

Penerapan Metode *Back Propagation Neural Network* pada Pendeteksian Kelainan Otak *Ischemic Cerebral Infraction* dengan Bahasa Pemrograman Delphi

Nurhayati*

Jurusan Fisika FMIPA Universitas Negeri Surabaya (UNESA)
Jl. Ketintang, Surabaya 60231

Fitri Adi Iskandarianto

Jurusan Teknik Fisika-FTI, Institut Teknologi Sepuluh Nopember
Kampus ITS Sukolilo, Surabaya 61111

Intisari

Penyakit stroke *ischemic cerebral infarction* merupakan salah satu gangguan kelainan otak yang banyak ditemukan. Pendeteksian dan pendiagnosaan kelainan yang terdapat pada otak yakni penyakit stroke infark dapat dilakukan oleh para radiolog dan dokter yang ahli dan berpengalaman dengan menggunakan *Magnetic Resonance Imaging* (MRI). Penelitian ini bertujuan untuk membuat suatu sistem komputasi pendeteksian otak dengan metode *backpropagation neural network* menggunakan bahasa pemrograman Delphi. Praproses data dilakukan pada gambar yang didapat dan digunakan sebagai inputan ke sistem Jaringan Saraf Tiruan (JST). Praproses data meliputi proses scanning, proses *greyscale*, equalisasi, filter, segmentasi, normalisasi *greyscale* tiap segmen yang dilakukan dengan software Delphi 5. Data berupa hasil normalisasi *greyscale* digunakan sebagai inputan ke JST. JST merupakan suatu model komputasi yang bekerja meniru jaringan saraf manusia. Jaringan yang digunakan adalah propagasi balik yang outputnya berupa bilangan biner pada kondisi T1 dan T2. Jaringan *Backpropagation* yang biasanya digunakan dalam berbagai aplikasi adalah metode *Backpropagation* dengan banyak lapisan. Proses training dilakukan untuk didapatkan struktur jaringan yaitu layer hidden 1, unit hidden 3, output 2 unit, kecepatan belajar 0,5; momentum 0,9; bobot awal random -0,1 sampai 0,4, kesalahan maksimum 0,001. Hasil pengujian menunjukkan bahwa dari 15 data pasien sakit dan 10 data pasien normal terdapat empat kesalahan pendeteksian.

KATA KUNCI: Back propagation, Neural Network, Image processing

I. PENDAHULUAN

Sistem saraf manusia adalah suatu jalinan jaringan saraf yang kompleks yang berfungsi mengkoordinasikan, menafsirkan dan mengontrol interaksi antara individu dengan lingkungan sekitarnya. Adanya pengaturan saraf tersebut maka jalinan komunikasi antara berbagai sistem tubuh dapat berfungsi sebagai unit yang harmonis.

Seringkali setelah seorang mendapat hasil MRI, ia masih harus menunggu ahli radiolog /dokter untuk mendapat diagnosa dari gambaran kelainan yang ada. Salah satu tugas seorang dokter dan ahli radiolog yang penting dan cukup sulit adalah ketika mendeteksi dan mendiagnosa kelainan otak. Hal ini dapat disebabkan karena otak merupakan organ tubuh yang letaknya tersembunyi sehingga sulit dideteksi dengan mata telanjang atau dapat disebabkan karena rendahnya kualitas gambar yang diperoleh atau bahkan juga disebabkan karena faktor subyektivitas dan kriteria pengambilan keputusan yang berbeda antara dokter atau ahli radiolog.

Perlu dikembangkan suatu *computer aided diagnoses* yang

mempunyai kemampuan untuk membantu dokter atau ahli radiolog. Pada tulisan ini penulis berusaha mencoba merancang simulator sistem JST yang berbasis pada perangkat lunak komputer sehingga diharapkan mampu mengidentifikasi gambaran jaringan kanker agar lebih cepat dan akurat. Menurut McCulloch dan Pits, JST merupakan suatu model komputasi yang bekerja meniru jaringan saraf manusia dengan menerima masukan berupa data numerik yang berasal dari ciri fisik dan struktur suatu obyek yang mengalami pra-proses data sebelumnya.

