

**OPTIMALISASI KINERJA SISTEM SEKURITAS RUMAH
MODERN BERBASIS JARINGAN SARAF TIRUAN (JST)**

***PERFORMANCE OPTIMALIZATION OF MODERN HOUSE
SECURITY SYSTEM BASED BACKPROPAGATION NEURAL
NETWORK (BNN)***

**ANTHOINETE P.Y WAROH
P2700211443**



**PROGRAM PASCASARJANA
UNIVERSITAS HASANUDDIN
MAKASSAR
2013**

**OPTIMALISASI KINERJA SISTEM SEKURITAS RUMAH
MODERN BERBASIS JARINGAN SARAF TIRUAN (JST)**

Tesis

Sebagai Salah Satu Syarat Untuk Mencapai Gelar Magister

Program Studi

Teknik Elektro

Disusun dan diajukan oleh

Anthoinete P.Y. Waroh

Kepada

**PROGRAM PASCASARJANA
UNIVERSITAS HASANUDDIN
MAKASSAR**

2013

PERNYATAAN KEASLIAN TESIS

Yang bertanda tangan dibawah ini:

Nama : Anthoinete P.Y. Waror

Nomor mahasiswa : P2700211443

Program Studi : Teknik Elektro

Konsentrasi : Teknik Informatika

Menyatakan dengan sebenarnya bahwa tesis yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan tulisan atau pemikiran orang lain. Apabila kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan tesis ini hasil karya orang lain, saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, Agustus 2013

Yang menyatakan

Anthoinete P.Y. Waroh

PRAKATA

Pujian dan Hormat penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena hanya perkenaanNya sehingga saya dapat menyelesaikan Tesis yang berjudul "Optimalisasi Kinerja Sistem Sekuritas Rumah Modern Berbasis Jaringan Saraf Tiruan (JST)".

Diharapkan melalui Tesis ini dapat memberikan masukan pada masyarakat bahwa dibutuhkan suatu Teknologi yang dapat mengoptimalkan sistem untuk keamanan rumah yang lebih baik dan terhindar dari pencurian .

Besar Harapan Penulis bahwa proposal yang merupakan pengajuan penelitian ini dapat diterima berdasarkan relevansi yang nyata di masyarakat. Pada kesempatan ini kami ucapkan terima kasih yang sebesar-besarnya kepada:

1. Suamiku tercinta Saul Yan Makalew serta anakku marcellino makalew dan keluargaku yang telah memberikan doa dan motivasi kepada penulis selama study sampai penulisan tesis.
2. Bapa Prof.Dr.Ir. Salama Manjang,MT, Selaku Ketua Program Study.
3. Bapak Dr.Ir.Andani Achmad,MT,Selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Hasanuddin.
4. Bapak Dr.Elyas Palantei,ST.M.Eng.Selaku pembimbing I.
5. Bapak Dr. Adnan Achmad,MT.Selaku pembimbing II
6. Seluruh dosen pengajar program pascasarjana program studi teknik elektro universitas hasanuddin.
7. Seluruh staf karyawan Jurusan Teknik Elektro Fakultas Teknik Universitas Hasanuddin atas bantuan yang diberikan kepada saya.
8. Kepada rekan-rekan seangkatan,Tracy Kereh, Ali Ramchie, Ronny Katuuk, Olga Melo, Sulastri Eksan, Daud Salemba, Tony

Wungkana, Johan Pongoh, Anritsu Polii, Doostenreyk Kantohe, Robby Lumbu, Nikita Sajangbati, Harson Kapoh, Robby Tangkudung.

9. Kepada Marson Budiman MT ,Veny Pongawa MT bersedia menjadi objek penelitain tesis saya.

Akhir kata semoga tesis ini dapat memberikan manfaat bagi pembaca, khususnya para pengguna yang ingin menggunakan sistem ini agar saat ditinggal pergi rumah tetap aman.

Manado, Agustus 2013

Penulis,

Anthoinete P.Y Waroh

ABSTRAK

ANTHOINETE P.Y.WAROH . Optimalisasi Kinerja Sistem Sekuritas Rumah Modern Berbasis Jaringan Saraf Tiruan (JST) (dibimbing oleh Elyas Palantei dan Adnan).

Perkembangan yang pesat dibidang teknologi komputer, elektronik, telekomunikasi maupun mekanik telah menghasilkan berbagai aplikasi canggih . Tujuan penelitian ini untuk membuat sistem sekuritas masuk rumah modern yang cerdas dengan memanfaatkan pola pencitraan wajah, menganalisis akses pencitraan yang cerdas dan membuat prototype dari sistem agar teroptimalisasi kinerjanya. Metode yang digunakan adalah analisis akuisisi. Sistem pengenalan wajah menggunakan citra wajah yang diambil dari 7 wajah manusia dengan 5 posisi wajah yang berbeda-beda, yaitu posisi biasa, tersenyum, serong ke kiri, serong ke kanan, dan mulut terbuka. Dalam prakteknya, algoritma backpropagation capturing (perekaman) wajah dengan jarak konstan dikumpulkan melalui proses mengakses data terhadap citra wajah dengan jarak konstan 30 cm (dekat kamera webcam) dengan kondisi pencahayaan yang relatif sama. Selanjutnya diadakan pengujian dan pengukuran kepada 7 wajah manusia yang tidak dikenal (tidak diteliti). Hasil penelitian menunjukkan bahwa dari 35 citra wajah yang diteliti dan 35 citra wajah yang tidak diteliti memiliki ukuran yang bervariasi, yaitu mulai dari 300 sampai dengan 500 piksel. Dapat disimpulkan bahwa klasifikasi yang menunjukkan angka keberhasilannya adalah ketika diaplikasikan pada wajah yang dikenal oleh sistem maka sinyal lampu akan menyala (pintu terbuka) dan ketika wajah tak dikenalnya lampu akan mati (pintu tertutup). Data pengujian keberhasilan pengenalan pola wajah 96% dan yang tidak sesuai pengenalan pola wajah 98 %.Sistem yang dibuat adalah sebuah sekuritas rumah modern berbasis jaringan saraf tiruan (JST) yang diuji pada software matlab R2009a dengan pengenalan pola citra wajah face parttern recognition.

Kata kunci : algoritma backpropagation, pengenalan pola wajah.

ABSTRACT

ANTHOINETE P.Y.WAROH. *Performance Optimization of Modern House Security System Based Backpropagation Neural Network (BBN) (supervised by Elyas Palantei and Adnan).*

Security is staying home to be a depressing thing coveted by everyone, at the time of leaving the house to do a trip. The face recognizing system used the face image taken from 7 human faces with 5 different face positions i.e.the normal face position, smiling face, left tilt face, right tilt face and face with open mouth. In practice, the back propagation capturing algorithm (recording) of the faces with the constant distance was collected through the data accessing process on the face image with the constant distance of 30 cm (near to webcam camera),with relatively similar lightning condition, the examination and measurement were then carried out of the 7 human faces determined and 7 human faces who were not known by the program because they were entered into the image taking data through the research process, this was proven that the system recognize 35 face images determined and 35 face which were not entered into data system. The image measurements could be various starting from 300 to 500 pixel. It can be concluded the classification indicated the success figures, the application when the faces were recognized, the light signal was on (the door opened), and if the faces were not recognized the light would be off (the door opened).The success examination data of the face pattern recognition was 96 % and which was not suitable with the face pattern recognition was 98%. The research result indicates that the system made is the artificial nerve network based a modern house security which is examined on the software matlab R2009a with the face pattern recognition.

