



UNIVERSITAS INDONESIA

**Pengenalan Wajah dengan Metode
Backpropagation Menggunakan
Kamera CCTV Inframerah**

SKRIPSI

**DEDI DARMAWAN
0706199211**

**FAKULTAS TEKNIK
PROGRAM SARJANA EKSTENSI
DEPOK
JUNI 2010**



UNIVERSITAS INDONESIA

**Pengenalan Wajah dengan Metode
Backpropagation Menggunakan Kamera
CCTV Inframerah**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

**DEDI DARMAWAN
0706199211**

**FAKULTAS TEKNIK
PROGRAM SARJANA EKSTENSI
DEPOK
JULI 2010**

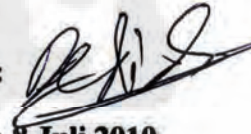
PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Dedi Darmawan

NPM : 0706199211

Tanda Tangan :



Tanggal : 8 Juli 2010

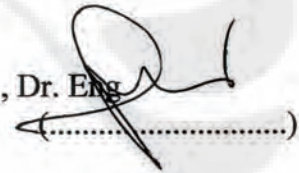
HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Dedi Darmawan
NPM : 0706199211
Program Studi : Teknik Elektro
Judul Skripsi : **PENGENALAN WAJAH DENGAN METODE
BACKPROPAGATION MENGGUNAKAN
KAMERA CCTV INFRAMERAH**

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Elektro Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Prof. Benyamin Kusumoputro, Drs., M.Eng., Dr. Eng



Penguji : Dr. Abdul Muis ST, M.Eng



Penguji : Dr. Ir. Dodi Sudiana M.Eng



Ditetapkan di : Depok

Tanggal : 8 Juli 2010

UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan kehadirat ALLAH SWT, karena atas berkah dan rahmat-Nya, penulis dapat menyelesaikan Seminar ini. Penulisan Seminar ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Penulis menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan Tugas Akhir ini, sangatlah sulit bagi penulis untuk menyelesaikan Tugas Akhir ini. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

- (1) Prof. Benyamin Kusumoputro, Drs., M.Eng., Dr. Eng selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam menyelesaikan ini;
- (2) Kedua orang tua dan keluarga penulis yang telah memberikan dukungan material dan moral;
- (3) Sahabat dan teman-teman yang telah banyak membantu penulis dalam menyelesaikan Seminar ini.

Akhir kata, penulis berharap ALLAH SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga Tugas Akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, 8 Juli 2010

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Dedi Darmawan
NPM : : 0706199211
Program Studi : Teknik Elektro
Departemen : Elektro
Fakultas : Teknik
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

**“PENGENALAN WAJAH DENGAN METODE BACKPROPAGATION
MENGUNAKAN KAMERA CCTV INFRAMERAH”**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 8 Juli 2010

Yang menyatakan

(Dedi Darmawan)

ABSTRAK

Nama : Dedi Darmawan
Program studi : Teknik Elektro
Judul : Pengenalan Wajah Dengan Metode Backpropagation
Menggunakan Kamera CCTV Inframerah

Penggunaan kamera CCTV Inframerah sangat membantu dalam penerapan sistem keamanan, dengan demikian pengambilan gambar akan lebih mudah terutama jika ruangan dalam kondisi gelap. Sehingga proses pengenalan wajah dalam kondisi gelap dapat dilakukan sebagaimana kondisi normal. Metode yang digunakan untuk proses pengenalan adalah Jaringan Saraf Tiruan. Jaringan Saraf Tiruan (JST) adalah suatu metode komputasi untuk memodelkan suatu sistem. Hal ini memungkinkan Jaringan Saraf Tiruan untuk melakukan pembelajaran terhadap sinyal yang diterima oleh sistem. Metode yang digunakan adalah *backpropagation* yang terdiri atas lapisan masukan, lapisan tersembunyi dan lapisan keluaran. Pada penelitian ini analisis yang dilakukan adalah training data dengan data matriks image serta menggunakan Matriks Kovarian untuk mempermudah proses inputan.

Kata kunci :
Kamrea Inframerah, Jaringan Saraf Tiruan, *backpropagation*.

ABSTARCT

Name : Dedi Darmawan
Study Program : Electrical Engineering
Title : Face Recognition With Backpropagation Method Using Infrared CCTV Cameras

the use of infrared CCTV camera is very helpfull in implementation of security system, because its capability to take images in the night so it can be used for identification process. The Method which performed in this identification process is Neural Network with back propagation. In this system is formed a network that consisting several input layers, hidden layers and output layers. The Data on input layer would be sent to hidden layer so the value of hidden weight can be obtained. From hidden layer, the value of hidden weight would be sent to output layer, so the value of output weight can be obtained. Difference between the output layer and the value of output weight is used to acknowledge error level in data identification. to obtain better identification result, the value of output weight is used as input data on input layer, so the same process is perfomed with better result. This is mentioned as back propagation process in Neural Network and this system is assesed capable to analyze the received data as input data.

Key word :
Infrared Camera, Neural Network, Backpropagation

DAFTAR ISI

PERYATAAN ORISINILITAS	ii
PENGESAHAN	iii
UCAPAN TERIMA KASIH	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan	2
1.3 Batasan Masalah	2
1.4 Metode Penulisan	2
1.5 Sistematika Penulisan	3
BAB 2 DASAR TEORI	4
2.1 Kamrea Inframerah	4
2.1.1 Inframerah	4
2.1.2 Penglihatan malam	5
2.1.3 Kamera CCTV Inframerah	6
2.2 Jaringan Saraf Tiruan	7
2.2.1 Sejarah Jaringan Saraf Tiruan	8
2.2.2 Neuron	10
2.2.3 Arsitektur Jaringan	11
2.2.4 Perceptron	13
2.2.5 Pembelajaran	13
2.2.6 Fungsi Aktifasi	14
2.2.7 Bias	17
2.2.8 Algoritma Backpropagation	19
BAB 3 PENGENALAN WAJAH DENGAN METODE BACKPROPAGATION MENGGUNAKAN KAMERA CCTV INFRAMERAH	23
3.1 Data Input	23
3.1.1 Pengumpulan database wajah	25
3.1.2 Matriks Kovarian	26
3.2 Pembentukan Jaringan Saraf Tiruan	28
3.2.1 Pelatihan	29
3.2.2 Proses Pengenalan	35
BAB 4 HASIL DAN ANALISA	38
4. 1 Hasil Pelatihan	38
4. 2 Pengujian	41
4.2.1 Pengujian dengan menggunakan kamera CCTV Inframerah	41

4.2.2 Pengujian dengan menggunakan kamera Visible	44
4.3 Analisa	47
4.3.1. Analisa pengujian dengan menggunakan kamera CCTV Inframerah	47
4.3.2. Analisa Pengujian dengan menggunakan kamera <i>Visible</i>	49
Bab 5 KESIMPULAN	50
DAFTAR ACUAN.....	51



DAFTAR GAMBAR

Gambar 2.1	Seekor anjing yang diambil dalam cahaya inframerah	4
Gambar 2.2	Gambar yang diambil dengan teropong dengan kemampuan night vision	5
Gambar 2.3	Bentuk kamera yang digunakan Kamera CCTV IR Outdoor JMK JK-212	6
Gambar 2.4	Hasil gambar menggunakan Kamera CCTV IR Outdoor JMK JK-212	7
Gambar 2.5	Gambar Pola Neural Network	8
Gambar 2.6	Jaringan Saraf Neuron	10
Gambar 2.7	Jaringan Saraf Tiruan <i>FeedForward</i>	11
Gambar 2.8	Jaringan Saraf Tiruan <i>BackPropagation</i>	12
Gambar 2.9	Bentuk Perceptron	13
Gambar 2.10	<i>Linear Transfer Function</i>	15
Gambar 2.11	Fungsi Tangga-Biner	16
Gambar 2.12	Fungsi Sigmoid (Logistik)-Biner	16
Gambar 2.13	Fungsi Sigmoid Bipolar	17
Gambar 2.14	Jaringan Saraf Tiruan dengan Bias.....	18
Gambar 2.15	Penggambaran Bias dengan Linearitas	18
Gambar 2.16	Penggambaran Bias dengan input	18
Gambar 3.1	Proses pengambilan sample wajah	24
Gambar 3.2	Contoh sample wajah yang sudah di <i>crop</i>	25
Gambar 3.3	Contoh sample wajah yang sudah di <i>resize</i> untuk database	25
Gambar 3.4	Penggabungan nilai matriks tiap image	25
Gambar 3.5	Plot Matrik data	26
Gambar 3.6	Plot data matrik Kovarian	27
Gambar 3.7	Diagram Alir Proses pembentukan Input	28
Gambar 3.8	Arsitektur komputasi balik	29
Gambar 3.9	Blok Diagram Fungsi Aktifasi	30
Gambar 3.10	Kurva Gradien Perbaikan Bobot	31
Gambar 3.11	Proses Perbaikan Bobot Jaringan Saraf Tiruan	32
Gambar 3.12	Diagram Alir Proses Pelatihan Jaringan Saraf Tiruan.....	34
Gambar 3.13	Diagram Alir Proses Pengenalan.....	36
Gambar 4.1	Grafik Fungsi Error untuk 10.000 Epoch.....	38
Gambar 4.2	Grafik Fungsi Error dengan Alpha 0.9	39
Gambar 4.3	Grafik Fungsi dengan alpha 0.6 untuk 2.000 epoch.....	39
Gambar 4.4	Grafik Fungsi Error yang didapat dari kesalahan proses pembelajaran	40
Gambar 4.5	Gambar sample uji dengan kamera <i>visible</i> dengan pencahayaan	45
Gambar 4.6	Gambar sample uji dengan kamera <i>visible</i> tanpa pencahayaan	45
Gambar 4.7	Gambar sample uji dengan kamera CCTV inframerah	45
Gambar 4.7	Grafik persentase tingkat keakurasian pengujian	48

DAFTAR TABEL

Tabel 4.1	Hasil pengujian dengan alpha 0.2	41
Tabel 4.2	Hasil pengujian dengan alpha 0.6	42
Tabel 4.3	Hasil pengujian dengan alpha 0.9	43
Tabel 4.4	Hasil pengujian sample kamera visible dengan pencahayaan	46
Tabel 4.5	Hasil pengujian sample kamera visible tanpa pencahayaan	46
Tabel 4.6	Hasil Pengujian sample kamera CCTV inframerah	46

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dengan perkembangan teknologi sekarang ini, sudah banyak jenis-jenis sistem pendeteksi yang dikembangkan untuk kepentingan pengguna teknologi itu sendiri. Sistem pendeteksi ini bisa digunakan pada sebuah benda maupun anggota tubuh manusia. Teknologi yang menggunakan tubuh manusia sebagai objek deteksi adalah teknologi biometrik. Dimana teknologi biometrik inilah yang digunakan sebagai dasar dari sistem pendeteksi dari anggota tubuh manusia. Dalam dunia medis sudah dikenal kalau beberapa dari anggota tubuh manusia memiliki perbedaan dengan manusia yang lainnya. Seperti perbedaan pada kontur sidik jari, retina mata dan wajah manusia. Selain itu anggota tubuh manusia juga dapat menghasilkan ekspresi yang berbeda seperti senyum, sedih, marah dan lain-lain.

Salah satu metode yang digunakan dalam proses pendeteksian adalah metode *Backpropagation* atau Komputasi Balik yang lebih dikenal merupakan bagian dari Jaringan Saraf Tiruan atau *Neural Network*. Metode ini bisa dipergunakan untuk melakukan pendeteksian pada wajah yang sangat berguna untuk sistem keamanan berbasis komputer. Sehingga proses pengenalan terhadap seseorang dapat dilakukan dengan cepat.