II. TINJAUAN PUSTAKA

A. Jaringan Saraf Tiruan

JST adalah sistem pemrosesan informasi yang mempunyai karakteristik secara umum seperti jaringan saraf biologis (otak manusia). Elemen-elemen struktural dari JST secara umum adalah simpul-simpul, lapisan-lapisan, sambungan antar simpul, paket bobot sambungan.

Simpul-simpul dianggap sebagai elemen komputasional atau elemen pemroses karena didalamnya terjadi penjumlahan input-input yang diterima terlebih dahulu dikalikan den-

*E-MAIL: nurhayati090806@yahoo.com

gan bobot sambungan yang dibawa tiap input. Hasil penjumlahan itu disebut dengan net input atau nilai input untuk suatu simpul. Net input kemudian dimasukkan ke dalam suatu fungsi yang disebut fungsi transfer atau fungsi aktivasi untuk menghasilkan suatu nilai yang dinamakan nilai aktivasi. Nilai aktivasi ini disebut juga dengan output simpul.

$$y_{in} = x_1w_1 + x_2w_2 \quad (1)$$

1. Cara Belajar

JST dibuat berdasarkan inspirasi dari jaringan saraf biologis, salah satu kemampuan yang diharapkan adalah belajar. Kemampuan belajar dari JST ditunjukkan dengan tingkah lakunya terhadap lingkungan. Misal suatu paket input diperlihatkan kepada JST lengkap dengan output yang diharapkan (untuk model jaringan saraf tertentu output tidak diperlihatkan) maka JST akan mengatur dirinya sendiri (dengan metode belajar tertentu) sedemikian rupa sehingga menghasilkan tanggapan-tanggapan seperti yang telah ditetapkan.

Istilah belajar pada JST adalah suatu proses pengaturan bobot-bobot sambungan antar simpul yang dilakukan dengan metode tertentu, sedemikian rupa hingga didapat bobot-bobot sambungan yang diinginkan. Proses belajar terjadi karena antara output yang dihasilkan jaringan tidak sama dengan output yang diharapkan. Bila bobot sambungan yang ada belum mampu menghasilkan output yang diharapkan, maka bobot akan diatur metode tertentu melalui latihan terus-menerus sehingga didapat susunan bobot yang baru dan lebih baik. Susunan bobot ini membrikan perbedaan antara output jaringan dan output pola akan semakin kecil sehingga output yang dikeluarkan jaringan akan sama dengan output yang diharapkan. Setelah dilatih, suatu JST siap diuji dan dipakai. Pengujian dikatakan berhasil apabila JST sanggup menerima input dan mengeluarkan output seperti yang diharapkan.

Dalam berlatih terdapat istilah himpunan pelatihan yaitu suatu himpunan yang terdiri dari pasangan-pasangan pola input dan pola output yang diinginkan. Tiap satu pola input dipasangkan dengan satu pola output target. Istilah pola digunakan sebagai pengganti dari semua informasi atau data-data yang masuk dalam JST. Selama proses belajar, bobot-bobot sambungan yang ada pada jaringan secara konvergen menuju suatu harga tertentu yang menandakan semakin kecilnya perbedaan antara output target dengan output jaringan.

2. Faktor-faktor belajar

Ada beberapa parameter-parameter yang turut menentukan keberhasilan suatu proses belajar. Parameter-parameter ini meliputi kecepatan belajar, momentum, inialisasi bobot, jenis adaptasi bobot dan penentuan jumlah neuron pada layer hidden. Faktor-faktor belajar tersebut sangat berpengaruh pada proses pembelajaran pada metode *Backpropagation*.