Key-words : *Back propagation Algorithm, face pattern recognition*

DAFTAR ISI

PRAKATA	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xiv
I. PENDAHULUAN	1
A. Latar Belakang Masalah.....	1
B. Rumusan Masalah	2
C. Tujuan Penelitian	2
D. Manfaat Penelitian	3
E. Ruang Lingkup Penelitian.....	3
II. TINJAUAN PUSTAKA	4
A. Road Map	4
B. Jaringan Syaraf Tiruan	8
C. Arsitektur Jaringan.....	10
1. Jaringan Layer Tunggal (single layer network).....	5
2. Jaringan Layer Jamak (multi layer network).....	11
3. Model JST Dua Lapisan Dengan Umpan Balik.....	12
4. Model JST Lapisan Kompetitif.....	13
D. Fungsi Aktifasi.....	14

E. Backpropagation	20
1. Arsitektur Backpropagation.....	20
2. Neuron.....	21
3. Threshold.....	22
4. Pelatihan Standart Backpropagation.....	23
5. Optimalitas Arsitektur Backpropagation.....	28
6. Variasi Backpropagation.....	31
F. Pemograman Backpropagation dengan Matlab	33
1. Membentuk Jaringan.....	33
G. Konsep Pemrosesan Citra	36
III. METODE PENELITIAN	40
A. Tempat Dan Waktu Penelitian	40
B. Jenis Penelitian Dan Metode Penelitian	40
C. Rancangan Penelitian	40
D. Rancangan Sistem	40
E. Pelatihan Backpropagation	45
F. Teknik Meningkatkan Kecepatan Proses Image Wajah Pada Sistem JST Backpropagation	49
G. Proses Awal Sistem	56
H. Proses Pembentukan Pola Jaringan Saraf Tiruan Propagasi Balik Dan Pelatihan	65
I. Pengujian Sistem melalui Lampu Indikator	66

IV. HASIL DAN PEMBAHASAN	69
A. Pendeteksi Wajah	69
B. Proses Inisial Dan Input Wajah	69
C. Proses Image-Neural Network	72
D. Pengujian Kondisi Citra Biner.....	75
E. Neural Network.....	81
F. Kelemahan Sistem Yang Dibangun.....	129
V. PENUTUP	131
1. Kesimpulan.....	131
2. Saran.....	132

Daftar Pustaka

Lampiran

DAFTAR GAMBAR

Nomor	Halaman
1. Diagram Alir riset JST untuk rumah modern.....	4
2. Model Sederhana Jaringan Syaraf Tiruan	9
3. Jaringan Layar Tunggal.....	10
4. Jaringan Layar Jamak	11
5. Model JST Dua Lapisan Dengan Umpan Balik	13
6. Model JST Lapisan Kompetitif	13
7. Struktur Neuron Jaringan Syaraf	14
8. Fungsi Aktivasi Pada Jaringan Syaraf Sederhana.....	16
9. Fungsi Aktivasi Sigmoid Biner	18
10. Fungsi Aktivasi Sigmoid Bipolar	19
11. Fungsi Aktivasi Linear (Identitas).....	19
12. Arsitektur Backpropagation	21
13. Konfigurasi Sistem Sekuritas Rumah Modern Berbasis JST	41
14. Tampilan aplikasi matlab	43
15. Diagram alir pengenalan citra wajah	55
16. Diagram alir pengidentifikasian wajah.....	56
17. Diagram alir pengidentifikasian wajah (lanjutan)	57
18. Aplikasi pengenalan wajah	70
19. Pemasukan citra wajah sebagai input	72
20. Tampilan hasil Citra wajah warna yang sudah	

diproses dalam program.....	73
21. Citra wajah dalam bentuk citra <i>Grayscale</i> , dalam bentuk citra biner dan nilai biner.....	73
22. Wajah dikenal Sinyal yang keluar pada PC Lampu Hidup.....	76
23. Citra wajah dalam bentuk citra <i>Grayscale</i> , citra biner dan nilai biner.....	76
24. Wajah Tidak dikenal Sinyal yang keluar pada PC Lampu Mati.....	77
25. Tampilan hasil training JST.....	81

DAFTAR TABEL

Nomor	Halaman
1. Citra wajah dan data referensi.....	71
2. karakter objek wajah.....	74
3. hasil pengujian citra wajah pertama	78
4. hasil pengujian citra wajah	80
5. Hasil uji data testing sesuai input wajah untuk inisial eta	82
6. Hasil uji data testing sesuai input wajah untuk inisial lino	85
7. Hasil uji data testing sesuai input wajah untuk inisial eta.....	89
8. Hasil uji data testing sesuai input wajah untuk inisial mami.....	92
9. Hasil uji data testing sesuai input wajah untuk inisial jeine	95
10. Hasil uji data testing sesuai input wajah untuk inisial veny	99
11. Hasil uji data testing tidak sesuai input wajah marson.....	102
12. Hasil uji data testing tidak sesuai input wajah inisial bela.....	106
13. Hasil uji data testing tidak sesuai input wajah untuk orang yang berinisial seren	110
14. Hasil uji data testing wajahnya inisial meisi mirip inisial eta	113
15. Hasil uji data testing tidak sesuai input wajah untuk inisial felix ..	116
16. Hasil uji data testing tidak sesuai input wajah untuk inisial asrid .	120
17. Hasil uji data testing tidak sesuai input wajah untuk inisial asrid .	123
18. Hasil uji data testing tidak sesuai input wajah	126

DAFTAR LAMPIRAN

Nomor		Halaman
1.	Lampiran 1	134
2.	Lampiran 2	158
3.	Lampiran 3.....	165

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan yang pesat dibidang teknologi komputer, elektronik, telekomunikasi maupun mekanik telah menghasilkan berbagai aplikasi canggih dan cerdas yang merubah kehidupan manusia pada saat ini dan mendatang khususnya di Indonesia.

Sejumlah literature dan peneliti sebelumnya membuat aplikasi dengan metode segmentasi warna kulit, untuk pencitraan wajah, kekurangannya disaat warna kulit berubah makah sistem tidak mengenali orang tersebut. Peneliti kedua pengenalan wajah secara real time menggunakan algoritma (JST) dengan wajah tampak depannya saja , untuk mengaktifkan pintu terbuka dan tertutup, program tidak mengenal wajah tampak samping kiri dan wajah samping kanan wajah manusia.

Sehingga para pelaku tindak kriminal tetap melakukan kejahatan didalam rumah mereka memiliki cara tersendiri untuk mengatasi hal tersebut sehingga sangat sulit untuk melindungi rumah terutama pintu rumah.

Untuk mengatasi permasalahan diatas, perlu dibuat suatu sistim pengaman pintu secara otomatis yang hanya dapat mengenali wajah yang keterkaitan yang direset terlebih dahulu mendapatkan hak akses

masuk rumah modern. Pada tesis ini, dibuat suatu sistem identifikasi wajah seseorang dengan metode JST (Jaringan Saraf Tiruan) menggunakan algoritma pola pengenalan imege wajah (face recognition) BackPropagation sebagai parameter untuk membuka pintu. Pelatihan dilakukan terhadap citra wajah yang berjarak konstan 30 cm dengan pencahayaan yang diatur konstan pada webcam TECGO.

Adapun sensor yang digunakan adalah sensor kamera webcam , dimana data output dari sensor, kemudian data input tersebut dibandingkan dengan nilai set point (data referensi foto wajah) untuk proses akses masuk rumah modern.

B. Rumusan Masalah

Kajian tentang sistem sekuritas rumah moderen atau biasa dikenal sebagai smart home yang mengoptimalisasi sistem jaringan saraf tiruan yang cukup luas cakupannya, maka perlu dilakukan pembatasan-pembatasan pada sejumlah permasalahan yang akan diteliti, diantaranya:

1. Pencitraan wajah dengan pencahayaan yang konstan pada sistem sekuriti akses masuk rumah moderen.
2. Pendeteksian wajah pada ke lima posisi dengan jarak konstan 30 cm.
3. Konsep dasar pengenalan pola berbasis Jaringan Saraf Tiruan dengan algoritma backpropagation.

C. Tujuan Penelitian

Penelitian ini bertujuan untuk :

1. Membuat suatu sistem sekuritas masuk rumah modern yang cerdas dengan memanfaatkan pola pencitraan wajah
2. Menganalisis teknik akuisisi data untuk suatu sistem akses pencitraan yang cerdas
3. Membuatan prototype dari sistem agar dapat teroptimalisasi kinerja sistem

D. Manfaat Penelitian

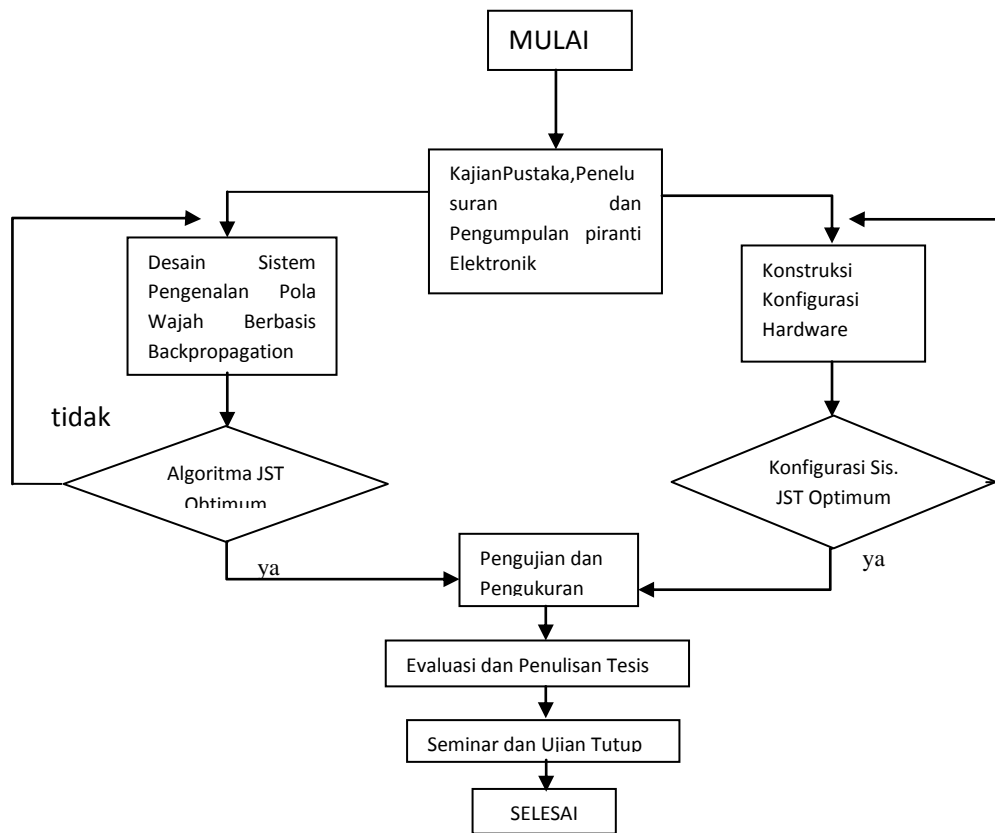
Dengan adanya penelitian ini diharapkan dapat memberikan situasi yang aman bagi penghuni rumah moderen dan adapun manfaat bagi penulis yaitu :