Dan metode ini dapat diaplikasikan pada sistem keamanan berbasis komputer dengan sistem akses. Dalam pengoperasian sistem keamanan yang beroperasi selama 24 jam terdapat masalah pada penggunaan media kamera yang digunakan, dimana kamera visible hanya mampu mengambil gambar dengan pencahayaan yang cukup. Untuk itu perlu dipergunakannya kamera yang mampu beroperasi pada malam hari maupun siang hari, sehingga sistem keamanan dapat bekerja selama 24 jam penuh.

1.2 Tujuan Penulisan

Tujuan penulisan Skripsi ini adalah untuk perancangan pendeteksian wajah yang digunakan untuk sistem keamanan berbasis komputer yang dapat dilakukan baik pada siang hari maupun malam hari dan mengetahui tingkat keberhasilan pengujian dengan menggunakan *metode Backpropagation* Jaringan Saraf Tiruan. Selain akan dibandingkan hasil penggunaan kamera CCTV inframerah dengan kamera *visible* pada kondisi dengan pencahayaan dan tanpa pencahayaan.

1.3 Batasan Masalah

Informasi yang diberikan pada tulisan Skripsi ini adalah perangkat dan sistem yang digunakan dalam penelitian. Kamera CCTV inframerah yang digunakan untuk mengambil sample data yang digunakan sebagai data base untuk proses training. Pengolahan data base menggunakan sistem Jaringan Saraf Tiruan.

1.4 Metodologi

Metodologi yang digunakan dalam penulisan Skripsi ini adalah dengan studi literature. Proses yang dilakukan adalah sebagaiberikut:

- Merumuskan permasalahan
- Melakukan pengumpulan informasi mengenai Jaringan Saraf Tiruan
- Merencanakan dan merancang sistem dengan melengkapi dengan perangkat yang dibutuhkan
- Perancangan aplikasi pendeteksian wajah dengan kamera CCTV inframerah
- Pengujian rancangan sistem
- Evaluasi dan perbaikan

1.5 Sistematika Penulisan

Sistematika penulisan laporan Skripsi ini adalah sebagai berikut:

Bab 1 Pendahuluan, berisi penjelasan mengenai latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, serta sistematika penulisan yang digunakan untuk menyusun laporan Tugas Akhir.

Bab 2 Dasar Teori, berisi dasar teori yang digunakan dalam analisis dan perancangan pada skripsi yang dipresentasikan. Dasar teori yang diangkat antara lain teknologi biometri, citra digital, metode pencocokan wajah template, metode deteksi dengan jaringan syaraf tiruan, dan Sistem Autentikasi.

Bab 3 Perancangan, berisi rancang bangun system pengenalan wajah menggunakan metode jaringan syaraf tiruan dan pencocokan pola wajah, mencakup analisis metode serta proses pembelajaran jaringan syaraf tiruan, analisis metode pencocokan wajah, representasi wajah penggabungan kedua metode, untuk kemudian di aplikasikan kedalam simulasi system autentikasi.

Bab 4 Hasil Dan Analisa, hasil dari percobaan dan analisa.

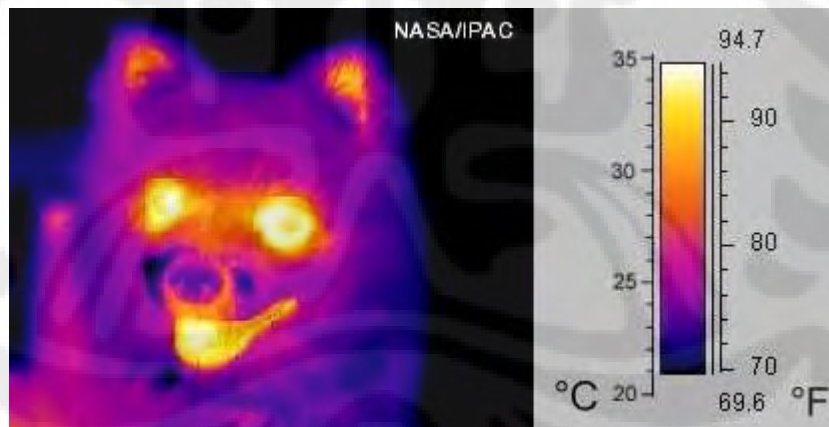
Bab 5 Kesimpulan berisi kesimpulan dari hasil.

BAB 2 DASAR TEORI

2.1. Kamera Inframerah

2.1.1. Inframerah [5]

Inframerah (Infrared) adalah radiasi elektromagnetik dari panjang gelombang lebih panjang dari cahaya tampak, tetapi lebih pendek dari radiasi gelombang radio. Nama *infrared* secara bahasa diartikan "bawah merah" yang dalam bahasa Latin *infra* berarti "bawah", dan *red* atau warna merah dalam bahasa Inggris merupakan warna dari cahaya tampak dengan gelombang terpanjang. Radiasi inframerah memiliki tiga kategori jangkauan, Inframerah dengan jarak dekat dengan panjang gelombang 0,75 – 1,5 μm , Inframerah dengan jarak menengah dengan panjang gelombang 1,50 – 10 μm dan Inframerah dengan jarak jauh dengan panjang gelombang 10 – 100 μm . Inframerah ditemukan secara tidak sengaja oleh Sir William Herschell, astronom kerajaan Inggris ketika ia sedang mengadakan penelitian mencari bahan penyaring optik yang akan digunakan untuk mengurangi kecerahan gambar matahari dalam tata surya teleskop.



Gambar 2.1 Seekor anjing yang diambil dalam cahaya inframerah

Dari gambar 2.1 dapat dilihat hasil gambar yang diambil dalam cahaya inframerah, dan dapat terlihat dengan jelas setiap anggota tubuh dari anjing tersebut memancarkan warna yang berbeda. Perbedaan warna tersebut mewakili suhu yang dipancarkan oleh bagian tubuh dari anjing tersebut, semakin tinggi suhu yang dipancarkan maka warna yang ditampilkan akan semakin terang. Dan

warna gelap yang di tampilkan pada gambar mewakili suhu yang rendah dari objek yang ditangkap.

2.1.2. *Night Vision* [7]

Penglihatan malam atau *night vision* adalah kemampuan untuk melihat baik dalam arti dengan kemampuan biologis atau teknologi dalam lingkungan gelap. Kemampuan *night vision* dapat dicapai dengan menggunakan dua pendekatan yaitu meningkatkan batas spektrum gelombang yang dapat dilihat atau meningkatkan kemampuan untuk melihat intensitas cahaya yang kurang. Pada pendekatan pertama, kemampuan penglihatan mata manusia dibatasi hanya pada batasan tertentu dalam gelombang elektromagnetik yang disebut cahaya tampak. Dengan meningkatkan batas spektrum gelombang cahaya yang dapat dilihat, pengamat dapat melihat sumber-sumber cahaya tidak tampak seperti gelombang inframerah atau ultraungu. Pendekatan yang kedua, pengamat diberi kemampuan untuk melihat cahaya walaupun dalam intensitas kecil. Hal ini dapat dicapai dengan menggunakan teknologi *image intensifier*.



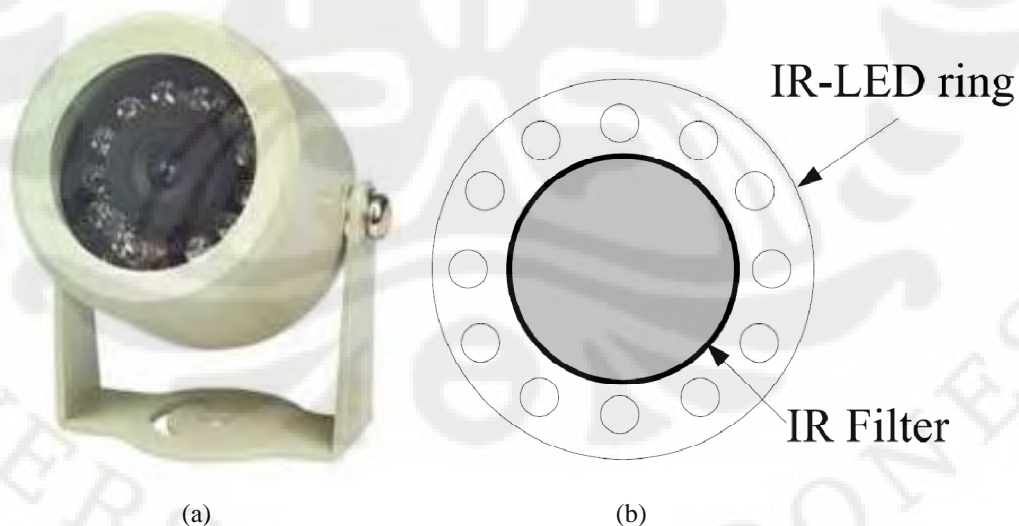
Gambar 2.2 Gambar yang diambil dengan teropong dengan kemampuan *night vision*

Hasil dari peralatan teknologi yang menggunakan kemampuan *night vision* dapat di lihat pada gambar 2.2, pada gambar tersebut ditampilkan hasil tangkapan dari teropong yang menggunakan kemampuan *night vision* yang dipergunakan untuk kepentingan militer dalam peperangan. Dalam dunia militer peralatan teknologi dengan kemampuan *night vision* tentunya sangat diperlukan untuk mengetahui kondisi medan perang pada saat malam hari. Pada malam hari tentunya kita perlu bantuan cahaya untuk melihat, tapi dalam medan perang

penggunaan cahaya pada malamhari akan memudahkan musuh untuk mengetahui posisi kita. Oleh karena itu penggunaan peralatan teknologi dalam bidang militer seperti teropong dengan kemampuan *night vision* sangat diperlukan untuk membantu penglihatan dimedan perang.

2.1.3. Kamera CCTV Inframerah

Kamera CCTV infrared adalah jenis kamera CCTV yang digunakan untuk sistem keamanan yang menggunakan *Infrared LED* dan *Infrared filter* dalam mengambil gambar. Sinar inframerah memang tidak dapat ditangkap oleh mata telanjang manusia, namun sinar inframerah tersebut dapat ditangkap oleh kamera digital atau video handycam. Dengan adanya suatu teknologi yang berupa filter iR PF [4] yang berfungsi sebagai penerus cahaya inframerah, maka kemampuan kamera atau video tersebut menjadi meningkat, seperti pengambilan gambar dalam kondisi gelap. Penggunaan kamera jenis ini dikarenakan keunggulannya dalam menangkap gambar dalam penglihatan malam (*night vision*). Pada saat malam hari kamera biasa (*visible camera*) tidak dapat menangkap gambar dengan baik karena kamera biasa memanfaatkan pencahayaan untuk mengambil gambar. Sedangkan kamera CCTV inframerah memanfaatkan cahaya inframerah dan filter inframerah untuk menangkap gambar.



Gambar 2.3 Bentuk kamera yang digunakan Kamera CCTV IR Outdoor JMK JK-212

Pada gambar 2.3 (a) ditampilkan bentuk dari perangkat kamera CCTV inframerah yang akan dipergunakan untuk kepentingan penelitian dalam penulisan skripsi ini. Jenis kamera yang digunakan adalah kamera CCTV IR *Outdoor* JMK

JK-212, dimana kamera CCTV tersebut terdiri dari 12 LED (*Light Emiting Diode*) inframerah dan IR Filter yang bentuknya dapat dilihat pada gambar 2.3 (b). LED inframerah ini berfungsi untuk memancarkan cahaya inframerah jika kamera CCTV kekurangan cahaya. LED inframerah yang berjumlah 12 tersebut tersusun mengelilingi kamera dan membentuk *ring* atau cincin dengan jarak yang sama. Tujuannya untuk menyeimbangkan cahaya inframerah yang dipancarkan dari masing-masing LED inframerah, sehingga ketika kamera mengambil gambar cahaya inframerah yang ditangkap sama. Sedangkan fungsi dari IR Filter atau filter inframerah adalah untuk menyaring cahaya tampak disekitar kamera CCTV dan meneruskan cahaya inframerah yang dipancarkan oleh LED inframerah. Sehingga hasil tangkapan dengan kamera CCTV inframerah tidak terdapat warna atau cahaya tampak. Untuk hasil tangkapan kamera CCTV IR Outdoor JMK JK-212 dapat dilihat pada gambar 2.4.