1. Inialisasi bobot

Salah satu pilihan yang dapat digunakan adalah

memilih nilai bobot v_{ij} dan w_{jk} pada suatu rentang kawasan nilai antara -0,5 dan 0,5 atau beberapa interval lain yang seimbang. Banyak studi empiris membuktikan bahwa meneruskan pelatihan pada saat error mencapai harga kecil yang stabil atau datar menghasilkan bobot yang tak diinginkan. Ini dapat menyebabkan kesalahan meningkat dan kualitas *mapping* dari jaringan menurun untuk mengatasi hal tersebut sebaiknya nilai bobot diberi harga yang random. Ada dua jenis adaptasi bobot dalam pelatihan pada algoritma JST yaitu adaptasi kumulatif dan adaptasi biasa. Adaptasi biasa ialah bobot diadaptasi pada setiap pola yang masuk, sedangkan adaptasi kumulatif ialah bobot baru diadaptasi setelah seluruh pola masuk. Kedua jenis adaptasi diatas dapat dipakai karena sama-sama memberikan hasil yang baik Tetapi perlu diingat hasil pelatihan yang baik didapat dari kondisi yang random.

2. Kecepatan Belajar (α)

Parameter kecepatan belajar menunjukkan intensitas dalam proses belajar, parameter ini juga menentukan efektifitas dan konvergensi dari pelatihan. Umumnya harga optimum dari kecepatan belajar tergantung dari permasalahan yang dihadapi dan tidak ada suatu harga yang cocok untuk semua kasus pelatihan. Nilai α yang besar dapat menyebabkan meningkatnya kecepatan dalam menentukan nilai yang diinginkan tetapi juga memungkinkan adanya suatu overshoot. Harga kecepatan belajar dipilih pada harga antara 0 sampai 10.

3. Momentum (μ)

Peningkatan kecepatan proses training dilakukan dengan menambahkan parameter momentum. Rentang nilai momentum yang umum digunakan adalah antara 0 sampai dengan 1, dengan nilai yang paling optimal adalah 0,9. Momentum dipakai untuk mempercepat *konvergensi*.

$$\Delta w_{jk}(t+1) = \alpha \delta_k z_j + \mu \Delta w_{jk}(t) \quad (2)$$

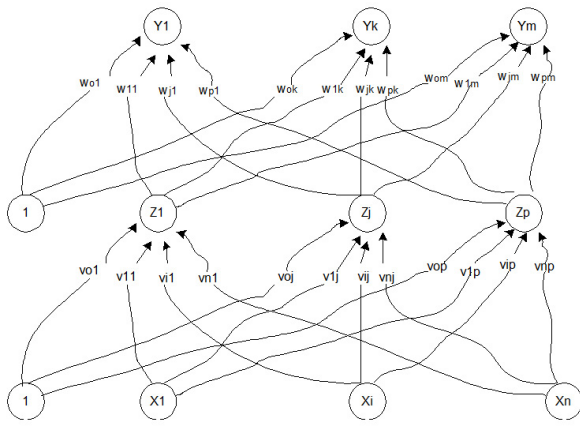
$$\Delta v_{ij}(t+1) = \alpha \delta_j z_i + \mu \Delta v_{ij}(t) \quad (3)$$

4. Kesalahan Belajar dan batas iterasi

Tingkat keberhasilan proses belajar dari sebuah jaringan yang sedang dilatih dapat dilihat dari parameter E (error) yang menunjukkan kemampuan jaringan. Parameter E tidak harus bernilai nol, tapi bernilai cukup kecil tergantung masalah yang dihadapi. Semakin kecil nilai E berarti jaringan semakin peka. Ini berakibat kurang baik karena biasanya kita menginginkan sebuah sistem pengenalan yang dapat mengabaikan perubahan minor sampai level tertentu.

3. Metode Jaringan Saraf Tiruan

Jaringan *Backpropagation* yang biasanya digunakan dalam berbagai aplikasi adalah metode *Backpropagation* dengan banyak lapisan.