1. Menciptakan sistem sekuriti yang cerdas dan aman
2. Mendesain Teknik akuisisi data digital secara programmable
3. Dapat mengoptimalisasi kerja sistem sekuritas rumah moderen

E. Ruang Lingkup Penelitian

Sistem sekuritas rumah modern berbasis jaringan saraf tiruan bekerja secara optimal, Adapun cakupan batasan penelitian saya adalah :

1. Desain Kontrol mencakup: control pintu bagi pengguna dalam mengakses masuk ruangan,
2. Sekuriti akses masuk mencakup: deteksi citra wajah dengan metode JST backpropagation, dalam kondisi lima fariasi wajah.



Gambar 1. Diagram Alir riset JST untuk rumah modern

Pada gambar.1, menjelaskan tahapan-tahapan yang akan dilaksanakan dalam proses penelitian pada rumah modern, langkah pertama yaitu memulai dengan mengkaji materi sesuai dengan tesis dan mengumpulkan piranti elektronik. langkah kedua ada dua tahapan yaitu pertama mendesain sistem pengenalan pola wajah berbasis backpropagation, kemudian membuat algoritma JST optimum jika sudah diproses dan benar (ya) lanjut pengujian program jika tidak maka kembali ke desain sistem dan pengenalan pola wajah. Tahapan kedua konstruksi konfigurasi hardware sistem JST memproses konfigurasi sistem JST optimum jika tidak maka kembali ke konstruksi konfigurasi hardware

sistem JST. Jika terkonfigurasi sistem JST optimum maka proses pengujian dan pengukuran dilakukan. Langkah ketiga evaluasi dan penulisan tesis, Langkah keempat seminar dan ujian tutup setelah proses ini maka penelitian ini selesai dilaksanakan.

BAB II

TINJAUAN PUSTAKA

A. Road Map

Penelitian yang terkait dalam desain control sekuritas akses masuk rumah modern yang akan saya buat sebagai berikut:

1. Sistem deteksi dan verifikasi wajah pada sistem pembuka pintu (SOFTWARE). EEPIS Final Project.

Hasil yang diperoleh : Aplikasi penelitian ini mengenai deteksi wajah menggunakan metode segmentasi warna kulit, pencarian nilai bobot untuk data referensi menggunakan LVQ (Learning Vector Quantization) dan algoritma verifikasi (pengenalan pola wajah) wajah atau bukan wajah menggunakan metode Euclidian distance dan titik kritis distribusi. Kelemahan pada peneliti ini yaitu pada metode pendeteksi warna kulit, dimana warna kulit dari seseorang akan berubah sesuai dengan keadaan orang tersebut, contohnya jika orang tersebut berada dalam keadaan yang kurang sehat, maka warna kulit dari orang tersebut akan berubah, hal ini akan membuat sistim tidak dapat mengenali orang tersebut.

Sistem pengenalan wajah yang akan dibangun oleh penulis yaitu sistem yang dapat mendeteksi wajah orang yang ditetapkan pada inputnya berdasarkan bentuk wajah orang tersebut, disamping itu set

point (referensi) yang digunakan lebih dari lima posisi dengan 7 citra wajah untuk keakurasian data.

Leonardo , Kamajaya and Eru Puspita, Puspita and Ronny , Susetyoko (2009) (<http://www.cepis.its.edu/id/ta/776/>)

2. Sistem pengenalan wajah manusia real-time menggunakan algoritma jaringan saraf tiruan .

Hasil dari penelitian ini yaitu sistem dapat mengenal wajah dengan metode jaringan saraf tiruan (JST) mengenal wajah tampak depannya saja ketika tampak samping kiri dan samping kanan wajah program tidak dapat mengenal, saya akan membuat sistem dengan metode yang sama, dan di kembangkan lagi yaitu dapat mengenal wajah sesuai dengan yang di akses yaitu wajah biasa, wajah tersenyum, wajah serong kiri, wajah serong kanan, wajah dengan mulut terbuka aplikasinya lampu akan menyala jika wajah dikenal dan pintu akan tertutup jika wajah tidak dikenal dan pintu akan tertutup.

(Abdul Fadlil, Ikhson Hidayat, Sunardi) dapat di lihat pada alamat email (jurnal.uii.id/index.php/snati/article/view/881/842)

3. Prototype mengenal wajah melalui webcam dengan menggunakan algoritma principal component analysis (PCA) dan Linier discriminant analisis (LDA)

Hasil jurnal ini yaitu pengenalan wajah melalui webcam dengan metode PCA dan LDA.

Oleh : Jimmy E Purwanto

<http://journal.ipb.ac.id/index.php/artical/3202/2158>)

4. Sistem pengenalan wajah real-time ruang eigen dengan segmentasi berdasarkan warna kulit.

Penelitian ini dapat menghasilkan system yang dapat mengenal wajah, deteksi waja dengan metode jaringan syaraf tiruan (JST), Agus Bueno, Amad Ridha, Harief Batian)

<http://journal.ipb.ac.id/index.php/artical/3202/2158>)

Jurnal-jurnal penelitian ini menjadi acuan bagi saya untuk membuat desain suatu alat sistem sekuritas akses masuk rumah modern secara optimal, dari petunjuk dan saran dari peneliti diatas maka saya akan mengembangkan ide dari peneliti-peneliti sebelumnya, dengan judul “Optimalisasi Kinerja Sistem Sekuritas Ruamah Moderen Berbasis Jaringan Saraf Tiruan (JST)”.

B. Jaringan Syaraf Tiruan

Jaringan syaraf tiruan (*artificial neural networks*) atau disingkat JST adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi.

JST dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi, dengan asumsi :

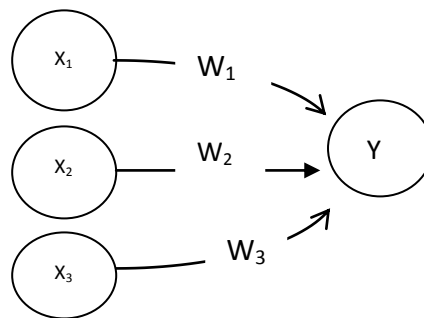
1. Pemrosesan informasi terjadi pada banyak elemen sederhana (*neuron*)
2. Sinyal dikirimkan diantara *neuron-neuron* melalui penghubung-penghubung

3. Penghubung antar *neuron* memiliki bobot yang akan memperkuat atau memperlemah sinyal
4. Untuk menentukan *output*, setiap *neuron* menggunakan fungsi aktivasi (biasanya bukan fungsi linier) yang dikenakan pada jumlahan *input* yang diterima. Besarnya *output* ini selanjutnya dibandingkan dengan suatu batas ambang.

JST ditentukan oleh tiga hal :

- a. Pola hubungan antar *neuron* (disebut arsitektur jaringan)
- b. Metode untuk menentukan bobot penghubung (disebut metode *training* / *learning* / algoritma)
- c. Fungsi aktivasi

Sebagai contoh, perhatikan *neuron* Y pada Gambar 2.



Gambar 2. Model Sederhana Jaringan Syaraf Tiruan

Y menerima *input* dari *neuron* x_1 , x_2 dan x_3 dengan bobot hubungan masing-masing adalah w_1 , w_2 dan w_3 . Ketiga impuls *neuron* yang ada dijumlahkan

$$\text{net} = x_1w_1 + x_2w_2 + x_3w_3 \quad (1)$$

Besarnya impuls yang diterima oleh Y mengikuti fungsi aktivasi $y = f(\text{net})$. Apabila nilai fungsi aktivasi cukup kuat, maka sinyal akan diteruskan. Nilai fungsi aktivasi (keluaran model jaringan) juga dapat dipakai sebagai dasar untuk merubah bobot.