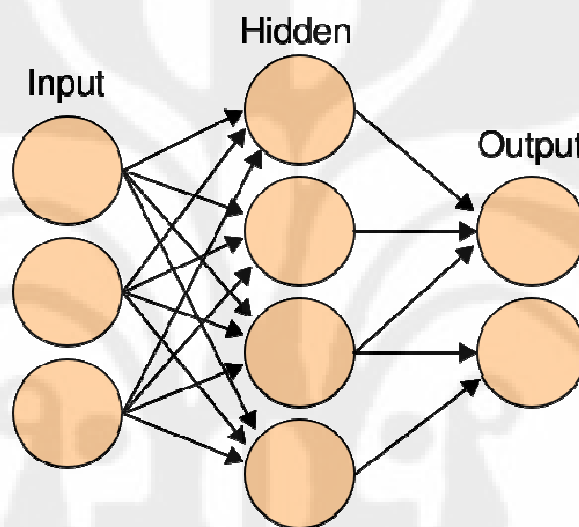
Gambar 2.4 Hasil gambar menggunakan Kamera CCTV IR *Outdoor* JMK JK-212

Dari hasil tangkapan kamera CCTV inframerah gambar yang dihasilkan lebih buram dari hasil tangkapan kamera *visible* karena cahaya tampak pada kamera CCTV inframerah sudah terfilter oleh IR Filter. Dengan terfilternya cahaya tampak ini proses pengolahan gambar akan lebih mudah karena pada gambar yang diolah tidak terdapat cahaya tampak yang terdapat disekitar objek gambar yang akan diproses.

2.2. Jaringan Syaraf Tiruan

Jaringan saraf tiruan (JST) atau umumnya hanya disebut neural network (NN), adalah sistem pembelajaran berbentuk jaringan yang terinspirasi

berdasarkan jaringan saraf manusia. Jaringan Saraf Tiruan merupakan sistem adaptasi yang dapat merubah strukturnya untuk memecahkan masalah berdasarkan informasi yang masuk dan keluar melalui jaringan tersebut. Bisa dikatakan, Jaringan Saraf Tiruan adalah sebuah sistem pemodelan data statistik non-linier. Jaringan Saraf Tiruan dapat digunakan untuk memodelkan hubungan yang kompleks antara input, hidden dan output yang berupa layer-layer untuk menemukan pola-pola pada data. Untuk pola dari Jaringan Saraf Tiruan dapat dilihat pada gambar 2.5 yang terdiri dari input layer, hidden layer dan output layer.



Gambar 2.5 Gambar Pola Neural Network

2.2.1. Sejarah Jaringan Syaraf Tiruan[2]

Jaringan syaraf tiruan dibuat pertama kali pada tahun 1943 oleh *neurophysiologist* Waren McCulloch dan *logician* Walter Pitts, namun teknologi yang tersedia pada saat itu belum memungkinkan mereka berbuat lebih jauh. Jaringan syaraf tiruan diartikan sebagai susunan dari proses penghitung yang disebut neuron yang saling terhubung sehingga bisa meniru fungsi otak manusia. Jaringan syaraf tiruan bisa diketahui melalui proses pembelajaran (*learning*) yang berfungsi untuk mengadaptasi parameter-parameternya. Bobot dalam jaringan yang diusulkan oleh McCulloch dan Pitts diatur untuk melakukan fungsi logika sederhana. Fungsi aktivasi yang dipakai adalah fungsi *threshold*.

Tahun 1958, Rosenblatt memperkenalkan dan mulai mengembangkan model jaringan yang disebut *perceptron*. Metode pelatihan diperkenalkan untuk

mengoptimalkan hasil iterasinya. Widrow dan Hoff (1960) mengembangkan *perceptron* dengan memperkenalkan aturan pelatihan jaringan, yang dikenal sebagai aturan delta (atau sering disebut kuadrat rata-rata terkecil). Aturan ini akan mengubah bobot *perceptron* apabila keluaran yang dihasilkan tidak sesuai dengan target yang diinginkan. Apa yang dilakukan peneliti terdahulu hanya menggunakan jaringan dengan layer tunggal (*single layer*). Rumelhart (1986) mengembangkan *perceptron* menjadi *Backpropagation*, yang memungkinkan jaringan diproses melalui beberapa layer. Selain itu, beberapa model jaringan syaraf tiruan lain juga dikembangkan oleh Kohonen (1972), Hopfield (1982), dan lain-lain. Pengembangan yang ramai dibicarakan sejak tahun 1990an adalah aplikasi model-model jaringan syaraf tiruan untuk menyelesaikan berbagai masalah di dunia nyata. Jaringan Syaraf Tiruan ditentukan oleh beberapa hal berikut hal :

1. Pola hubungan antar neuron atau arsitektur jaringan
2. Metode untuk menentukan bobot penghubung atau metode *learning*
3. Penggunaan Bias
4. Fungsi aktivasi

Jaringan Syaraf tiruan dibentuk sebagai generalisasi model matematika dari syaraf biologi, dengan asumsi bahwa :

1. Pemrosesan informasi terjadi pada elemen sederhana yang di namakan neuron.
2. Sinyal antar neuron berhubungan melalui saluran penghubung.
3. Setiap saluran penghubung mempunyai nilai bobot, dan melakukan operasi perkalian dengan sinyal yang ditranmisikan.
4. Setiap neuron memberlakukan fungsi aktivasi (biasanyatidak linier) pada masukan total.

Karakteristik dari jaringan syaraf buatan :

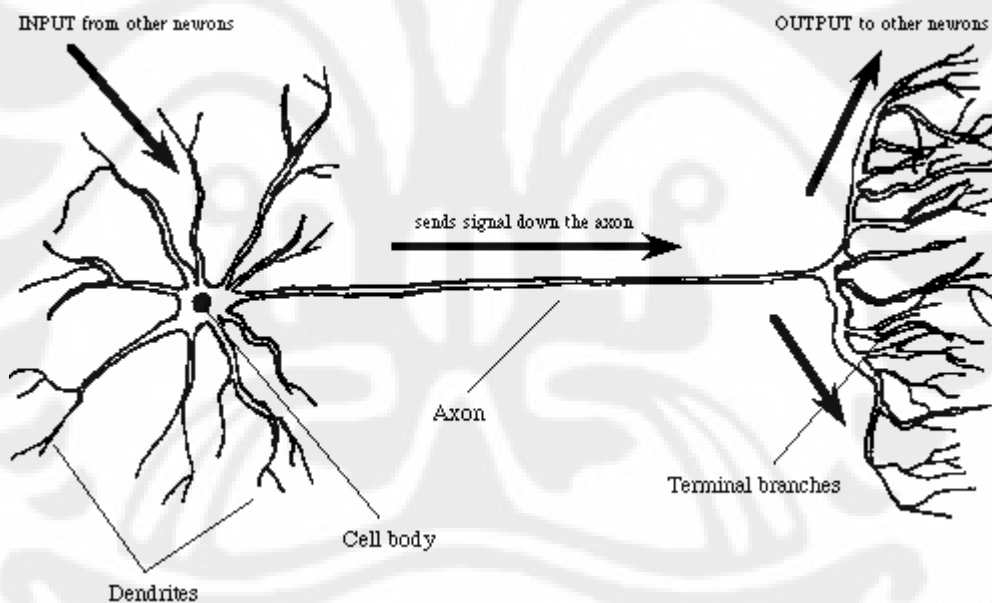
1. Arsitektur Jaringan : pola keterhubungan antar neuron.
2. Algoritma Jaringan (pelatihan, pembelajaran) metode menentukan nilai bobot hubungan
3. Fungsi Aktivasi : menentukan nilai keluaran berdasarkan nilai total masukannya.

Jaringan saraf tiruan dapat digunakan untuk :

1. Menyimpan dan memanggil data (pola)
2. Mengklasifikasi data-data
3. Melakukan pemetaan dari data-data masukan kedalam data-data keluaran
4. Pemetaan berdasarkan data-data sejenis
5. Menyelesaikan persoalan optimasi terbatas

2.2.2. Neuron

Neuron atau sel syaraf pada manusia yang berfungsi sebagai pengirim pesan yang berupa rangsangan untuk dikenali. Dalam Jaringan Saraf Tiruan neuron diartikan sebagai pembawa data untuk proses pengolahan[2]. Jadi neuron disebut juga dapat dinyatakan sebagai proses sederhana dari sistem Jaringan Saraf Tiruan.



Gambar 2.6 Jaringan Saraf Neuron

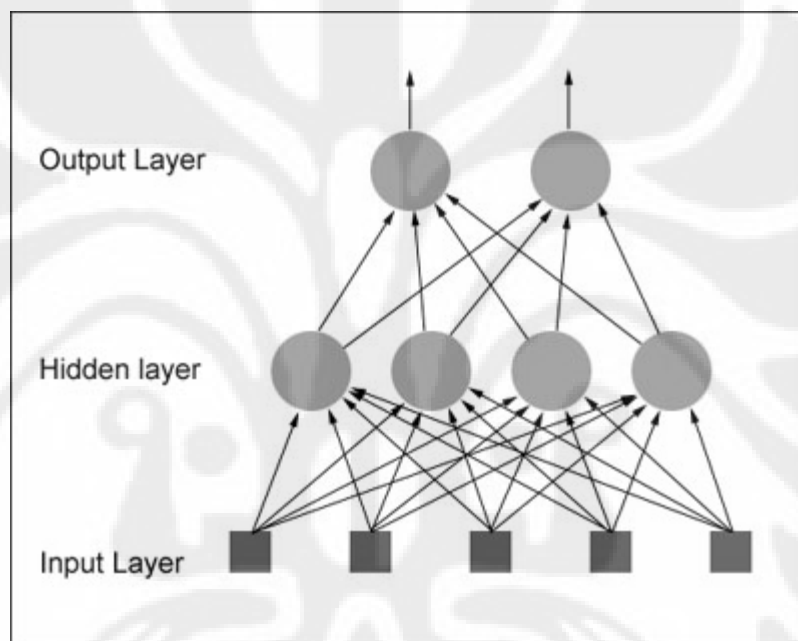
Gambar 2.6 merupakan gambar dari Jaringan Saraf Neuron pada manusia, dimana Jaringan Saraf Neuron saling berhubungan satu sama lain untuk mengirimkan sinyal ke otak manusia. Pada Jaringan Saraf Neuron, *Dendrites* berfungsi sebagai penerima input dari Neuron yang lain, kemudian input akan dikirimkan melalui *Axon* dan dikeluarkan melalui *Terminal Branches* ke Neuron yang lain. Sistem inilah yang menjadi inspirasi dari sistem Jaringan Saraf Tiruan dalam pengolahan data.

2.2.3. Arsitektur Jaringan

Berdasarkan dari arsitektur, Jaringan Syaraf Tiruan dapat dibagi kedalam dua kategori :

a) Struktur Komputasi Maju (*FeedForward*)

Feedforward adalah tahapan awal dari proses jaringan syaraf buatan, signal akan masuk dimulai dari data input dan kemudian akan melewati lapisan tersembunyi (hidden layer), dari lapisan tersembunyi akan diteruskan menuju layer output dan dari output layer akan didapat hasil keluaran. Untuk pola dari struktur *Feedforward* dapat dilihat pada gambar 2.7.