Gambar 1: Jaringan *Backpropagation* dengan satu hidden layer[8]

Pada pengoperasian JST terdapat 2 tahap operasi yang terpisah yaitu tahap belajar dan tahap pemakaian. Tahap belajar merupakan proses untuk mendapatkan bobot koneksi yang sesuai. Penyesuaian bobot dimaksudkan agar setiap pemberian input ke jaringan menghasilkan output yang diinginkan. Adapun algoritma *Backpropagation* dapat diketahui dari serangkaian perhitungan. Berdasarkan Gambar 2, diketahui bahwa masing-masing sinyal input x_1 akan meneruskan sinyal inputan ke semua lapisan di atasnya. Tiap hidden unit ($Z_j, j = 1, ,p$) akan menjumlahkan bobot sinyal input:

$$z_in_j = v_{oj} + \sum_{i=1}^n x_i v_{ij} \tag{4}$$

Output sinyal diperoleh dengan menerapkan fungsi aktivasi

$$z_j = f_-(z_in_j) \tag{5}$$

Unit hidden ini akan mengirimkan sinyal ke seluruh lapisan di atasnya yaitu lapisan output. Tiap unit output akan menjumlahkan bobot sinyal input.

$$z_in_k = w_{ok} + \sum_{j=1}^p z_j w_{jk} \tag{6}$$

Dengan menerapkan fungsi aktivasi maka,

$$y_k = f(y_in_k) \tag{7}$$

Tahap ini disebut dengan tahap propagasi maju. Sedangkan proses perhitungan propagasi mundur (*backpropagation*) perlu perhitungan kesalahan outputnya. Apabila output yang dihasilkan belum sesuai dengan apa yang diinginkan maka jaringan akan menghitung error output yang besarnya:

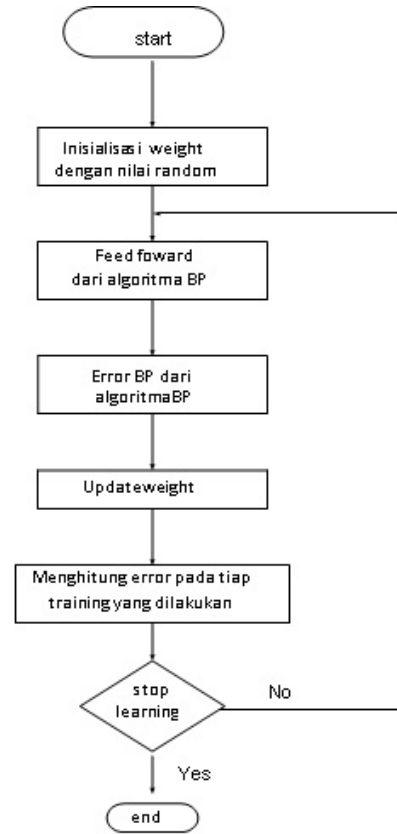
$$\delta_k = (t_k - y_k) f'(y_in_k) \tag{8}$$

Perhitungan update bobot

$$\Delta w_{jk} = \alpha \delta_k z_j \tag{9}$$

Sedangkan pengoreksian bias

$$\Delta w_{ok} = \alpha \delta_k \tag{10}$$

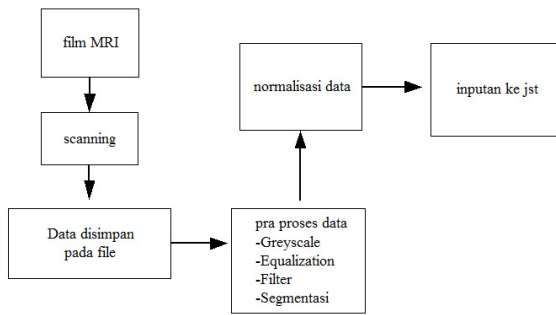


Gambar 2: Algoritma *Backpropagation*[12]

B. Penyakit Stroke dan MRI

Penyakit stroke adalah gangguan fungsi otak akibat aliran darah ke otak mengalami gangguan. Akibatnya, nutrisi dan oksigen yang dibutuhkan otak tidak terpenuhi dengan baik. Penyebab stroke ada 2, yaitu adanya sumbatan di pembuluh darah (trombus) dan adanya pembuluh darah yang pecah. Pada CVD (*Cerebral Vascular Disease*) autoregulasi terganggu.