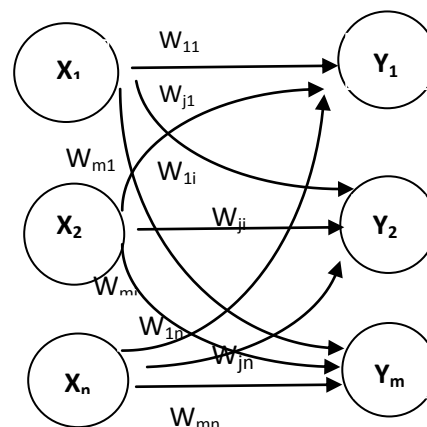
C. Arsitektur Jaringan

Beberapa arsitektur jaringan yang sering dipakai dalam jaringan syaraf tiruan antara lain :

1. Jaringan Layar Tunggal (*single layer network*)

JST dengan layar tunggal pertamakali dirancang oleh Widrow dan Hoff pada tahun 1960. Walaupun JST layar tunggal ini sangat terbatas penggunaannya, namun konsep dan gagasannya banyak dipakai oleh beberapa pakar untuk membuat model JST layar jamak.

Dalam jaringan ini, sekumpulan *input neuron* dihubungkan langsung dengan sekumpulan *output*. Dalam beberapa model (misal perceptron), hanya ada sebuah unit *neuron output*.



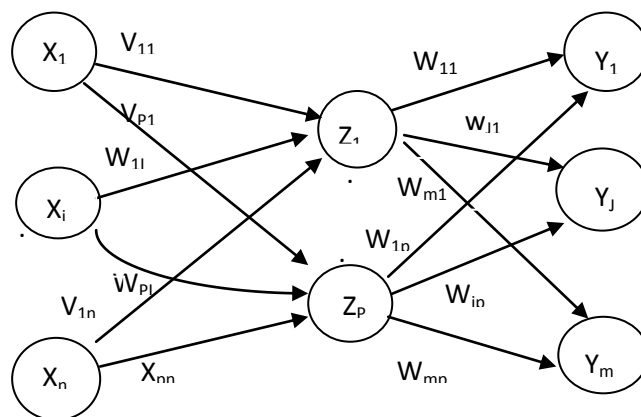
Gambar 3. Jaringan Layar Tunggal

Gambar 2.2. menunjukkan arsitektur jaringan dengan n unit *input* (x_1, x_2, \dots, x_n) dan m buah unit *output* (Y_1, Y_2, \dots, Y_m). Perhatikan bahwa dalam jaringan ini, semua unit *input* dihubungkan dengan semua unit *output*, meskipun dengan bobot yang berbeda-beda. Tidak ada unit *input* yang dihubungkan dengan unit *input* lainnya. Demikian pula dengan unit *output*.

Besarnya w_{ji} menyatakan bobot hubungan antara unit ke- i dalam *input* dengan unit ke- j dalam *output*. Bobot-bobot ini saling independen. Selama proses pelatihan, bobot-bobot tersebut akan dimodifikasi untuk meningkatkan keakuratan hasil.

2. Jaringan Layer Jamak (*multi layer network*)

Jaringan layer (lapisan jamak merupakan perluasan dari layer tunggal. Dalam jaringan ini, selain unit *input* dan *output*, ada unit-unit lain (sering disebut lapisan tersembunyi). Dimungkinkan pula ada beberapa layer tersembunyi. Sama seperti pada unit *input* dan *output*, unit-unit dalam satu layer tidak saling berhubungan.



Gambar 4. Jaringan Layer Jamak

Gambar 4. adalah jaringan dengan n buah unit *input* (x_1, x_2, \dots, x_n), sebuah layar tersembunyi yang terdiri dari p buah unit (z_1, \dots, z_p) dan m buah unit *output* (Y_1, Y_2, \dots, Y_m)

Jaringan layar jamak dapat menyelesaikan masalah yang lebih kompleks dibandingkan dengan layar tunggal, meskipun kadangkala proses pelatihan lebih kompleks dan lama.

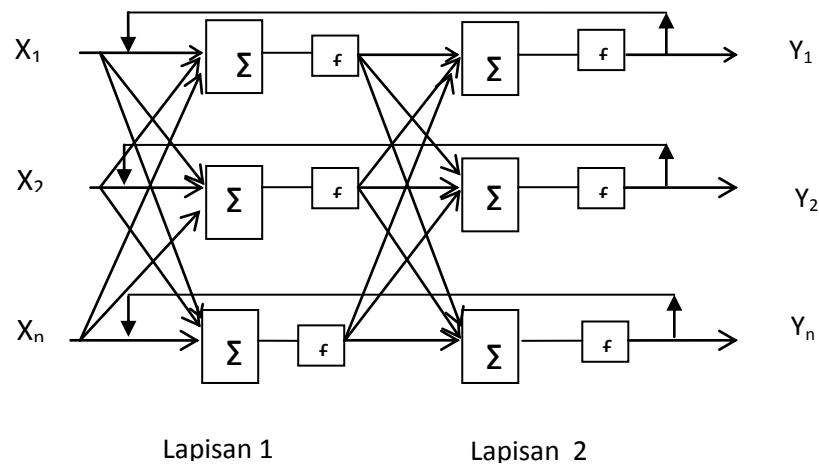
3. Model JST dua lapisan dengan umpan balik

Tokoh yang pertamakali mencetuskan ide tentang model jaringan syaraf tiruan dengan umpan balik adalah John Hopfield dari California Institute of Technology pada tahun 1982. Hopfield berpendapat bahwa kumpulan *neuron* tiruan dalam jumlah yang sangat besar dapat melakukan tugas-tugas tertentu.

Hopfield juga membandingkan antara jumlah *neuron* pada binatang dengan jumlah *neuron* diperkirakan sekitar 1000 buah dan bila dibandingkan dengan manusia, jumlah *neuron*-nya mencapai 100 trilyun buah. Sungguh jumlah yang sangat fantastis.

Dengan jumlah *neuron* yang sangat besar, JST memiliki sifat yaitu *fault tolerance*. Sifat ini mengandung maksud kerusakan sedikit atau sebagian pada sel-sel dalam jaringan tidak akan mempengaruhi *output* yang akan dikeluarkan.

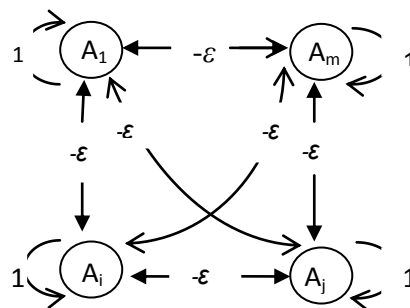
Model JST dua lapisan ini mempunyai sifat umpan balik, sehingga *output* yang dihasilkan akan mempengaruhi *input* yang akan masuk lagi ke dalam jaringan syaraf tersebut.



Gambar 5. Model JST Dua Lapisan Dengan Umpan Balik

4. Model JST lapisan kompetitif

Bentuk dari lapisan kompetitif merupakan bagian dari jumlah yang besar pada jaringan syaraf. Pada dasarnya, hubungan antara *neuron* satu dengan *neuron* yang lain pada lapisan kompetitif tidak ditunjukkan secara arsitektur pada beberapa jaringan syaraf. Contoh dari model atau arsitektur lapisan kompetitif dapat dilihat pada Gambar 6, dimana koneksi dari lapisan tersebut memiliki bobot $-\varepsilon$.

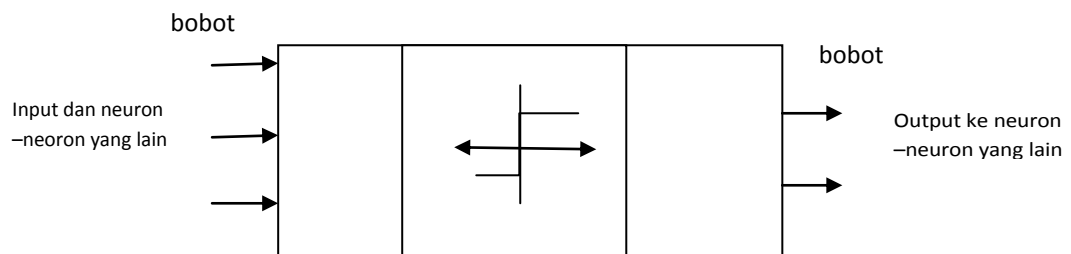


Gambar 6. Model JST Lapisan Kompetitif

D. Fungsi Aktivasi

Jaringan syaraf merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran.