Gambar 2.7 Struktur *FeedForward*

Yang termasuk dalam struktur *feedforward* :

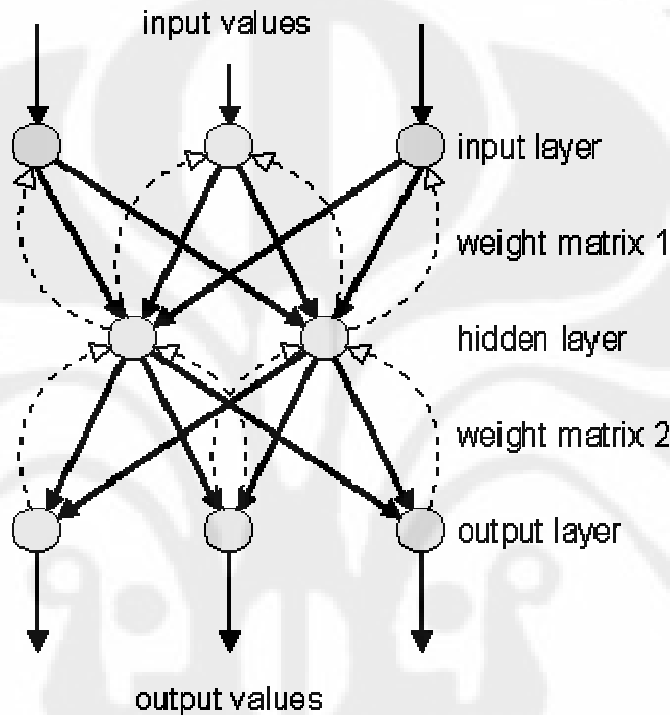
- Single-layer Network
- Multilayer Network
- Radial-basis function networks
- Higher-order networks
- Polynomial learning networks

b) Struktur komputasi balik (*Backpropagation*)

Backpropagation merupakan proses feedback dari proses feedforward, dimana output dari feedforward akan dijadikan input kembali untuk melakukan proses feedforward berikutnya sehingga menjadi jaringan yang berulang. Dengan sistem

seperti ini akan maka jaringan akan melakukan pembelajaran terhadap hasil yang didapat sebelumnya. Yang termasuk dalam struktur *backpropagation* :

- Competitive networks
- Self-organizing maps
- Hopfield networks
- Adaptive-resonance theory models

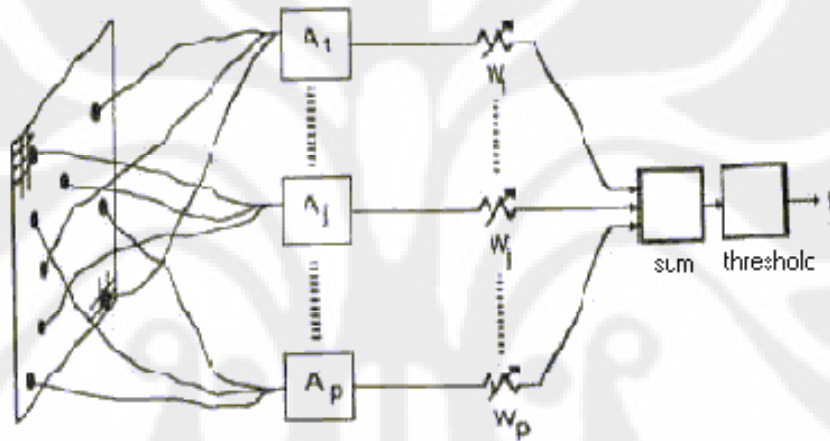


Gambar 2.8 Struktur *Backpropagation*

Gambar 2.8 merupakan struktur Backpropagation dari Jaringan Saraf Tiruan, ketika sistem tersebut diberi input dari nilai suatu variable maka input akan ditempatkan dalam input layer dan kemudian akan diteruskan menuju hidden layer dan selanjutnya menuju output layer sehingga didapatlah nilai output. Setiap layer tersebut menghitung nilai aktivasi dengan mengambil jumlah bobot (weight) output dari setiap unit dari layer sebelumnya dan kemudian dikurangi dengan nilai ambang. Aktivasi pada input layer menunjukkan informasi dasar yang selanjutnya digunakan sebagai inputan pada sistem. Aktivasi pada hidden layer bergantung pada input layer dan nilai bobot dari hubungan antara input layer dan hidden layer (weight matrik 1). Sedangkan karakteristik pada output layer tergantung pada aktivasi di hidden layer dan nilai bobot dari hubungan antara hidden layer dan output layer (weight matrik 2).

2.2.4. Perceptron [2]

Perceptron termasuk kedalam salah satu bentuk Jaringan Syaraf Tiruan yang sederhana dan merupakan pengembangan jaringan Hebb. Perceptron biasanya digunakan untuk mengklasifikasikan suatu tipe pola tertentu yang sering dikenal dengan istilah pemisahan secara linear. Pada dasarnya perceptron pada Jaringan Syaraf dengan satu lapisan memiliki bobot yang bisa diatur dan suatu nilai ambang. Algoritma yang digunakan oleh aturan perceptron ini akan mengatur parameter-parameter bebasnya melalui proses pembelajaran. Fungsi aktivasi dibuat sedemikian rupa sehingga terjadi pembatasan antara daerah positif dan daerah negatif. Untuk bentuk dari Perceptron dapat dilihat pada gambar 2.9



Gambar 2.9 Bentuk Perceptron

2.2.5. Pembelajaran

Berdasarkan cara pembelajaran bobotnya, ada 2 macam pelatihan dari jaringan syaraf tiruan yang dikenal yaitu dengan supervisi (*supervised*) atau lebih dikenal dengan SOM (*Self Organizing Map*) dan tanpa *supervise* (*unsupervised*) atau yang lebih dikenal dengan LVQ (*Learning Vektor Quantization*). Dalam pelatihan dengan SOM, terdapat sejumlah pengelompokan data (input – target output) yang dipakai untuk melatih jaringan hingga diperoleh bobot yang diinginkan. Pasangan data tersebut berfungsi sebagai patokan dasar untuk melatih jaringan hingga diperoleh bentuk yang terbaik. Patokan dasar akan memberikan informasi yang jelas tentang bagaimana sistem harus mengubah dirinya untuk meningkatkan kinerjanya. Pada setiap kali pelatihan, suatu input diberikan ke jaringan. Jaringan akan memproses dan mengeluarkan keluaran. Selisih antara

keluaran jaringan dengan target (keluaran yang diinginkan) merupakan kesalahan yang terjadi. Jaringan akan memodifikasi bobot sesuai dengan kesalahan tersebut. *backpropagation* merupakan model yang menggunakan pelatihan dengan supervisi.

Sedangkan pada LVQ tidak terjadi proses pembelajaran (*unsupervised*). Dalam pelatihannya, setiap unit output mewakili satu kelas tertentu dan beberapa unit output harus digunakan pada setiap kelas. Vector bobot dari unit output ini lebih dikenal sebagai vector acuan bagi tiap kelas yang diwakili oleh keluaran tersebut.

2.2.6. Fungsi Aktivasi

Jaringan syaraf adalah merupakan salah satu representasi buatan dari struktur biologis 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. 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.

Jaringan Saraf Buatan buatan ini sebenarnya meniru dari prinsip kerja sel neuron biologis. Neuron-neuron buatan tersebut bekerja dengan cara yang sama pula dengan neuron-neuron biologis. Informasi atau 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 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 outputnya ke semua neuron yang berhubungan dengannya. Pada jaringan syaraf,

neuron-neuron akan dikumpulkan dalam lapisan-lapisan 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, bias jadi informasi tersebut akan dirambatkan secara mundur pada jaringan. Ada beberapa fungsi aktivasi yang sering digunakan dalam jaringan syaraf tiruan. Fungsi Aktivasi yang disediakan pada toolbox Matlab, antara lain [3] :

1. Fungsi Identitas

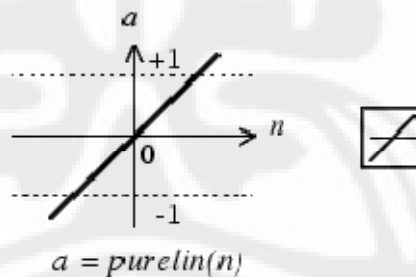
Fungsi Identitas memiliki nilai *output* yang sama dengan nilai inputnya. Fungsi Identitas dirumuskan sesuai dengan rumus 2.1 dan hasil grafik dapat dilihat pada gambar 2.10:

$$y = x \quad (2.1)$$

Keterangan:

x = input

y = output



Gambar 2.10 Linear Transfer Function

2. Fungsi Tangga-Biner

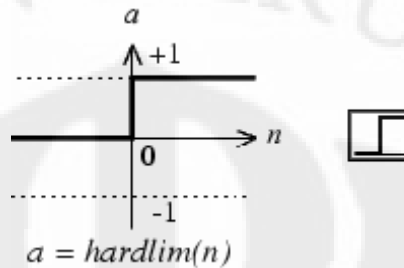
Jaringan dengan lapisan tunggal sering menggunakan fungsi Tangga-Biner (*step function*) untuk mengkonversikan *input* dari suatu variabel yang bernilai kontinu ke suatu *output* biner (0 atau 1). Fungsi undak biner (*hardlimit*) dirumuskan sesuai dengan rumus 2.2 dan hasil grafik dapat dilihat pada gambar 2.11:

$$y = \begin{cases} 0, & \text{Jika } x < 0 \\ 1, & \text{Jika } x \geq 0 \end{cases} \quad (2.2)$$

Keterangan:

x = input

y = output



Gambar 2.11 Fungsi Tangga-Biner

3. Fungsi Sigmoid (Logistik)-Biner

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 *outputnya* 0 atau Fungsi sigmoid biner dirumuskan sesuai dengan rumus 2.3 dan 2.4 dan hasil grafik dapat dilihat pada gambar 2.12:

$$y = f(x) = \frac{1}{1+e^{-\sigma x}} \quad (2.3)$$

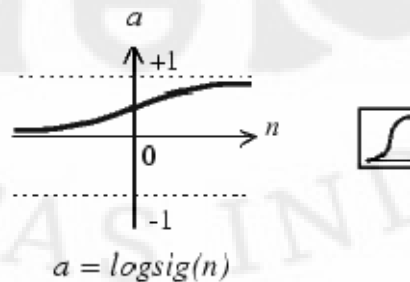
$$\text{Dengan } f'(x) = \sigma f(x)[1 - f(x)] \quad (2.4)$$

Keterangan:

y = output

$f(x)$ = fungsi output

$f'(x)$ = turunan dari $f(x)$



Gambar 2.12 Fungsi Sigmoid (Logistik)-Biner

4. Fungsi Sigmoid-Bipolar

Fungsi sigmoid bipolar hampir sama dengan fungsi sigmoid biner, hanya saja *output* dari fungsi ini memiliki range antara 1 sampai -1 seperti terlihat pada Gambar 2.13. Fungsi sigmoid bipolar dirumuskan sesuai dengan rumus 2.5 dan 2.6 dan hasil grafik dapat dilihat pada gambar 2.13:

$$y = f(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (2.5)$$

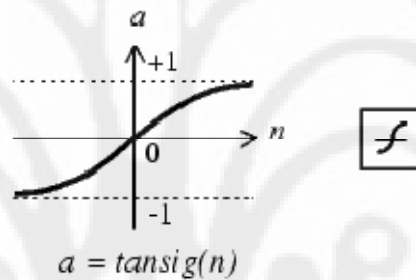
$$\text{Dengan } f'(x) = \frac{\sigma}{2} [1 + f(x)][1 - f(x)] \quad (2.6)$$