Resonansi magnetic (MR) merupakan salah satu cara pemeriksaan diagnostik dalam ilmu kedokteran dengan menggunakan medan magnet. Pencitraan yang diperoleh hampir mirip dengan CT dan tidak ada bahaya radiasi bagi pasien dan operator. Prinsip dasar MR adalah inti atom yang bergetar dalam medan magnet. Prinsip ini pertama kali ditemukan oleh Bloch dan Purcell pada tahun 1946, proton yang merupakan inti atom hidrogen bila ditembak tegak lurus pada intinya dalam medan magnet yang berfrekuensi tinggi secara periodik, maka proton tersebut akan bergetar. Bila medan magnet ini dimatikan, maka proton yang bergetar tadi akan kembali ke posisi semula dan akan menginduksi dalam suatu kumparan untuk menghasilkan sinyal elektrik yang lemah. Bila hal ini terjadi berulang-ulang dan sinyal elektrik tersebut ditangkap kemudian diproses dalam komputer akan dapat disusun menjadi suatu gambar. Pada tahun 1980 alat MR mulai ramai digunakan dirumah sakit besar terutama di Amerika dan Eropa.



Gambar 3: Pengolahan data Gambar kepala hasil MRI

III. METODOLOGI PENELITIAN

Data yang diambil pada penelitian ini berupa gambar otak potongan aksial berupa otak normal dan otak yang terdapat kelainan stroke yaitu jenis *ischemic cerebral infarction*. Gambar yang diambil berupa potongan aksial dikarenakan bagian otak yang sering terlihat adanya kelainan infark adalah bagian potongan aksial dan ini merupakan saran dari dokter pembimbing. Gambar 3 merupakan blok diagram tentang pengambilan data, praproses data, dan pengolahan data.

Program Delphi digunakan untuk mendapatkan citra yang dapat dimodifikasi. Data berupa gambaran irisan otak di scan dengan pengaturan yang sama dan disimpan dalam bentuk gambar warna. Agar pengolahan datanya lebih mudah maka citra tersebut diubah sehingga membentuk citra dengan nilai intensitas Red, Green dan Blue yang sama dan proses ini disebut dengan proses *greyscale* citra. Citra *greyscale* yang didapatkan kemudian dilakukan proses pembuatan histogram citra untuk mengetahui distribusi *greyscale* pada citra, sehingga mudah untuk diubah kekontrasan dan kecerahan citra. Bila citra yang didapatkan mempunyai penampakan yang kurang jelas, maka dapat diperbaiki dengan memodifikasi histogram pada citra tersebut yaitu dengan membuat histogram equalization. Histogram equalization yaitu suatu proses pemerataan distribusi intensitas dari citra (perataan histogram) yang diantaranya bertujuan untuk meningkatkan kekontrasan suatu citra. Citra berupa gambaran irisan otak ini berukuran (185×185) pixel untuk tiap gambaran irisan otak. Citra ini berupa gambaran otak yang juga terdiri dari gambaran latar belakang otak yang terdiri dari informasi pasien dan kondisi pemotongan yang dinyatakan dalam bentuk tulisan.

IV. ANALISIS DATA DAN PENGUJIAN

Metode training JST yang digunakan adalah JST dengan metode *Backpropagation*. Jaringan *Backpropagation* akan mengadakan proses pembelajaran dengan pengaturan bobot hingga output yang dihasilkan jaringan sesuai dengan yang diharapkan. Struktur jaringan saraf yang dirancang terdiri dari beberapa layer. Pada penelitian ini digunakan tiga layer yaitu layer input, layer hidden dan layer output.