Ada beberapa tipe jaringan syaraf, namun demikian, hampir semuanya memiliki komponen-komponen yang sama. Seperti halnya otak manusia, jaringan syaraf juga terdiri dari beberapa *neuron*, dan ada hubungan antara *neuron-neuron* tersebut. *Neuron-neuron* tersebut akan mentransformasikan informasi yang diterima melalui sambungan keluarnya menuju ke *neuron-neuron* yang lain. Pada jaringan syaraf, hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut. Gambar 7 menunjukkan struktur *neuron* pada jaringan syaraf.

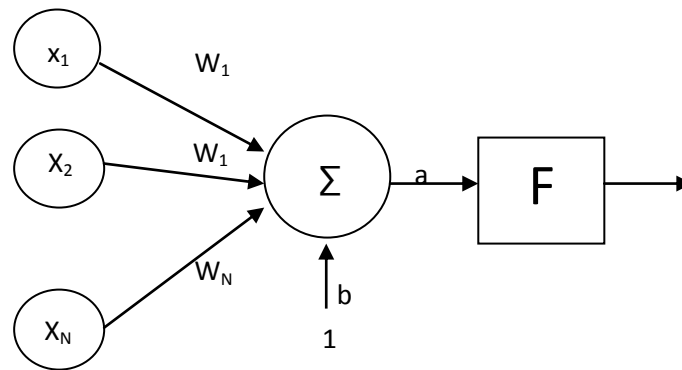


Gambar 7. Struktur Neuron Jaringan Syaraf

Jika dilihat, *neuron* buatan ini sebenarnya mirip dengan sel *neuron* biologis. *Neuron-neuron* buatan tersebut bekerja dengan cara yang sama pula dengan *neuron-neuron* biologis. Informasi (disebut dengan : *input*) akan dikirim ke *neuron* dengan bobot kedatangan tertentu. *Input* ini akan diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap *neuron*. Apabila *input* tersebut melewati suatu nilai ambang tertentu, maka *neuron* tersebut akan diaktifkan, tapi kalau tidak, maka *neuron* tersebut tidak akan diaktifkan. Apabila *neuron* tersebut diaktifkan, maka *neuron* tersebut akan mengirimkan *output* melalui bobot-bobot *output* nya ke semua *neuron* yang berhubungan dengannya. Demikian seterusnya.

Pada jaringan syaraf, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan (*layar*) yang disebut dengan lapisan *neuron* (*neuron layers*). Biasanya *neuron-neuron* pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya (kecuali lapisan *input* dan lapisan *output*). Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan *input* sampai ke lapisan *output* melalui lapisan yang lainnya, yang sering dikenal dengan nama lapisan tersembunyi (*hidden layer*). Tergantung pada algoritma pembelajarannya, bisa jadi informasi tersebut akan dirambatkan secara

mundur pada jaringan. Gambar 8 menunjukkan jaringan syaraf sederhana dengan fungsi aktivasi F.



Gambar 8. Fungsi Aktivasi Pada Jaringan Syaraf Sederhana

Pada Gambar 8. tersebut sebuah *neuron* akan mengolah N *input* (x_1, x_2, \dots, x_N) yang masing-masing memiliki bobot w_1, w_2, \dots, w_N dan bobot bias b , dengan rumus :

$$a = b + \sum_{i=1}^N x_i w_i \quad (2)$$

kemudian fungsi aktivasi F akan mengaktivasi a menjadi *output* jaringan y .

Ada beberapa fungsi aktivasi yang sering digunakan dalam jaringan syaraf tiruan. Fungsi Aktivasi yang digunakan pada *Backpropagation* antara lain :

- a. Fungsi sigmoid biner
- b. Fungsi sigmoid bipolar
- c. Fungsi linear

a. Fungsi sigmoid biner

Dalam *backpropagation*, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu : kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut sehingga sering dipakai adalah fungsi sigmoid biner yang memiliki range (0,1).

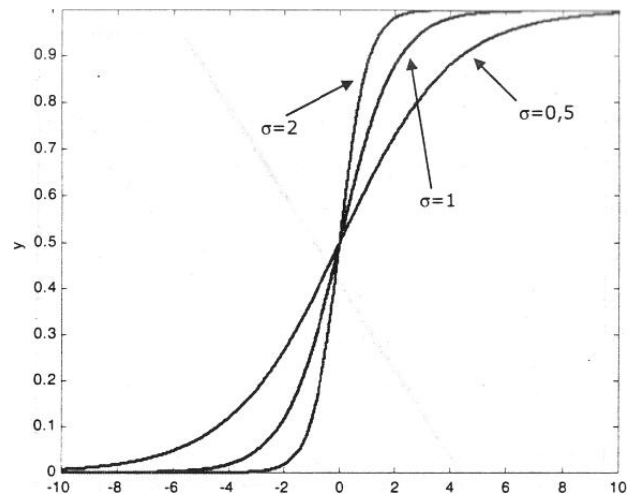
Fungsi ini digunakan untuk jaringan syaraf yang dilatih dengan menggunakan metode *backpropagation*. Fungsi sigmoid biner memiliki nilai pada *range* 0 sampai 1. Oleh karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai *output* yang terletak pada interval 0 sampai 1. Namun, fungsi ini bisa juga digunakan oleh jaringan syaraf yang nilai *output* nya 0 atau 1 (Gambar 9).

Fungsi sigmoid biner dirumuskan sebagai :

$$y = f(x) = \frac{1}{1 + e^{-\alpha x}} \quad (3)$$

dengan :

$$f'(x) = \sigma(x) (1 - \sigma(x)) \quad (4)$$



Gambar 9. Fungsi Aktivasi Sigmoid Biner

b. Fungsi sigmoid bipolar

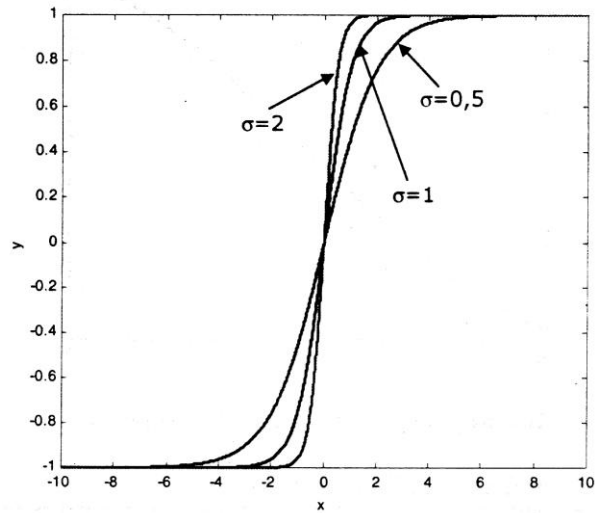
Fungsi sigmoid bipolar hampir sama dengan fungsi sigmoid biner, hanya saja *output* dari fungsi ini memiliki *range* antara 1 sampai -1 (Gambar 10).

Fungsi sigmoid bipolar dirumuskan sebagai :

$$y = f(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (5)$$

dengan:

$$f'(x) = \frac{\sigma}{2} [1 + f(x)][1 - f(x)]$$



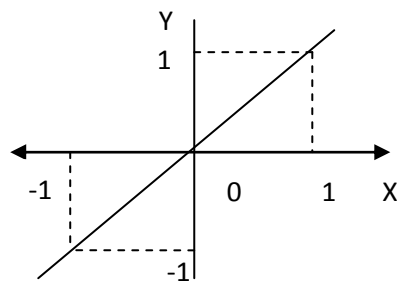
Gambar 10. Fungsi Aktivasi Sigmoid Bipolar

c. Fungsi linear (identitas)

Fungsi linear memiliki nilai *output* yang sama dengan nilai *input* (Gambar 11).

Fungsi linear dirumuskan sebagai :

$$y = x$$



Gambar 11. Fungsi Aktivasi Linear (Identitas)

E. Backpropagation

Kelemahan JST yang terdiri dari layer tunggal membuat perkembangan JST menjadi terhenti pada sekitar tahun 1970 an. Penemuan *backpropagation* yang terdiri dari beberapa layer membuka kembali cakrawala. Terlebih setelah berhasil ditemukannya berbagai aplikasi yang dapat diselesaikan dengan *backpropagation*, membuat JST semakin diminati orang.

JST dengan layer tunggal memiliki keterbatasan dalam pengenalan pola. Kelemahan ini bisa ditanggulangi dengan menambahkan satu/beberapa layer tersembunyi diantara layer masukan dan keluaran. Meskipun penggunaan lebih dari satu layer tersembunyi memiliki kelebihan manfaat untuk beberapa kasus, tapi pelatihannya memerlukan waktu yang lama. Maka umumnya orang mulai mencoba dengan sebuah layer tersembunyi lebih dahulu.

