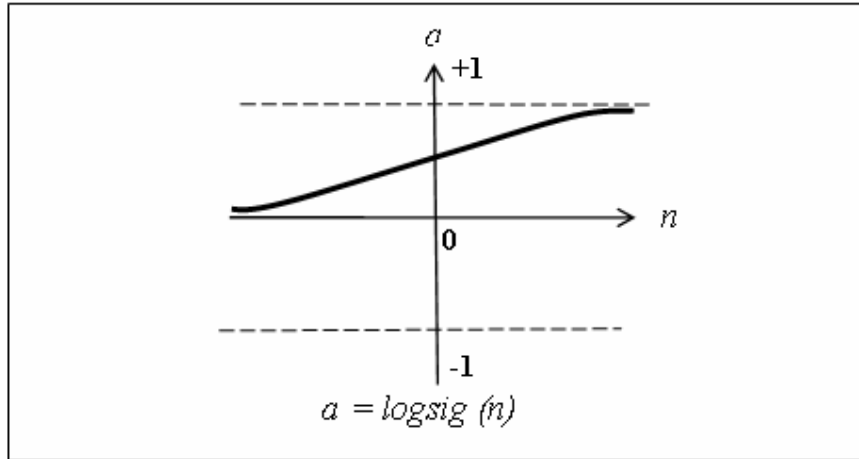
Gambar 4. 24. data ke-8 dari golongan darah O

1	2	3	4	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	0	1	1	1
1	1	1	0	0	0	0	1	1	1

Gambar 4. 25. bentuk dari golongan darah O data ke-8

4.5 PENGARUH FUNGSI AKTIVASI DAN TARGET

Perangkat lunak menggunakan fungsi aktivasi tansig untuk lapisan tersembunyi, penggunaan tansig menghasilkan akurasi yang lebih baik dari fungsi aktivasi logsig disebabkan karena fungsi aktivasi tansig menghasilkan keluaran yang memiliki rentang antara -1 sampai 1 sehingga data akan memiliki banyak klasifikasi lebih banyak jika dibandingkan dengan fungsi aktivasi tansig yang hanya memiliki rentang antara 0 sampai 1. Hal ini menyebabkan tingkat spesifikasi pada fungsi aktivasi tansig akan lebih jelas jika dibandingkan dengan fungsi aktivasi logsig, dapat kita lihat pada grafik yang dihasilkan oleh fungsi aktivasi tansig dan logsig, sebagai Gambar 4.26 berikut:



Gambar 4. 26. Fungsi sigmoid biner (logsig) dan sigmoid bipolar (tansig)

BAB V

KESIMPULAN

Berdasarkan analisis hasil ujicoba dan analisis, maka dapat disimpulkan beberapa hal, yaitu:

1. Sistem identifikasi golongan darah menggunakan Jaringan Syaraf Tiruan (JST) dengan menggunakan algoritma backpropagation dapat mengidentifikasi tipe-tipe golongan darah dengan tingkat akurasi sebesar 90% untuk golongan darah A, 80% untuk golongan darah B, 40% untuk golongan darah AB dan 90% untuk golongan darah O, dengan parameter-parameter yang digunakan ukuran besar matriks masukan JST 5x5, menggunakan fungsi aktivasi Tansig-Purelin, metode optimasi Levenberg-Marquadt, jumlah pelatihan sebanyak 83 kali, Jumlah data evaluasi identifikasi sebanyak sepuluh tiap golongan darah
2. Besarnya matriks masukan pada Jaringan Syaraf Tiruan akan mempengaruhi tingkat akurasi pada proses pengidentifikasian. Dengan ukuran matriks sebesar 5x5 menjadi masukan yang diproses dalam sistem, maka tingkat akurasi yang dihasilkan adalah 90% untuk golongan darah A, 80% untuk golongan darah B, 40% untuk golongan darah AB dan 90% untuk golongan darah O, sedangkan untuk besar matriks masukan dengan ukuran 10x10, maka tingkat akurasi yang dihasilkan adalah 40% untuk golongan darah A, 50% untuk golongan darah B, 30% untuk golongan darah AB, dan 90% untuk golongan darah O. Hal ini dikarenakan dengan lebih sedikit parameter untuk didefinisi akan menghasilkan jaringan syaraf yang jauh lebih adaptif. Oleh karena itu tindakan *pra-processing* dalam mereduksi data sangat penting.
3. *Marquadt* (LM) dapat memberikan peningkatan waktu pelatihan JST yang signifikan sebesar 2 epoh jika dibandingkan optimasi Gradient Descent yang memiliki waktu pelatihan sebesar 301 epoh pada input matriks 5x5