Pada proses pelatihan, untuk menentukan nilai ukuran unit

TABEL I: Variasi hidden unit dengan menggunakan data pasien sakit

No	Hidden	Kondisi	iterasi	error	Output1	Output2
1	3	T1	7	0,000979701	0,974014963	0,959728526
		T2	7	0,000889791	0,977384294	0,960963510
2	10	T1	3	0,000758384	0,985696381	0,962390535
		T2	3	0,000576012	0,984171110	0,967957120
3	50	T1	1	0,000000000	1,000000000	0,999999811
		T2	1	0,000000000	1,000000000	0,99999972

TABEL II: Variasi hidden unit dengan menggunakan data pasien normal

No	Hidden	Kondisi	iterasi	error	Output1	Output2
1	3	T1	7	0,000879825	0,977678573	0,961134510
		T2	7	0,000866681	0,978028329	0,961373769
2	10	T1	3	0,000529045	0,983769769	0,969563513
		T2	3	0,000444656	0,983935157	0,972426962
3	50	T1	1	0,000000000	1,000000000	0,999999811
		T2	1	0,000000000	1,000000000	0,99999972

hidden dilakukan dengan serangkaian percobaan dengan nilai momentum, kecepatan belajar, dan nilai random bobot awal yang sama. Pada pelatihan untuk mencari jumlah unit hidden digunakan error maksimum 0,001, konstanta kecepatan belajar (α) 0,5; momentum (μ) 0,9; inisialisasi bobot awal v_{ij} dan w_{jk} yaitu nilai yang random antara -0,1 hingga 0,4.

Jumlah unit hidden yang besar menyebabkan output antara jaringan normal dan sakit infark tidak bisa dibedakan dan jaringan langsung mencapai nilai yang konvergen dengan hanya satu kali iterasi dan ini tidak diharapkan. Pada pelatihan untuk mencari besarnya nilai konstanta belajar digunakan jumlah unit hidden 3, error maksimum 0,001; momentum (μ) 0,9; inisialisasi bobot awal v_{ij} dan w_{jk} yaitu nilai yang random antara -0,1 hingga 0,4, dan data yang digunakan hanya dipilih data dari satu pasien saja (data pasien no 4).

Proses latihan yang berulang-ulang dengan mengadakan perubahan pada parameter sistem pada semua data yang ada baik data pola normal maupun data pola sakit maka didapatkan parameter sebagai berikut:

- * Harga nilai error maksimum : 0,001
- * Harga konstanta belajar (α): 0,5
- * Harga momentum (μ) : 0,9
- * Harga bobot awal v_{ij} : random antara -0,1 sampai 0,4
- * Harga bobot awal w_{jk} : random antara -0,1 sampai 0,4
- * Iterasi maksimum : 10000

TABEL III: Variasi konstanta belajar dengan menggunakan data pasien sakit

No	kons. belajar	Kondi si	itera si	error	Output1	Output2
1	0,1	T1	9	0,00079138	0,976498843	0,963852971
		T2	9	0,000784959	0,976688635	0,963969336
2	0,5	T1	7	0,000899825	0,977678573	0,961134510
		T2	7	0,000866681	0,978028329	0,961373769
3	0,9	T1	6	0,000804198	0,981469467	0,962096088
		T2	6	0,000786941	0,981913690	0,962445705

Dari parameter di atas akan digunakan sebagai parameter sistem pada proses pengujian untuk mengetahui apakah data yang ada dapat diklasifikasikan dengan benar atau tidak. Proses pengujian pada keseluruhan data setelah data tersebut dilatihkan pada jaringan dilakukan untuk mengetahui validasi program. Pada tahap pengujian, data yang diinputkan pada jaringan sudah tidak ada perhitungan bobot karena jaringan yang digunakan hanya propagasi maju sehingga tidak terjadi looping balik. Pada proses pelatihan digunakan 10 data pasien dan 10 data normal.

Dengan melihat nilai keluaran pada data pelatihan maka pengklasifikasian dapat dilakukan dengan menentukan nilai ambang batas atau threshold dengan kriteria sebagai berikut:

- * Output = 0 jika output[Y] < 0,96
- * Output = 1 jika output[Y] > 0,96
- * Output [T1 T2] : [1 0 1 1] : pasien sakit infark
- * Output [T1 T2] : [1 0 1 0] : pasien sakit infark
- * Output [T1 T2] : [1 1 1 1] : pasien normal.