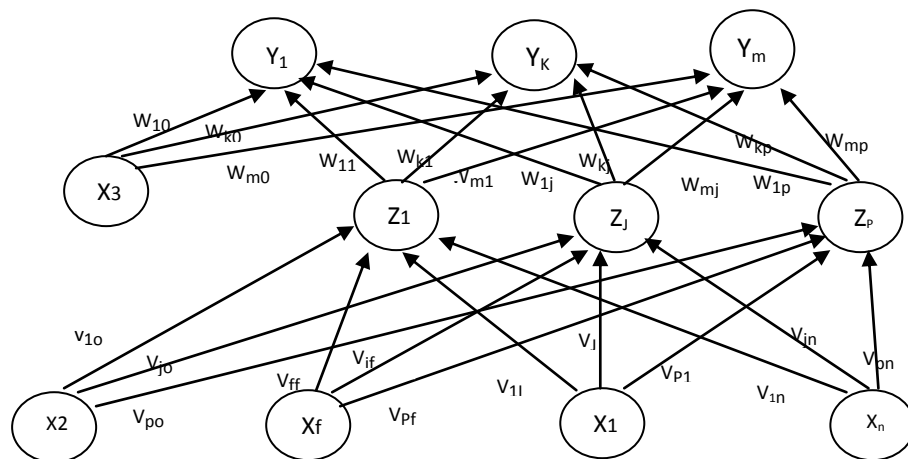
Seperti halnya model JST lain, *Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan.

1. Arsitektur Backpropagation

Backpropagation memiliki beberapa unit yang ada dalam satu atau lebih layer tersembunyi. Gambar 12 adalah arsitektur *backpropagation*

dengan n buah masukan (ditambah sebuah bias), sebuah layar tersembunyi yang terdiri dari p unit (ditambah sebuah bias), serta m buah unit keluaran.

v_{ji} merupakan bobot garis dari unit masukan x_i ke unit layar tersembunyi z_j (v_{j0} merupakan bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi z_j). w_{kj} merupakan bobot dari unit layar tersembunyi z_j ke unit keluaran y_k (w_{k0} merupakan bobot dari bias di layar tersembunyi ke unit keluaran z_k).



Gambar 12. Arsitektur Backpropagation

2. Neuron

Neuron adalah unit pemroses informasi yang menjadi dasar dalam pengoperasian jaringan syaraf tiruan. *Neuron* terdiri dari tiga elemen pembentuk :

- a. Himpunan unit-unit yang dihubungkan dengan jalur koneksi. Jalur-jalur tersebut memiliki bobot/kekuatan yang berbeda-beda. Bobot yang bernilai positif akan memperkuat sinyal dan yang bernilai negatif akan memperlemah sinyal yang dibawanya. Jumlah, struktur dan pola hubungan antar unit-unit tersebut akan menentukan arsitektur jaringan (dan juga model jaringan yang terbentuk).
- b. Suatu unit penjumlahan yang akan menjumlahkan *input-input* sinyal yang sudah dikalikan dengan bobotnya.
 Misalkan x_1, x_2, \dots, x_m adalah unit-unit *input* dan $w_{j1}, w_{j2}, \dots, w_{jm}$ adalah bobot penghubung dari unit-unit tersebut ke unit keluaran Y_j , maka unit penjumlah akan memberikan keluaran sebesar $u_j = x_1w_{j1} + x_2w_{j2} + \dots + x_mw_{jm}$
- c. Fungsi aktivasi yang akan menentukan apakah sinyal dari *input neuron* akan diteruskan ke *neuron* lain atautkah tidak.

3. Threshold

Threshold adalah suatu perubahan antara *gray-level image* dan *bilevel image*. *Bilevel image* terdiri dari banyak informasi yang penting dari sebuah *image* (contoh : angka, posisi dan bentuk objek), tetapi tidak dapat dibandingkan dengan informasi dari *gray-level image*. Kebanyakan dari *time pixels* dengan *gray levels* yang sama mempunyai objek yang sama juga. Selanjutnya, pengelompokan *image* oleh *pixel gray-level* dapat

memperkecil dan memudahkan beberapa pengoperasian proses *image* seperti membentuk *recognition* dan *classification*.

Beberapa operasi *thresholding* yang *essential* akan diseleksi oleh sebuah hasil tunggal *threshold*. Seluruh *gray-levels* menunjukkan hasil yang dapat diklasifikasikan menjadi *black* (0), dan disekelilingnya adalah *white* (1). Untuk waktu tertentu, bagian *image* tidak mungkin menjadi objek dan *background* dengan hasil tunggal *threshold* disebabkan oleh adanya *noise*. Pendekatan percobaan yang mudah harus menggunakan tujuan *gray-level* dalam *image* seperti *threshold*. Ini dapat disebabkan sebagian *pixels* menjadi *white* dan bagian lain menjadi *black*.

4. Pelatihan Standar Backpropagation

Pelatihan *Backpropagation* meliputi tiga fase. Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari layar masukan hingga layar keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur. Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di lapisan keluaran. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi.

a. Fase I : Propagasi maju

Selama propagasi maju, sinyal masukan ($=x_i$) dipropagasikan ke layar tersembunyi menggunakan fungsi aktivasi yang ditentukan.

Keluaran dari setiap unit layar tersembunyi ($=z_j$) tersebut selanjutnya dipropagasikan maju lagi ke layar tersembunyi di atasnya menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan keluaran jaringan ($=y_k$).

Berikutnya, keluaran jaringan ($=y_k$) dibandingkan dengan target yang harus dicapai ($=t_k$). Selisih $t_k - y_k$ adalah kesalahan yang terjadi. Jika kesalahan ini lebih kecil dari batas toleransi yang ditentukan, maka iterasi dihentikan. Akan tetapi apabila kesalahan masih lebih besar dari batas toleransinya, maka bobot setiap garis dalam jaringan akan dimodifikasi untuk mengurangi kesalahan yang terjadi.

b. Fase II : Propagasi mundur

Berdasarkan kesalahan $t_k - y_k$, dihitung faktor δ_k ($k = 1, 2, \dots, m$) yang dipakai untuk mendistribusikan kesalahan di unit y_k ke semua unit tersembunyi yang terhubung langsung dengan y_k . δ_k juga dipakai untuk mengubah bobot garis yang berhubungan langsung dengan unit keluaran.

Dengan cara yang sama, dihitung faktor δ_j di setiap unit di layar tersembunyi sebagai dasar perubahan bobot semua garis yang berasal dari unit tersembunyi di layar di bawahnya. Demikian seterusnya hingga semua faktor δ di unit tersembunyi yang berhubungan langsung dengan unit masukan dihitung.

c. Fase III : Perubahan bobot

Setelah semua faktor δ dihitung, bobot semua garis dimodifikasi bersamaan. Perubahan bobot suatu garis didasarkan atas faktor δ *neuron* di layar atasnya. Sebagai contoh, perubahan bobot garis yang menuju ke layar keluaran didasarkan atas δ_k yang ada di unit keluaran.

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi yang diijinkan.

Algoritma Backpropagation

1. Inisialisasi bobot (ambil bobot awal dengan nilai *random* yang cukup kecil).
2. Kerjakan langkah-langkah berikut selama kondisi berhenti bernilai salah.
3. Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan :

Feedforward

- a. Tiap-tiap unit *input* (X_i , $i=1,2,3,\dots,n$) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi).

- b. Tiap-tiap unit pada suatu lapisan tersembunyi (Z_j , $j=1,2,3,\dots,p$) menjumlahkan sinyal-sinyal *input* terbobot :

$$z_in_j = b1_j + \sum_{i=1}^n x_i v_{ij}$$

gunakan fungsi aktivasi untuk menghitung sinyal *output* :

$$z_j = f(z_in_j)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit *output*).

- c. Tiap-tiap unit *output* (Y_k , $k=1,2,3,\dots,m$) menjumlahkan sinyal-sinyal *input* terbobot.

$$y_in_k = b2_k + \sum_{i=1}^n z_i w_{jk} \quad (6)$$

gunakan fungsi aktivasi untuk menghitung sinyal *output* :

$$y_k = f(y_in_k) \quad (7)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit *output*).

Catatan :

Langkah (b) dilakukan sebanyak jumlah lapisan tersembunyi.

Backpropagation

- d. Tiap-tiap unit *output* (Y_k , $k=1,2,3,\dots,m$) menerima target pola yang berhubungan dengan pola *input* pembelajaran, hitung informasi errornya :

$$\delta 2_k = (t_k - y_k) f'(y_in_k) \quad (8)$$

$$\varphi_{2jk} = \delta_k z_j \quad (9)$$

$$\beta_{2k} = \delta_k \quad (10)$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai w_{jk}) :

$$\Delta w_{jk} = \alpha \varphi_{2jk} \quad (11)$$

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai b_{2k}) :

$$\Delta b_{2k} = \alpha \beta_{2k} \quad (12)$$

Langkah (d) ini juga dilakukan sebanyak jumlah lapisan tersembunyi, yaitu menghitung informasi error dari suatu lapisan tersembunyi ke lapisan tersembunyi sebelumnya.