DAFTAR ACUAN

- [1] “Darah” diakses dari wikipedia tanggal 15 Desember 2007
<http://id.wikipedia.org/wiki/Darah>
- [2] “Golongan Darah” diakses dari wikipedia tanggal 15 Desember 2007
http://id.wikipedia.org/wiki/Golongan_darah
- [3] Diakses dari homehealth-uk.com tanggal 15 Desember 2007
http://www.homehealth-uk.com/image_01/resultsblood.jpg
- [4] Gonzalez, Rafael C, Richard E. Woods,”*Digital Image Processing Second Edition*”, New Jersey USA: Prentice Hall, 2001, hal 53-55
- [5] Tinku Acharya, Ajoy K. Ray, “*Image Processing Principles and Application*” , New Jersey USA : A John Wiley & Sons.Mc.Publication , 2005, hal. 23-26
- [6] Kusumadewi, Sri, “*Membangun Jaringan syaraf Tiruan Menggunakan MATLAB dan EXCELLINK*”, Yogyakarta : Graha Ilmu, 2004
- [7] Fausett, Laurene, “*Fundamentals of Neural Network Architectures, Algorithms, and Applications*”, New Jersey USA: Prentice Hall, 1994

DAFTAR PUSTAKA

- Baiqing Sun, "Research on Rejection Capabilities of Paper Currency Recognition System with the Neural Network Employing Gaussian Function", Disertasi, Kochi University of Technology, Japan, 2006.
- Ben Krose, *An Introduction to Neural Network 8th Ed.*, (Amsterdam: Univ of Amsterdam, 1996), hal. 18, hal. 31
- Christopher M. Bishop, *Neural Network for Pattern Recognition*, (Oxford: Clarendon Press, 1995), hal 7 – 9, hal 268, hal 291-300
- Cybenko, G. *Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems*, 2(4), 1989, hal. 33
- Fausett, L., *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*, (USA : Prentice-Hall Inc, 1994)
- Gonzalez, Rafael C, Richard E. Woods, *Digital Image Processing Second Edition*, (New Jersey USA: Prentice Hall), 2001
- Hermawan, Arief, *Jaringan Saraf Tiruan : Teori dan Aplikasi*(Yogyakarta : ANDI, 2006)
- Jae-Kang Lee, Seong-Goo Jeon, et.al., "Distinctive Point Extraction and Recognition Algorithm for Various Kinds of Euro Banknotes", *International Journal of Control, Automation, and Systems* Vol. 2, No. 2, (2004)
- Kusumadewi, Sri, *Membangun Jaringan syaraf Tiruan Menggunakan MATLAB dan EXCELLINK*", (Yogyakarta: Graha Ilmu, 2004), hal. 50 - 51
- Maria Petrou, Panagiota Bosdogianni, *Image Processing – the Fundamentals* (New York: John Wiley & Sons, Ltd, 1999), hal. 1-2
- Mark S. Nixon, Alberto S. Aguado, *Feature Extraction in Computer Vision and Image Processing*, (Oxford: Newnes, 2002), hal 247, hal 33
- MATLAB HELP, *Image Processing Toolbox*, mathworks Inc.
- Randy Crane, *A Simplified Approach to Image Processing* (New Jersey: Prentice Hall, 1997), hal 304
- Rumelhart, D. E., R. Durbin, R. Golden, and Y. Chauvin. *Backpropagation: Theory, Architectures, and Applications* (New Jersey: Lawrence Erlbaum, 1995), hal 1-34
- Tinku Acharya, Ajoy K. Ray, *Image Processing Principles and Application*, (New Jersey USA: A John Wiley & Sons. Mc. Publication, 2005), hal. 23-26

LAMPIRAN

Lampiran 1 Hasil uji coba identifikasi (5x5 pixel)
optimasi *Levenberg-Marquadt*

no	Jenis golongan darah			
	A	B	AB	O
1	ok	salah	ok	ok
2	ok	ok	ok	ok
3	ok	ok	salah	ok
4	ok	ok	ok	ok
5	ok	ok	salah	ok
6	salah	ok	salah	ok
7	ok	ok	salah	ok
8	ok	ok	salah	salah
9	salah	ok	ok	ok
10	ok	ok	salah	ok
Akurasi	80%	90%	40%	90%

Lampiran 2 Hasil uji coba identifikasi (10x10 pixel)
optimasi *Levenberg-Marquadt*

No	Jenis golongan darah			
	A	B	AB	O
1	ok	salah	salah	ok
2	ok	ok	ok	ok
3	salah	ok	salah	ok
4	ok	ok	ok	ok
5	salah	ok	salah	ok
6	salah	ok	salah	ok
7	salah	salah	salah	ok
8	ok	salah	salah	salah
9	salah	salah	ok	ok
10	salah	salah	salah	ok
Akurasi	40%	50%	30%	90%