Keterangan:

y = output

$f(x)$ = fungsi output

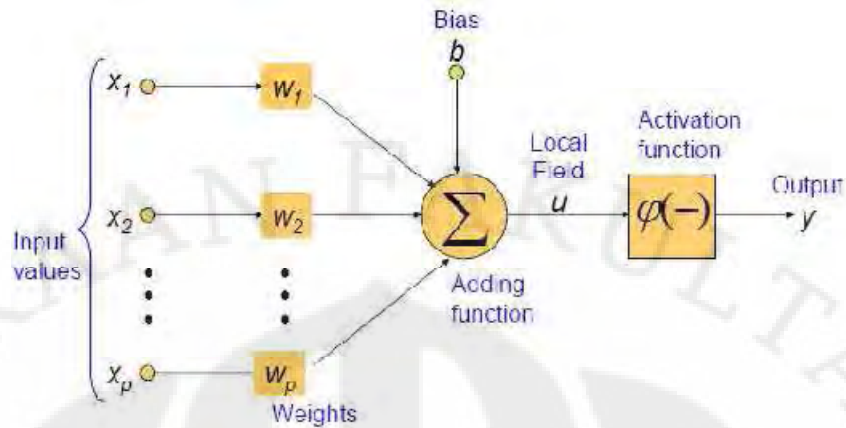
$f'(x)$ = turunan dari $f(x)$



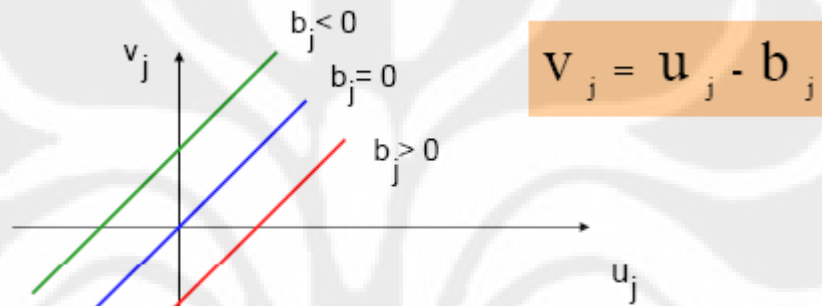
Gambar 2.13 Fungsi Sigmoid-Bipolar

2.2.7. Bias

Di antara layer yang satu dengan layer yang berikutnya pada jaringan Saraf tiruan dihubungkan dengan model koneksi yang memiliki bobot-bobot (*weights*), w adalah nilai bobot untuk layer output dan v adalah nilai bobot untuk lapisan tersembunyi. Lapis tersembunyi dapat memiliki bias yang memiliki nilai bobot yang sama dengan satu sama lain. Pada Jaringan Saraf Buatan komponen seluruh inputan (x) dan komponen weight (w) dengan fungsi tambahan dengan nilai bias dan fungsi aktifasinya berperan untuk membatasi amplitudo dari output yang dihasilkan. Jaringan Syaraf Tiruan dengan bias dapat dilihat sesuai dengan Gambar 2.14:

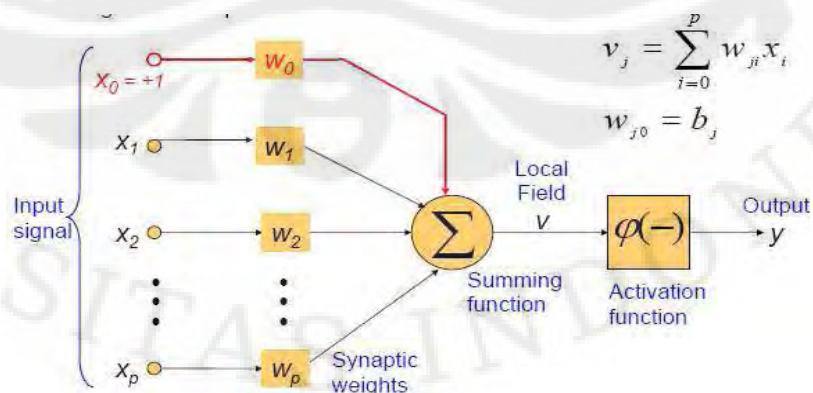


Gambar 2.14 Jaringan Saraf Tiruan dengan Bias



Gambar 2.15 Penggambaran Bias dengan Linearitas

Pada gambar 2.15 dapat dilihat perbandingan antara nilai bias dengan linearitas. Dimana nilai bias (b_j) berpengaruh dalam penerapan nilai transformasi pada jumlah bobot u_j . Bias merupakan parameter eksternal dari Jaringan Saraf tiruan, v_j disini merupakan sebagai *induced field* dari Jaringan Saraf buatan. Hal ini dapat dimodelkan dengan penambahan *extra input* dengan nilai bias (w_0) yang didapat dari faktor inputan seperti pada Gambar 2.16.



Gambar 2.16 Penggambaran Bias sebagai Input

2.2.8. Algoritma *Backpropagation*

Algoritma Pelatihan Jaringan Saraf Tiruan *Backpropagation* adalah [3]:

- Inisialisasi bobot (ambil awal dengan nilai acak yang kecil)
- Tetapkan : Maksimum Epoch, Target *error*, dan *learning rate* (α)
- Inisialisasi : Epoch = 0, MSE = 1.
- Kerjakan langkah-langkah berikut selama (Epoch < Maksimum Epoch) dan (MSE > Target *Error*) :

- Epoch = Epoch + 1
- Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan :

Feedforward :

- Tiap-tiap unit *input* ($x_i = 1, 2, 3, \dots, n$) menerima sinyal input dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi).
- Tiap-tiap unit pada lapisan tersembunyi ($Z_j, j=1, 2, 3, \dots, p$) menjumlahkan sinyal-sinyal *input* berbobot :

$$z_{in_j} = b_1 + \sum_{i=1}^n x_i v_{ij} \quad (2.7)$$

Gunakan fungsi aktivasi untuk menghitung sinyal outputnya :

$$z_j = f(z_{in_j}) \quad (2.8)$$

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

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

$$y_{in_k} = b_2 + \sum_{j=1}^p z_j w_{jk} \quad (2.9)$$

Gunakan fungsi aktivasi untuk menghitung sinyal outputnya :

$$y_k = f(y_{in_k}) \quad (2.10)$$

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

Langkah (b) dilakukan sebanyak jumlah lapisan tersembunyi.

Backpropagation :

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

$$\delta 2_k = (t_k - y_k) f'(y_{in_k}) \quad (2.11)$$

$$\varphi 2_{jk} = \delta_k z_j \quad (2.12)$$

$$\beta 2_k = \delta_k \quad (2.13)$$

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

$$\Delta w_{jk} = \alpha \varphi 2_{jk} \quad (2.14)$$

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

$$\Delta b_{2_k} = \alpha \beta 2_k \quad (2.15)$$

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 inputnya (dan unit-unit yang berada pada lapisan yang ada di atasnya):

$$\delta_{in_j} = \sum_{k=1}^m \delta 2_k w_{jk} \quad (2.16)$$

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

$$\delta 1_j = \delta_{in_j} f'(z_{in_j}) \quad (2.17)$$

$$\varphi 1_{ij} = \delta 1_j x_j \quad (2.18)$$

$$\beta 1_j = \delta 1_j \quad (2.19)$$

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

$$\Delta v_{ij} = \alpha \varphi 1_{ij} \quad (2.20)$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai b_1)

$$\Delta b_1 = \alpha \beta_1 \quad (2.21)$$

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

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

$$b_{2k}(\text{baru}) = b_{2k}(\text{lama}) + \Delta b_{2k} \quad (2.23)$$

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

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

$$b_{1j}(\text{baru}) = b_{1j}(\text{lama}) + \Delta b_{1j} \quad (2.25)$$

➤ Hitung MSE (*Mean Square Error*)

Setelah dilakukan proses pada algoritma tersebut maka kita akan mendapatkan Jaringan Saraf Buatan yang sudah ditraining. Sehingga untuk melakukan *indentifikasi*, dapat dilakukan dengan langsung memberikan input dan jaringan akan mengklasifikasinya sesuai dengan bobot-bobot yang diperoleh dari proses *training* sebelumnya. Perhitungan MSE ini merupakan pengukuran yang memungkinkan Jaringan Saraf Tiruan dapat melakukan proses belajar dengan baik. Perhitungan MSE ini merupakan pengukuran ketepatan jaringan saraf tiruan terhadap data target pembelajaran. MSE pada keluaran Jaringan Saraf Tiruan merupakan selisih antara keluaran yang didapat dari proses pembelajaran dengan target yang diinginkan dari masukan data tertentu. Pada proses pembelajaran, data yang menjadi pembanding adalah data pembelajaran, sedangkan pada proses pengujian, data yang dipakai adalah data uji.

Jumlah MSE pada jaringan saraf tiruan dapat dihitung dengan menggunakan rata-rata MSE kuadrat. Berikut ini adalah cara menghitung MSE untuk kumpulan data d , pada jaringan saraf tiruan dengan kumpulan neuron keluaran *outputs*:

$$MSE = \frac{\sum_d \sum_{k \in \text{outputs}} (t_k - o_k)^2}{n_d n_{\text{outputs}}} \quad (2.26)$$

Dengan:

t_k adalah nilai target pada neuron keluaran ke- k

o_k adalah nilai output pada neuron keluaran ke- k

n_d adalah jumlah data pembelajaran, dan

n_{output} adalah jumlah neuron keluaran

BAB 3

Pengenalan Wajah dengan Metode Backpropagation Menggunakan Kamera CCTV Inframerah

Dalam penulisan Skripsi ini, penulis mengajukan rancangan sistem pengenalan wajah dengan metode *backpropagation* (komputasi balik). Sistem ini dapat digunakan untuk aplikasi keamanan dalam suatu ruangan yang sudah dilengkapi dengan kamera CCTV inframerah. Kamera CCTV inframerah tersebut dapat digunakan untuk menangkap gambar wajah seseorang dalam baik pada waktu siang hari maupun malam hari yang melakukan akses kedalam suatu ruangan. Setiap wajah yang tertangkap dapat diidentifikasi oleh sistem, sehingga dapat diketahui siapa saja yang melakukan akses kedalam ruangan tersebut.

Untuk bisa melakukan indentifikasi wajah seseorang yang tertangkap oleh kamera CCTV inframerah, diperlukan suatu metode pengenalan yang dapat diterapkan dengan mudah dan efektif. Dalam hal ini penulis memilih metode *Backpropagation* Jaringan syaraf tiruan.

3.1. Data input

Data yang diambil untuk proses pengenalan adalah gambar (*image*) dari wajah orang-orang yang dijadikan contoh (*sample*) pengenalan. Proses pengambilan data dilakukan dengan perangkat (*interface*) yang dipergunakan dalam sistem.