- e. Tiap-tiap unit tersembunyi (Z_j , $j=1,2,3,\dots,p$) menjumlahkan delta *input* (dari unit-unit yang berada pada lapisan di atasnya) :

$$\delta_{in_j} = \sum_{k=1}^m \delta_{2k} w_{jk} \quad (13)$$

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error* :

$$\delta_{1j} = \delta_{in_j} f'(z_{in_j}) \quad (15)$$

$$\varphi_{1ij} = \delta_{1j} x_j \quad (16)$$

$$\beta_{1j} = \delta_{1j} \quad (17)$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai v_{ij}) :

$$\Delta v_{ij} = \alpha \varphi_{1ij} \quad (18)$$

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai b_{1j}) :

$$\Delta b_{1j} = \alpha \beta_{1j} \quad (19)$$

f. Tiap-tiap unit *output* (Y_k , $k=1,2,3,\dots,m$) memperbaiki bias dan bobotnya ($j=0,1,2,\dots,p$) :

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (20)$$

$$b_{2k}(\text{baru}) = b_{2k}(\text{lama}) + \Delta b_{2k} \quad (21)$$

Tiap-tiap unit tersembunyi (Z_j , $j=1,2,3,\dots,p$) memperbaiki bias dan bobotnya ($i=0,1,2,\dots,n$) :

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (22)$$

$$b_{1j}(\text{baru}) = b_{1j}(\text{lama}) + \Delta b_{1j} \quad (23)$$

4. Tes kondisi berhenti.

5. Optimalitas Arsitektur Backpropagation

Masalah utama yang dihadapi dalam *Backpropagation* adalah lamanya iterasi yang harus dilakukan. *Backpropagation* tidak dapat memberikan kepastian tentang berapa *epoch* yang harus dilalui untuk mencapai kondisi yang diinginkan. Oleh karena itu orang berusaha meneliti bagaimana parameter-parameter jaringan dibuat sehingga menghasilkan jumlah iterasi yang relatif lebih sedikit.

a. Inisialisasi bobot awal secara *random*

Pemilihan bobot awal sangat mempengaruhi jaringan syaraf dalam mencapai minimum global (atau mungkin hanya lokal saja)

terhadap nilai *error*, serta cepat tidaknya proses pelatihan menuju kekonvergenan. Apabila nilai bobot awal terlalu besar, maka *input* ke setiap lapisan tersembunyi atau lapisan *output* akan jatuh pada daerah dimana turunan fungsi sigmoidnya akan sangat kecil. Sebaiknya, apabila nilai bobot awal terlalu kecil, maka *input* ke setiap lapisan tersembunyi atau lapisan *output* akan sangat kecil, yang akan menyebabkan proses pelatihan akan berjalan sangat lambat. Biasanya bobot awal diinisialisasi secara *random* dengan nilai antara -0.5 sampai 0.5 (atau -1 sampai 1, atau interval yang lainnya).

b. Jumlah unit tersembunyi

Hasil teoritis yang didapat menunjukkan bahwa jaringan dengan sebuah layar tersembunyi sudah cukup bagi *Backpropagation* untuk mengenali sembarang pola antara masukan dan target dengan tingkat ketelitian yang ditentukan. Akan tetapi penambahan jumlah layar tersembunyi kadangkala membuat pelatihan lebih mudah.

Dalam propagasi maju, keluaran harus dihitung untuk tiap layar, dimulai dari layar tersembunyi paling bawah (terdekat dengan masukan). Sebaliknya, dalam propagasi mundur, faktor δ perlu dihitung untuk tiap layar tersembunyi, dimulai dari layar keluaran.

c. Jumlah pola pelatihan

Tidak ada kepastian tentang berapa banyak pola yang diperlukan agar jaringan dapat dilatih dengan sempurna. Jumlah pola yang dibutuhkan dipengaruhi oleh banyaknya bobot dalam jaringan

serta tingkat akurasi yang diharapkan. Aturan kasarnya dapat ditentukan berdasarkan rumusan :

$$\text{Jumlah pola} = \text{Jumlah bobot} / \text{tingkat akurasi}$$

Untuk jaringan dengan 80 bobot dan tingkat akurasi 0.1, maka 800 pola masukan diharapkan akan mampu mengenali dengan benar 90 % pola diantaranya.

d. Lama iterasi

Tujuan utama penggunaan *Backpropagation* adalah mendapatkan keseimbangan antara pengenalan pola pelatihan secara benar dan respon yang baik untuk pola lain yang sejenis (disebut data pengujian). Jaringan dapat dilatih terus menerus hingga semua pola pelatihan dikenali dengan benar. Akan tetapi hal itu tidak menjamin jaringan akan mampu mengenali pola pengujian dengan tepat. Jadi tidaklah bermanfaat untuk meneruskan iterasi hingga semua kesalahan pola pelatihan = 0.

Umumnya data dibagi menjadi dua bagian, yaitu pola data yang dipakai sebagai pelatihan dan data yang dipakai untuk pengujian. Perubahan bobot dilakukan berdasarkan pola pelatihan. Akan tetapi selama pelatihan (misal setiap 10 *epoch*), kesalahan yang terjadi dihitung berdasarkan semua data (pelatihan dan pengujian). Selama kesalahan ini menurun, pelatihan terus dijalankan. Akan tetapi jika kesalahannya sudah meningkat, pelatihan tidak ada gunanya untuk diteruskan lagi. Jaringan sudah mulai mengambil sifat yang hanya

dimiliki secara spesifik oleh data pelatihan (tapi tidak dimiliki oleh data pengujian) dan sudah mulai kehilangan kemampuan melakukan generalisasi.

6. Variasi Backpropagation

Disamping model standar *Backpropagation*, kini sudah berkembang berbagai variasinya. Variasi tersebut bisa berupa model *Backpropagation* yang digunakan untuk keperluan khusus, atau teknik modifikasi bobot untuk mempercepat pelatihan dalam kasus tertentu.

a. Momentum

Pada standar *Backpropagation*, perubahan bobot didasarkan atas gradien yang terjadi untuk pola yang dimasukkan saat itu. Modifikasi yang dapat dilakukan adalah melakukan perubahan bobot yang didasarkan atas arah gradien pola terakhir dan pola sebelumnya (disebut momentum) yang dimasukkan. Jadi tidak hanya pola masukan terakhir saja yang diperhitungkan.

Penambahan momentum dimaksudkan untuk menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda dengan yang lain (*outlier*). Apabila beberapa data terakhir yang diberikan ke jaringan memiliki pola serupa (berarti arah gradien sudah benar), maka perubahan bobot dilakukan secara cepat. Namun apabila data terakhir yang dimasukkan memiliki pola yang berbeda dengan pola sebelumnya, maka perubahan dilakukan secara lambat.

Dengan penambahan momentum, bobot baru pada waktu ke (t+1) didasarkan atas bobot pada waktu t dan (t-1). Disini harus ditambahkan 2 variabel baru yang mencatat besarnya momentum untuk 2 iterasi terakhir. Jika μ adalah konstanta ($0 \leq \mu \leq 1$) yang menyatakan parameter momentum maka bobot baru dihitung berdasarkan persamaan :

$$w_{kj}^{(+1)} = w_{kj}^{(+)} + \alpha \delta_k z_j + \mu (w_{kj}^{(+)} - w_{kj}^{(-1)})$$

(24)

dan

$$v_{ji}^{(+1)} = v_{ji}^{(+)} + \alpha \delta_j x_i + \mu (v_{ji}^{(+)} - v_{ji}^{(-1)}) \quad (25)$$

b. Delta - Bar - Delta

Dalam standar *Backpropagation*, laju pemahaman (α) merupakan suatu konstanta yang dipakai dalam seluruh iterasinya. Perubahan dapat dilakukan dengan memberikan laju pemahaman yang berbeda-beda untuk setiap bobotnya (atau bahkan laju pemahaman yang berbeda-beda untuk tiap bobot dalam tiap iterasinya). Apabila perubahan bobot berada dalam arah yang sama dalam beberapa pola terakhir (dapat dilihat dari tanda suku $\delta_k z_j$ yang selalu sama), maka laju pemahaman yang bersesuaian dengan bobot w_{kj} ditambah. Sebaliknya apabila arah perubahan bobot dua pola terakhir berbeda (ditandai dengan suku $\delta_k z_j$ yang berselang-seling positif - negatif) maka laju pemahaman untuk bobot tersebut harus dikurangi.