Bila pada pembelajaran maupun pada pengujian didapatkan variasi output yang berbeda dari kaidah pola identifikasi di atas maka pengklasifikasian tidak dapat dilakukan dan dapat dimungkinkan bahwa data yang dilatihkan tersebut merupakan pasien yang terkena kelainan otak lain. Hasil pengklasifikasian didapatkan data seperti ditunjukkan Tabel IV - VI.

Data pasien sakit infark yang berjumlah 15 data pasien sakit terdapat dua kesalahan pengidentifikasian, dan dari 12 data pasien normal terdapat dua kesalahan pengidentifikasian. Jumlah kesalahan pengidentifikasian dari keseluruhan data yang berjumlah 27 ada 4 kesalahan pengidentifikasian. Kebenaran dari sistem JST ini sebesar:

$$\frac{\text{jumlah data yang benar}}{\text{keseluruhan data}} \times 100\% = 85,19\%$$

V. SIMPULAN

Simpulan yang dapat diambil dari proses pelatihan metode *Backpropagation* neural network adalah:

1. Data berupa gambar irisan aksial kepala dari hasil gambaran MRI dapat dikenali terkena sakit infark atau normal dengan menggunakan sistem JST. Sebelum data diinputkan ke JST maka gambar akan melalui praproses data berupa data proses *greyscale*, equalisasi, filter background, segmentasi, perhitungan rata segmen *greyscalenya* dan normalisasi. Kesemuanya proses ini dilakukan dengan bantuan software delphi 5.
2. Untuk mengidentifikasi keadaan sakit infark atau normal maka jaringan dilatihkan data pada kondisi T1 dan T2 untuk tiap pasien yang tiap kondisi mempunyai dua output yang berbeda. Dalam proses pengidentifikasiannya di buat kaidah:
 - * Output = 0 jika output[Y] < 0,96
 - * Output = 1 jika output[Y] > 0,96
 - * Output [T1 T2] : [1 0 1 1] : pasien sakit infark
 - * Output [T1 T2] : [1 0 1 0] : pasien sakit infark
 - * Output [T1 T2] : [1 1 1 1] : pasien normal
3. Kebenaran sistem JST yang dibuat sebesar 85,19 %

[1] Arune Kulkarni, *Artificial Neural Network for Image Understanding* (Van Nostrand Reinhold, New York, 1994)

[2] A.Price,Lorraine, *Patofisiologi Proses-proses Penyakit* (Buku Kudu, Edison empat, Buku Kedokteran EGC, Jakarta , 1994)

[3] Aniati Murni, *Pengantar Pengolahan Citra* (PT Elex Media Komputindo Kelompok Gramedia, Jakarta, 1992)

[4] Chin, Teng Lin dan CS George Lee, *Neural Fuzzy System* (PTR Prentice Hall, 1996)

[5] G.W.Awcock & R.Thomas, *Applied Image Processing* (Mc Graw-Hill,Inc, Singapore, 1996)

[6] Howard E Burdick;McGrawHill, *Digital Imaging Theori and Applications* (Mc Graw-Hill, 1997)

[7] Husni, *Deteksi Tuberkulosis PARu pada Gambaran Toraks Dada dengan menggunakan Fuzzy-Neural Networks* (Teknik Elektro,

Fakultas Teknologi Industri, ITS, 2001)

[8] Laurene Fauset, *Fundamental of Neural Networks* (Prentice Hall,Inc, United Stated, 1994)

[9] Prof DR. Mahar Mardjono, Prof DR. Priguna Sidharta, *Neurologi Klinis Dasar* (Edisi ke-6, Dian Rakyat, Jakarta, 1994)

[10] Sylvia Louis j.Galbiati,JR, *Machine Vision and Digital Iamge Processing* (Prentice-Hall International,Inc, NewYork, 1990)

[11] Syahriar Rasad, Sukonto Kartoleksono, Iwan Ekayuda, *Radiologik Dianostik* (FKUI,Jakarta, 1999)