Adapun perangkat yang digunakan dalam melakukan perancangan ini adalah:

- Kamera CCTV IR Outdoor JMK JK-212
- Gadmei external USB TV BOX UTV332
- 1 unit PC
 - Prosesor Intel Core 2 duo TE4300 @ 1.80 GHz
 - Memory (RAM) 1 GB
 - Operating System Windows 7
 - Matlab 7.8.0
 - GOM Player
 - ACDSsee 10 Photo Manager

Sample wajah terlebih dahulu direkam dalam bentuk video dengan format mpg yang direkam menggunakan bantuan TV Tuner external Gadmei external USB TV BOX UTV332. Hasil rekaman yang berdurasi 40-45 detik untuk tiap sample. Dari rekaman video tersebut diambil potongan-potongan screen shoot video dalam bentuk gambar dalam format jpg dengan menggunakan GOM Player, sehingga bisa didapat 150-200 gambar contoh wajah dengan jumlah sample yang diambil adalah 5 orang (kelas). Dari semua gambar dipilih gambar yang terbaik hasil tangkapannya (capture), kemudian di crop pada bagian wajah. Hasil crop kemudian di ubah ukuran resolusinya menjadi 30 x 30. Untuk proses crop dan resize menggunakan program ACDSsee 10 Photo Manager. Semua proses ini dilakukan secara manual sehingga memerlukan waktu yang lama untuk proses pembentukan database. Untuk proses pengambilan sample wajah dapat dilihat pada gambar 3.1 dan hasil crop dari sample image pada posisi wajah dapat dilihat pada gambar 3.2, sedangkan hasil resize yang digunakan sebagai database dapat dilihat pada gambar 3.3



Gambar 3.1 Proses pengambilan sample wajah



Gambar 3.2 Contoh sample wajah yang sudah di crop



Gambar 3.3 Contoh sample wajah yang di riseze untuk database

3.1.1. Pengumpulan database wajah

Sample wajah yang telah dikumpulkan masih merupakan RGB image, sehingga terlebih dahulu diubah menjadi grayscale image. Untuk menjadi database, image yang telah diubah dalam bentuk grayscale harus diubah dalam bentuk matriks dengan cara membaca nilai matriks image pada tiap titik resolusi image tersebut. Sehingga didapatkan nilai matriks dengan besaran yang sama dengan nilai resolusi image, yakni 30 x 30. Untuk memudahkan pembentukan database, maka setiap 1 image akan dibaca dalam bentuk matriks kolom dan akan terbentuk matriks 900 x 1. Dan matriks setiap image akan digabungkan sehingga terbentuk suatu matriks baru yang merupakan gabungan dari seluruh image dari tiap kelas. Untuk proses pembentukan matriks image dapat dilihat pada gambar 3.4.

$$x_1 = \begin{bmatrix} x_{1,1} \\ \cdot \\ \cdot \\ x_{900,1} \end{bmatrix} \longrightarrow x = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdot & \cdot & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdot & \cdot & x_{2,m} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ x_{900,1} & x_{900,2} & \cdot & \cdot & x_{900,m} \end{bmatrix}$$

Gambar 3.4 Penggabungan nilai matriks tiap image

Untuk proses pembacaan database yang akan dijadikan sebagai input dengan menggunakan Matlab adalah sebagai berikut:

```
I = 1:dataPerKelas
cd('./nama_folder')
s = int2str(I);
gambar = imread(s,'jpg');
data = rgb2gray(gambar);
for row = 1:Image
    for col = 1:Image
```

```

        input(M*(I-1) + 1,(row-1)*Image + col) = data(row,col);
        end
end
cd('...')

```

3.1.2. Matriks Kovarian

Matriks kovarian merupakan nilai-nilai kovarian antara 1 dimensi sama dimensi lain dalam database untuk menata nilai-nilai kovarian lebih tersusun.

Matriks kovarian ini dapat dicari dengan rumus:

$$C = x \times x^T$$

C = matriks kovarian

x = matriks input

x^T = matriks input transform

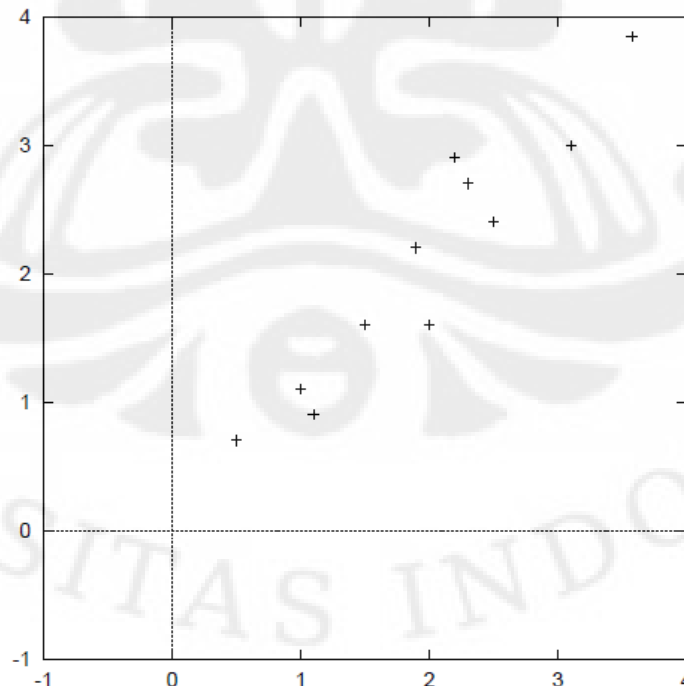
Tujuan pencarian matriks kovarian ini adalah untuk memudahkan pencarian eigen vector dan eigen value. Untuk mencari eigen vector dan eigen value dapat dicari dengan rumus:

$$C \times V = \lambda \times V$$

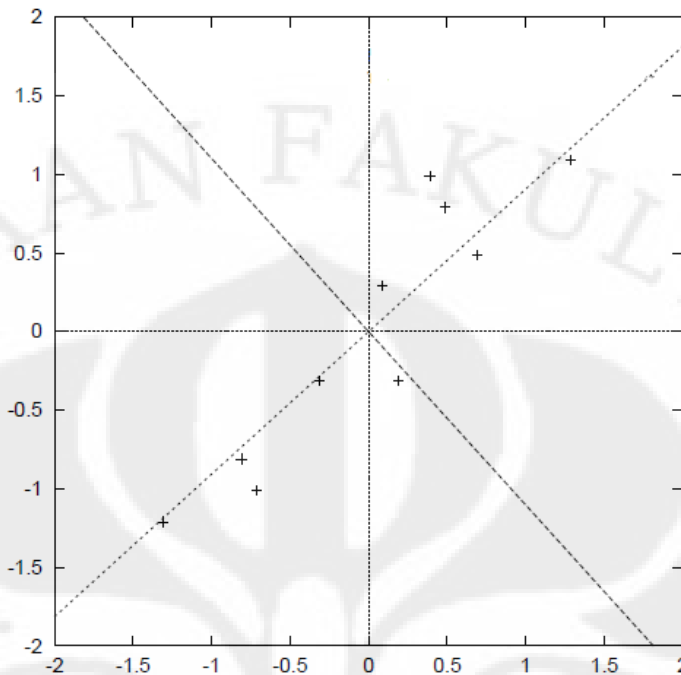
C = matriks kovarian

V = Eigen Vektor

λ = Eigen Value

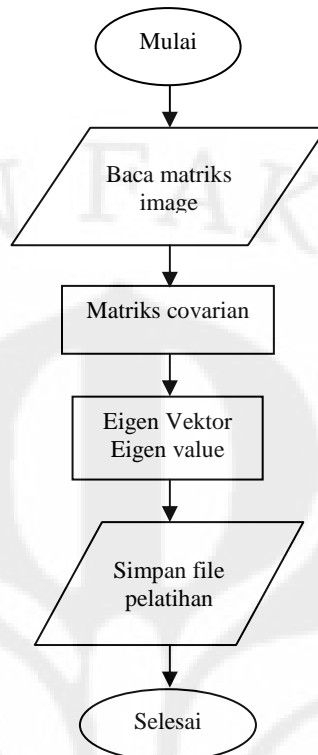


Gambar 3.5 Plot Matrik data



Gambar 3.6 Plot data matrik Kovarian

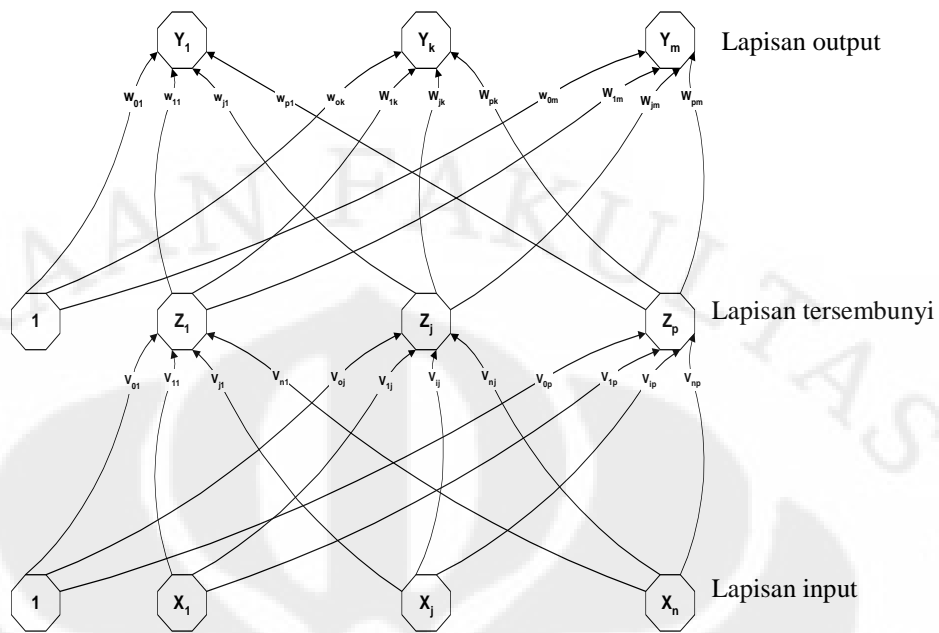
Pada gambar 3.5 terlihat plot matrik data yang akan digunakan vektor-vektornya masih berada pada posisi yang teracak. Kemudian setelah dicari nilai kovariannya maka pada gambar 3.6 terlihat data yang sudah dinormalisasi dengan vektor-vektor sudah tersusun dengan posisi eigen vektor yang sesuai dengan nilai eigennya. Proses penyusunan dilakukan setelah didapat eigen vector akan mewakili vektor yang memiliki arah yang sama dengan arah vektor pada matrik sebelumnya dan eigen value mewakili nilai dari arah vektor-vektor pada matrik baru. Nilai eigen value ini akan diurutkan dari nilai yang terbesar sampai yang terkecil. Kemudian eigen vector juga diurutkan bersesuaian dengan eigen value yang telah diurutkan sebelumnya, sehingga didapatlah matriks input baru dari eigen vektor yang sudah tersusun dan akan digunakan dalam pelatihan.



Gambar 3.7 Diagram Alir Proses pembentukan Input

3.2. Pembentukan Jaringan Syaraf Tiruan

Setelah terbentuknya matriks baru yang akan digunakan sebagai input, maka pembentukan jaringan syaraf tiruan bisa dilakukan. Metode yang digunakan adalah *Backpropagation* seperti yang ditunjukkan pada Gambar 3.8. Lapisan input (X) digunakan untuk menampung semua sampel masukan, dengan Lapisan tersembunyi (Z) yang berjumlah sekitar 200 buah dan dengan lapisan output (Y) berjumlah 5 yang sesuai dengan jumlah kelas dari sample. Antara lapisan input dan lapisan tersembunyi akan didapat nilai Nilai bobot layer tersembunyi (v_j) dan antara lapisan tersembunyi dan lapisan keluaran akan didapat nilai bobot keluaran (w_j). Jika target masih belum tercapai, maka proses akan diulang kembali dengan memasukan nilai output ke lapisan input. Kemudian mencari lagi nilai bobot pada lapisan tersembunyi (v_j) dengan menambahkan nilai bias lapisan tersembunyi (v_0), begitu juga dengan bobot pada lapisan keluaran (w_j) ditambahkan nilai bias pada lapisan keluaran (w_0)



Gambar 3.8 Arsitektur *Backpropagation*

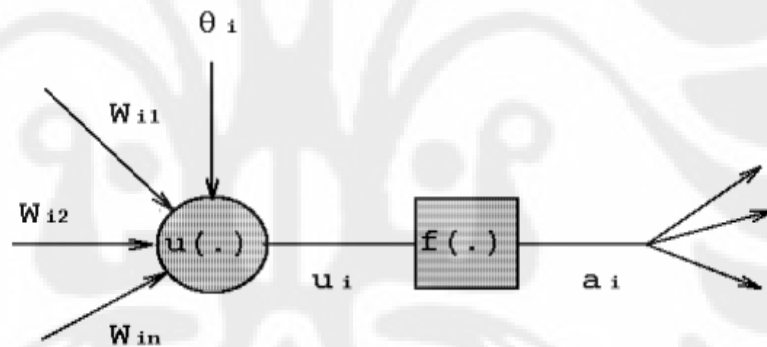
Pemakaian jaringan syaraf tiruan *Backpropagation* ini bertujuan untuk melakukan pembelajaran dan mengoreksi kembali nilai keluaran proses *feedforward*. Apabila antara nilai keluaran jaringan dengan nilai target yang diinginkan masih terdapat selisih *error* yang besar, maka nilai keluaran tersebut akan dijadikan sebagai input pada lapisan sebelumnya, kemudian dilakukan pembelajaran lagi pada jaringan syaraf tiruan sampai akhirnya didapatkan nilai keluaran yang sama atau mendekati nilai targetnya. Proses pembelajaran atau pelatihan jaringan syaraf tiruan ini mengarahkan jaringan pada satu nilai yang disebut vektor target sehingga proses ini bisa dikategorikan *supervised learning*. Proses perbaikan bobot, nilai bias dan perbaikan *gradien error* akan dijelaskan lebih lanjut pada tahap pelatihan.