Perubahan bobot dalam aturan delta - bar - delta adalah sebagai berikut :

$$w_{kj}(k+1) = w_{kj}(k) + \alpha_{kj}(k) \delta_k z_j \quad (26)$$

F. Pemrograman Backpropagation Dengan Matlab

1. Membentuk Jaringan

a. Inisialisasi Jaringan

Langkah pertama yang harus dilakukan untuk memprogram *backpropagation* dengan Matlab adalah membuat inisialisasi jaringan. Perintah yang dipakai untuk membentuk jaringan adalah `newff` yang formatnya adalah sebagai berikut :

```
net = newff(PR,[S1 S2...SN],[TF1 TF2...TFN],BTF,BLF,PF)
```

Dengan

net = jaringan Backpropagation yang terdiri dari n lapisan

PR = matriks ordo Rx2 yang berisi nilai minimum dan maksimum R buah elemen masukannya

Si (i=1,2,...,n) = jumlah unit pada layar ke-i (i=1,2,...,n)

Tfi (i=1,2,...,n) = fungsi aktivasi yang dipakai pada layar ke-i (i=1,2,...,n).

Default = `tansig` (sigmoid bipolar)

BTF = fungsi pelatihan jaringan. Default = `traingdx`

BLF = fungsi perubahan bobot/bias. Default = `learngdm`

PF = fungsi perhitungan error. Default = `mse`

Beberapa fungsi aktivasi yang dipakai Matlab dalam pelatihan *backpropagation* adalah :

a. tansig (sigmoid bipolar) $f(\text{net}) = \frac{2}{1+e^{-\text{net}}} - 1$. Fungsi ini adalah

default yang dipakai. Fungsi sigmoid bipolar memiliki *range* [-1,1]

b. logsig (sigmoid biner) $f(\text{net}) = \frac{1}{1+e^{-\text{net}}}$. Fungsi sigmoid biner

memiliki bentuk serupa dengan sigmoid bipolar, hanya rangenya adalah [0,1]

c. purelin (fungsi identitas) $f(\text{net}) = \text{net}$

Pelatihan yang dilakukan dalam Matlab dapat menggunakan berbagai fungsi, tujuannya adalah mempercepat pelatihan. Fungsi *default* yang dipakai oleh Matlab adalah *traingdx*. Dalam fungsi ini, perubahan bobot dilakukan dengan menambahkan momentum. Perubahan dilakukan dengan memperhatikan perubahan bobot pada iterasi sebelumnya. Disamping itu laju pemahaman (*learning rate* = α) bukan merupakan konstanta yang tetap, tetapi dapat berubah-ubah selama iterasi.

Umumnya, pelatihan *backpropagation* dalam Matlab dilakukan secara berkelompok (*batch training*). Semua pola dimasukkan dulu, baru kemudian bobot diubah. Dalam pelatihan berkelompok, semua data masukan harus diletakkan dalam sebuah matriks.

1. Inisialisasi Bobot

Setiap kali membentuk jaringan *backpropagation*, Matlab akan memberi nilai bobot dan bias awal dengan bilangan acak kecil. Bobot dan bias ini akan berubah setiap kali dibentuk jaringan. Akan tetapi jika diinginkan memberi bobot tertentu, dapat dilakukan dengan memberi nilai pada `net.IW`, `net.LW` dan `net.b`.

Perhatikan perbedaan antara `net.IW` dan `net.LW`. `net.IW{j,i}` digunakan sebagai variabel untuk menyimpan bobot dari unit masukan layar *i* ke unit tersembunyi (atau unit keluaran) layar *j*. Karena dalam *backpropagation*, unit masukan hanya terhubung dengan layar tersembunyi paling bawah, maka bobotnya disimpan dalam `net.IW{1,1}`.

Sebaliknya, `net.LW{k,j}` dipakai untuk menyimpan bobot dari unit di layar tersembunyi ke-*j* ke unit di lapisan tersembunyi ke-*k*. Sebagai contoh, `net.LW{2,1}` adalah penyimpan bobot dari layar tersembunyi paling bawah (lapisan tersembunyi ke-1) ke layar tersembunyi di atasnya (lapisan tersembunyi ke-2).

2. Simulasi Jaringan

Perintah `sim` digunakan pada *backpropagation* untuk menghitung keluaran jaringan berdasarkan arsitektur, pola masukan dan fungsi aktivasi yang dipakai.

`[Y,Pf,Af,E,perf] = sim(net,P,Pi,Ai,T)`

Dengan parameter masukan

net : nama jaringan dalam perintah newff

P : vektor masukan jaringan

Pi : kondisi delay awal masukan. Default = zeros

Ai : kondisi delay layar. Default = zeros

T : vektor target jaringan. Default = zeros

Dan parameter hasil

Y : keluaran jaringan

Pf : kondisi akhir delay masukan

Af : kondisi akhir delay layar

E : error jaringan = T-Y

Perf: unjuk kerja jaringan

Pi, Ai, Pf, Af hanya dipakai bagi jaringan yang memiliki *delay* masukan dan layar. Untuk sekedar menghitung keluaran jaringan, dapat dipakai statemen sederhana :

y = sim (net,p);

Perhatikan bahwa untuk menghitung keluaran jaringan, tidak perlu diketahui targetnya. Akan tetapi jika ingin dihitung *error* yang terjadi (selisih antara target dengan keluaran jaringan), maka harus diketahui targetnya.

G. Konsep Pemrosesan Citra

Tujuan utama dari pemrosesan citra adalah memperbaiki kualitas suatu gambar agar mudah diinterpretasikan oleh mata manusia dan

untuk mengolah informasi pada gambar untuk pengenalan objek secara otomatis oleh suatu mesin. Dalam pengenalan wajah dan pengenalan pola wajah diproses dicapture sampai pada ekstraksi cirinya. Pada pengenalan wajah proses capture ini sangat menentukan tingkat kesulitan dalam komputasinya, salah satunya bahwa dalam setiap proses capture ternyata cahaya, warna, posisi, jarak dan kemiringan menjadi suatu masalah yang diperhatikan. (Alan Budi (2007) Pengolahan Citra

1. Elemen Dasar Suatu Citra

a. Pixel (picture element)

Gambar yang bertipe bitmap tersusun dari pixel-pixel. pixel disebut juga dengan dot. Ukuran relative kecil pembentuk gambar bitmap.

Banyaknya pixel tiap satuan luas tergantung pada bit depth yang dipakai. Semakin banyak jumlah piksel tiap satuan luas, semakin baik kualitas gambar yang dihasilkan dan tentu semakin besar ukuran filenya.

b. Bit depth

Bit depth (kedalam warna)/ *pixel depth/ color depth*. Bit depth menentukan berapa banyak informasi warna yang tersedia untuk ditampilkan/dicetak dalam setiap pixel, semakin besar nilainya semakin bagus kualitas gambar yang dihasilkan, tentu ukuran juga semakin besar.

c. Resolusi

Resolusi adalah jumlah pixel persatuan luas yang ada disuatu gambar. Satuan pixel sering dipakai adalah dpi (*dot per inch*)/ppi (*pixel per inch*). Satuan dpi menentukan jumlah pixel yang ada setiap satu satuan luas. Yang dalam hal ini adalah satu inch kuadrat. Resolusi sangat berpengaruh pada detil dan perhitungan gambar.

d. Kecerahan dan Kontras

Kecerahan (*brightness*) adalah intensitas yang terjadi pada suatu titik citra. Sistem visual manusia mampu menyesuaikan diri dengan tingkat kecerahan dengan jangkauan.

2. Operasi Pengolahan Citra

Pengambilan citra wajah di klasifikasikan secara umum yaitu :

a. Peningkatan kualitas citra (*image enhancement*)

Tujuannya untuk memperbaiki kualitas citra dibandingkan dengan citra semula. Contoh operasi perbaikan citra :

1. Perbaikan kontras gelap/terang
2. Perbaikan tepian objek (*edge enhacement*)
3. Penajaman (*sharpening*)
4. Penapisan derau (*noise filtering*)

b. Perbaikan citra (*imge restoration*)

Bertujuan menghilangkan cacat pada citra, bertujuan pemugaran citra hamper sama dengan operasi perbaikan citra. yaitu :

1. Penghilang kesamaran (*debluring*)
2. Penghilang derau (*noise*)
3. Pelembutan citra (*smoothing*)

c. Segmentasi Citra (*image analysis*)

Pengenalan pola yang sesuai kriteria dari segmen yang di operasikan .

d. Analisis citra (*image analysis*)

Bertujuan untuk menghitung besaran kuantitatif dari citra untuk menghasilkan deskripsinya. dengan cara :

1. Pendeteksi Objek (*egde detection*)
2. Ekstraksi batas (*boundary*)
3. Representasi area wilayah (*region*)

e. Rekonstruksi citra (*image reconstruction*)

Bertujuan untuk membentuk ulang objek dari beberapa citra hasil proyeksi. (8)