[12] Zulhaida, *Identifikasi Kelainan SinyalJantung pada Electrocardiogram (ECG) dengan Jaringan Saraf Tiruan*(Teknik Fisika, Fakultas Teknologi Industri, FTI ITS, Surabaya, 1995)

TABEL IV: Identifikasi Infark pada proses pelatihan

T1 Output1	T1 Output2	T2 Output1	T2 Output2	Identifikasi	Deteksi
0,976956504	0,960718161	0,97741955	0,960960928	1 1 1 1	Normal
0,973998787	0,95972415	0,97743159	0,960984817	1 0 1 1	Infark
0,973990358	0,959720507	0,977111602	0,960811455	1 0 1 1	Infark
0,974014963	0,959728526	0,977384284	0,96096351	1 0 1 1	Infark
0,973343929	0,959479004	0,977202473	0,960822004	1 0 1 1	Infark
0,973527098	0,95954775	0,97617847	0,960479517	1 0 1 1	Infark
0,974390386	0,95986374	0,974904601	0,960059041	1 0 1 1	Infark
0,974498579	0,959911387	0,974708472	0,959981163	1 0 1 0	Infark
0,973396522	0,95949899	0,975014277	0,960085899	1 0 1 1	Infark
0,973802976	0,959651477	0,976962668	0,960729437	1 0 1 1	Infark

TABEL V: Identifikasi Infark pada proses pengujian

T1 Output1	T1 Output2	T2 Output1	T2 Output2	Identifikasi	Deteksi
0,976956504	0,960718161	0,977419550	0,960960928	1 1 1 1	Normal
0,973998787	0,959724150	0,977431590	0,960984817	1 0 1 1	Infark
0,973990358	0,959720507	0,977111602	0,960811455	1 0 1 1	Infark
0,974014963	0,959728526	0,977384284	0,960963510	1 0 1 1	Infark
0,973343929	0,959479004	0,977202473	0,960822004	1 0 1 1	Infark
0,973527098	0,959547750	0,976178470	0,960479517	1 0 1 1	Infark
0,974390386	0,959863740	0,974904601	0,960059041	1 0 1 1	Infark
0,974498579	0,959911387	0,974708472	0,959981163	1 0 1 0	Infark
0,973396522	0,959498990	0,975014277	0,960085899	1 0 1 1	Infark
0,973802976	0,959651477	0,976962668	0,960729437	1 0 1 1	Infark
0,974621648	0,959953565	0,977097058	0,960779594	1 0 1 1	Infark
0,973562002	0,959560972	0,974196542	0,959794527	1 0 1 0	Infark
0,973334793	0,959475681	0,975851041	0,960388541	1 0 1 1	Infark
0,976238104	0,960525082	0,977332412	0,960948708	1 1 1 1	Normal
0,973457652	0,959522633	0,976903691	0,960708642	1 0 1 1	Infark

TABEL VI: Identifikasi Normal pada proses pengujian

T1 Output1	T1 Output2	T2 Output1	T2 Output2	Identifikasi	Deteksi
0,978336542	0,961662048	0,975171545	0,960138199	1 1 1 1	Normal
0,976814081	0,960714902	0,977028476	0,960751666	1 1 1 1	Normal
0,977248537	0,960909	0,978247252	0,961556812	1 1 1 1	Normal
0,977678573	0,9611345	0,978028329	0,961373769	1 1 1 1	Normal
0,975443791	0,9602486	0,976677307	0,96033337	1 1 1 1	Normal
0,974934931	0,9600639	0,975999774	0,96041593	1 1 1 1	Normal
0,978406627	0,9617185	0,975669635	0,960344762	1 1 1 1	Normal
0,973602291	0,9595761	0,976613419	0,960608297	1 0 1 1	Infark
0,975750366	0,9603428	0,978403536	0,961715953	1 1 1 1	Normal
0,973892205	0,9596842	0,977997454	0,961341374	1 0 1 1	Normal
0,976135599	0,9604719	0,978175634	0,961491997	1 1 1 1	Normal
0,977593273	0,9610543	0,975116984	0,960119952	1 1 1 1	Infark