3.2.1. Pelatihan

Selama perambatan maju, tiap unit masukan (x_i) menerima sebuah masukan sinyal ini ke tiap-tiap lapisan tersembunyi z_1, \dots, z_{10} . Tiap lapisan tersembunyi ini kemudian menghitung aktivasinya dan mengirimkan sinyalnya (z_j) ke tiap unit keluaran (seperti yang diilustrasikan oleh Gambar 3.8). Tiap unit keluaran (y_k) menghitung aktivasinya (y_k) untuk membentuk respon pada Jaringan Syaraf Tiruan untuk memberikan pola masukan.

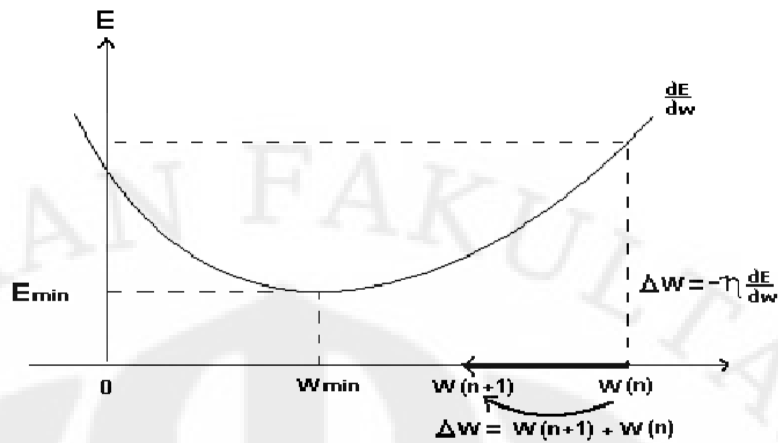
Selama proses pelatihan, tiap unit keluaran membandingkan perhitungan aktivasinya (y_k) dengan nilai targetnya (t_k) untuk menentukan kesalahan pola tersebut dengan unit tersebut. Kemudian dihitung faktor δ_k ($k = 1$). δ_k digunakan untuk menyebarkan kesalahan pada unit keluaran y_k kembali ke semua unit pada lapisan sebelumnya (unit-unit lapisan tersembunyi yang terhubung ke y_k). Nantinya nilai ini juga digunakan untuk mengupdate bobot-bobot antara lapisan keluaran dengan lapisan tersembunyi. Dengan cara yang sama, faktor δ_j ($j = 1, \dots, 10$) dihitung untuk tiap unit tersembunyi z_j . Nilai δ_j hanya digunakan untuk mengupdate bobot-bobot antara lapisan tersembunyi dan lapisan masukan.

Setelah seluruh faktor δ ditentukan, bobot untuk semua lapisan diatur secara serentak. Pengaturan bobot w_{jk} (dari unit tersembunyi z_j ke unit keluaran y_k) didasarkan pada faktor δ_k dan aktivasi z_j dari unit tersembunyi z_j didasarkan pada faktor δ_j dan aktivasi x_i unit masukan. Blok diagram fungsi aktivasi diilustrasikan pada Gambar 3.4.



Gambar 3.9 Blok Diagram Fungsi Aktivasi

Masing-masing bobot ($w_{i1}, w_{i2}, \dots, w_{in}$) kemudian ditambahkan dengan suatu nilai bias tertentu (θ_i). Lalu dengan menggunakan fungsi aktivasi *purelin* atau *tansig* maka akan didapatkan keluaran jaringan (a_i). Nilai a_i nantinya akan dibandingkan dengan vektor target yang ditetapkan diatas. Nilai yang diinginkan yaitu apabila nilai gradien perbaikan bobotnya minimum seperti yang diilustrasikan pada Gambar 3.9.



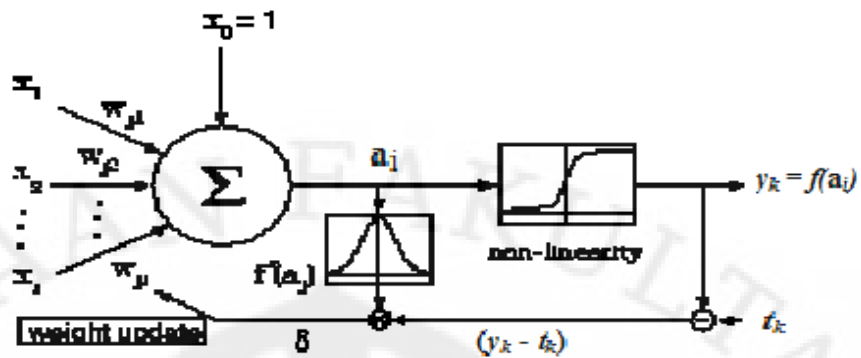
Gambar 3.10 Kurva Gradien Perbaikan Bobot

Pada Gambar 3.10 nilai perubahan bobot yang diperbaiki proporsional dengan gradien negatif yang dituliskan sebagai berikut:

$$\Delta W_{ij} = -\eta \frac{\partial E}{\partial w_{ji}} \quad (3.1)$$

Faktor proporsional η adalah laju belajar (learning rate, lr) yang mendefinisikan kedalaman langkah iterasi, dimana nilai laju belajar terletak antara 0 sampai 1. Proses iterasi berjalan lambat saat galat minimum hampir tercapai. Pemilihan parameter laju belajar yang terlalu besar membuat bobot yang terlalu besar sehingga membuat bobot yang dihasilkan overshoot terhadap galat E minimum. Laju belajar yang terlalu kecil membuat proses konvergensi lambat, dengan kata lain jaringan lebih lambat belajar. Kesalahan pemilihan laju belajar akan mengakibatkan osilasi di sekitar nilai galat E minimum. Pada penelitian ini, Jaringan Syaraf Tiruan dilatih dengan nilai laju belajar sebesar 0,01.

Apabila selisih antara keluaran jaringan dengan vektor target masih terdapat error maka nilai ini akan diumpan-balikan ke input sebagai bobot baru seperti yang diilustrasikan pada Gambar 3.11. Proses ini akan berulang sampai nilai gradien perbaikan bobotnya minimum. Proses perbaikan bobot secara rinci dijelaskan pada prosedur pelatihan.



Gambar 3.11 Proses Perbaikan Bobot Jaringan Syaraf Tiruan

Algoritma

Langkah 0 : Inisialisasi Bobot

Langkah 1 : Selama kondisi stopping FALSE, lakukan langkah 2-9

Langkah 2 : Untuk setiap pasang pelatihan, lakukan langkah 3-8

Feedforward:

Langkah 3 : Tiap unit masukan ($x_i, i = 1, \dots, n$) menerima sinyal input dan menghantarkan sinyal ini ke semua unit lapisan di atasnya (unit tersembunyi).

Langkah 4 : Setiap unit tersembunyi ($x_i, i = 1, \dots, p$) jumlahkan bobot sinyal masukannya,

$$z_in_j = v_{oj} + \sum_{i=0}^n x_i v_{ij} \quad (3.2)$$

v_{oj} = bias pada unit tersembunyi j aplikasikan fungsi aktivasinya untuk menghitung sinyal keluarannya, $z_j = f(z_in_j)$, dan kirimkan sinyal ini keseluruh unit pada lapisan diatasnya (unit keluaran)

Langkah 5 : Tiap unit keluaran ($y_k, k = 1, \dots, m$) jumlahkan bobot sinyal masukannya,

$$y_in_k = w_{ok} + \sum_{i=0}^n z_j w_{jk} \quad (3.3)$$

w_{ok} = bias pada unit keluaran k dan aplikasikan fungsi aktivasinya untuk menghitung sinyal keluarannya, $y_k = f(y_in_k)$.

Backpropagation:

Langkah 6 : Tiap unit keluaran (y_k , $k = 1, \dots, m$) menerima pola target yang saling berhubungan pada masukan pola pelatihan, hitung kesalahan informasinya,

$$\delta_k = (t_k - y_k)f'(y_{in_k}) \quad (3.4)$$

hitung koreksi bobotnya (nantinya digunakan untuk memperbaharui w_{jk}),

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (3.5)$$

hitung koreksi biasnya (digunakan untuk memperbaharui w_{ok} nantinya), dan kirimkan δ_k ke unit-unit pada lapisan dibawahnya,

Langkah 7 : Setiap unit tersembunyi (z_j , $j = 1, \dots, p$) jumlahkan hasil perubahan masukannya (dari unit-unit lapisan diatasnya),

$$\delta_{in_j} = \sum_{i=0}^n \delta_k w_{jk} \quad (3.6)$$

kalikan dengan turunan fungsi aktivasinya untuk menghitung informasi kesalahannya,

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (3.7)$$

hitung koreksi bobotnya (nantinya digunakan untuk memperbaharui v_{oj}),

Langkah 8 : Tiap unit keluaran (y_k , $k = 1, \dots, m$) update bias dan bobotnya ($j = 0, \dots, p$):

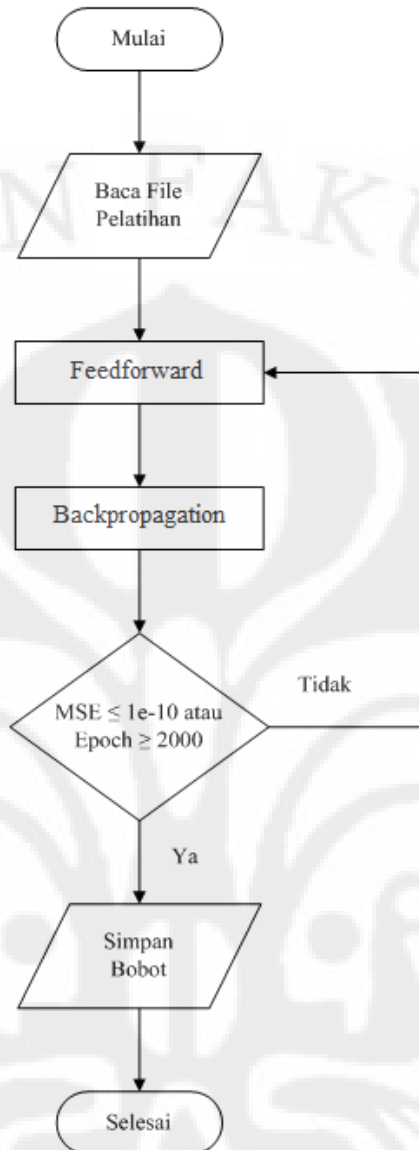
$$w_{jk} (\text{baru}) = w_{jk} (\text{lama}) + \Delta w_{jk} \quad (3.8)$$

Tiap unit lapisan tersembunyi (z_j , $j = 1, \dots, p$) update bias dan bobotnya ($i = 0, \dots, n$):

$$v_{ij} (\text{baru}) = v_{ij} (\text{lama}) + \Delta v_{ij} \quad (3.9)$$

Langkah 9 : Test kondisi berhenti.

Untuk proses diagram alir dari Jaringan Saraf Tiruan dapat dilihat pada gambar 3.12



Gambar 3.12 Diagram Alir Proses Pelatihan Jaringan Syaraf Tiruan

Untuk proses simulasi dengan program Matlab:

- Menentukan nilai epoch, alpha,

```
epoch = 2000;
a = 0.9;
```

- Menentukan lapisan

```
lapisanHidden = 200;
kelas = 5;
P = lapisanHidden;
M = kelas;
```

```
Z = zeros(P,1);
Y = zeros(M,1);
```

- Menentukan target

```
T = [0.8 0.2 0.2 0.2 0.2;0.2 0.8 0.2 0.2 0.2;0.2 0.2 0.8 0.2
0.2;0.2 0.2 0.2 0.8 0.2;0.2 0.2 0.2 0.2 0.8];
```

- Menampilkan hasil pelatihan

```

pref(L)= mse(diffE(:,F));
figure(1);
plot(pref)
    if pref(L) < 0.001
        fprintf('converged at epoch: %d\n',L);
        break
    end

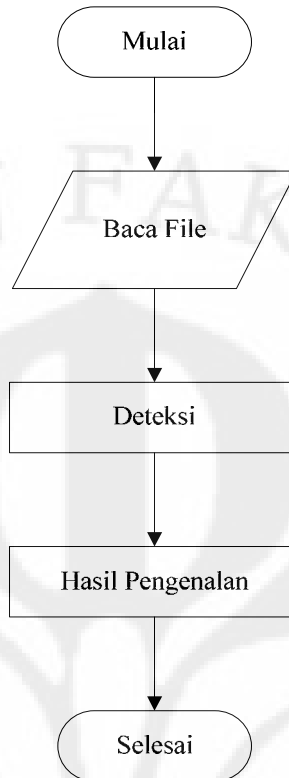
```

3.2.2. Proses Pengenalan

Setelah melakukan proses pelatihan *Backpropagation* Jaringan Saraf Tiruan dan telah menghasilkan nilai keluaran yang diinginkan, maka sistem siap untuk melakukan proses pengenalan terhadap data-data wajah yang bukan termasuk kedalam data yang sudah ditraining. Data-data yang akan dikenali merupakan data wajah baru yang masih dari kelas yang sama, sehingga nanti akan diketahui data wajah baru tersebut akan termasuk kedalam kelas yang sudah ditraining.

Pada proses pengenalan dengan *Backpropagation* Jaringan Saraf Tiruan ini yang digunakan adalah nilai bobot keluaran dari proses training yang akan dicocokkan dengan nilai bobot keluaran wajah yang baru. Jadi data wajah baru yang diujikan juga mengalami proses pencarian nilai bobot keluaran dari data pengujian. Dari bobot keluaran yang baru inilah nantinya dicari nilai MSE yang akan digunakan untuk mengenali data yang diuji

Pada proses pengenalan ini, data wajah yang baru diambil dari 5 kelas yang sama dengan data yang telah ditraining. Untuk masing-masing kelas data yang diambil berjumlah 10 sample untuk tiap kelasnya, sehingga total data wajah yang diujikan berjumlah 50 data. Untuk diagram alir proses pengenalan dapat dilihat pada gambar 3.13



Gambar 3.13 Diagram Alir Proses Pengenalan

Untuk simulasi proses pengenalan dengan menggunakan Matlab:

- Membaca data pengujian

```

G = 1:dataUji
cd('./Uji')
s = int2str(I);
gambar = imread(s,'jpg');
data = rgb2gray(gambar);
for row = 1:Image
    for col = 1:Image
        input(M*(I-1) + 1,(row-1)*Image + col) = data(row,col);
    end
end
cd('.')
  
```

- Mencari nilai error dari data pengujian

```

for F = 1:M
    diff(:,F) = Y2 - T(:,F);
    error(1,F) = mse(diff(:,F));
end
  
```

- Melakukan identifikasi

```

[C,I] = min(error);
if I == 1
    fprintf('foto ke-%d adalah foto Thuso.\n',G)
elseif I == 2
    fprintf('foto ke-%d adalah foto Adi.\n',G)
elseif I == 3
    fprintf('foto ke-%d adalah foto Dedi.\n',G)
  
```



```
elseif I == 4
    fprintf('foto ke-%d adalah foto Riki.\n',G)
elseif I == 5
    fprintf('foto ke-%d adalah foto Nandar.\n',G)
end
```

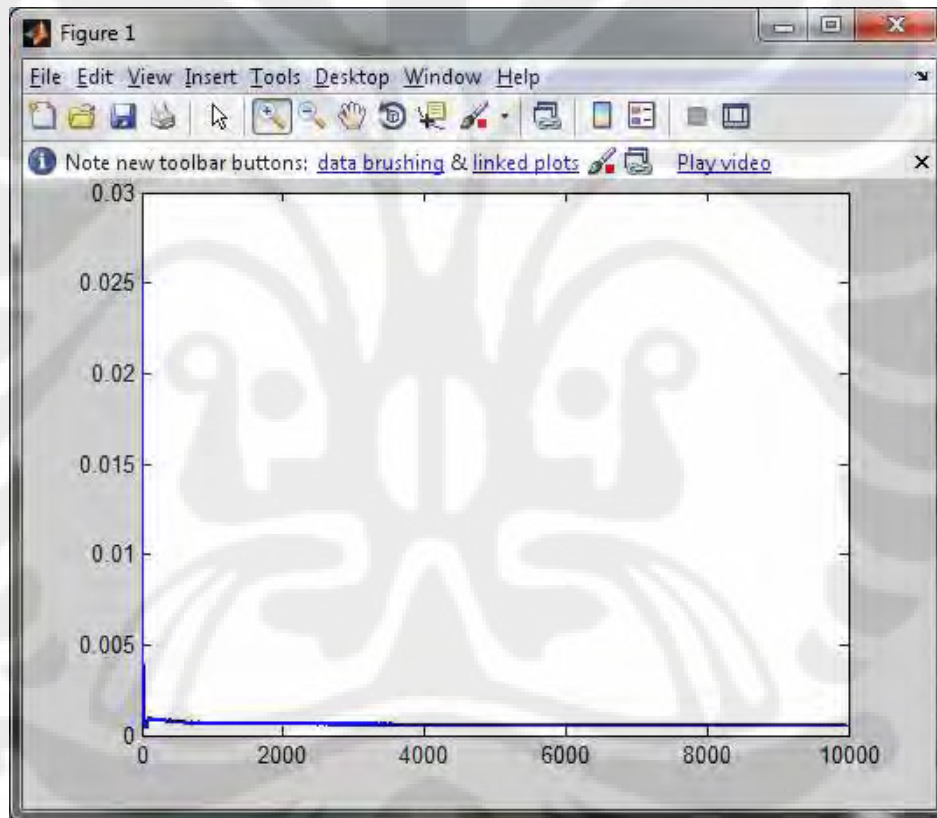


BAB 4 HASIL DAN ANALISA

4.1. Hasil Pelatihan

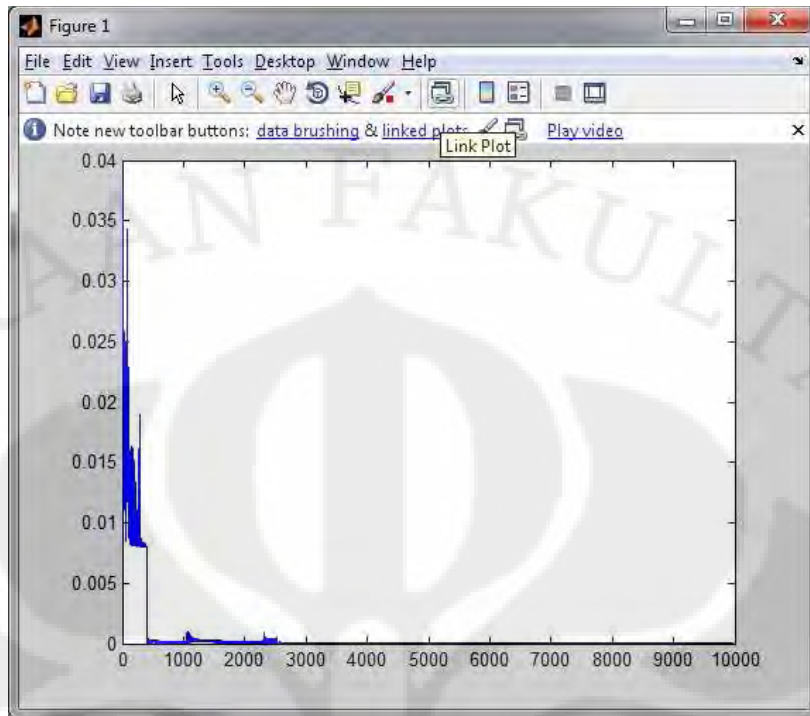
Setelah dilakukan pelatihan terhadap sample data, maka akan didapat nilai error yang ditampilkan dalam grafik fungsi error. Penampilan grafik ini sangat penting untuk mengetahui tingkat keberhasilan dari proses pelatihan yang dilakukan. Dari grafik dapat dilihat hasil error yang didapat selama proses pelatihan.

Berikut adalah Grafik dari hasil pelatihan yang telah dilakukan.



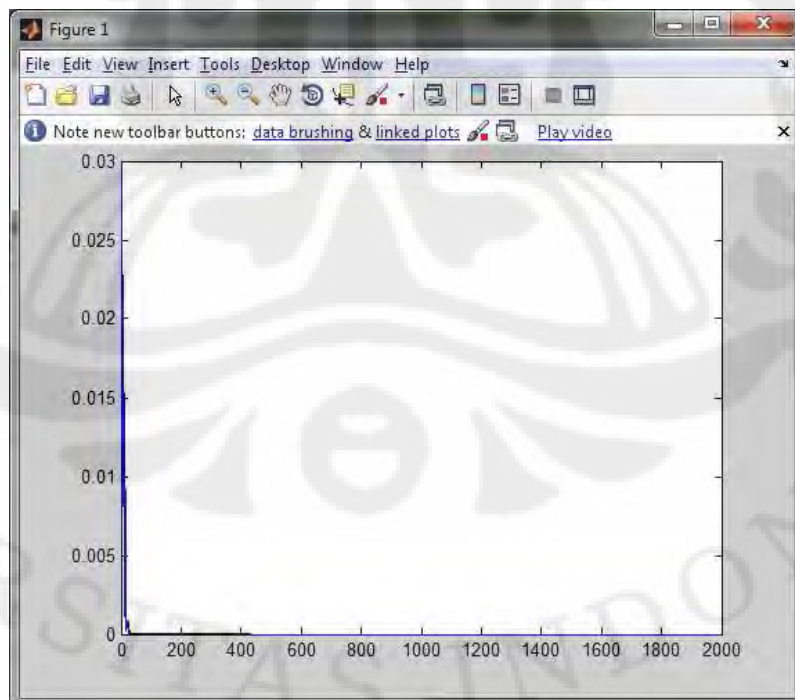
Gambar 4.1 Grafik Fungsi *Error* untuk 10.000 Epoch

Pada gambar 4.1 diatas proses pembelajaran dilakukan secara maksimal, yaitu dilakukan sampa dengan 10.000 (sepuluh ribu) Epoch. Dimana nilai *error* yang didapat cukup bagus dengan nilai yang kecil dan tidak terjadi kesalahan.



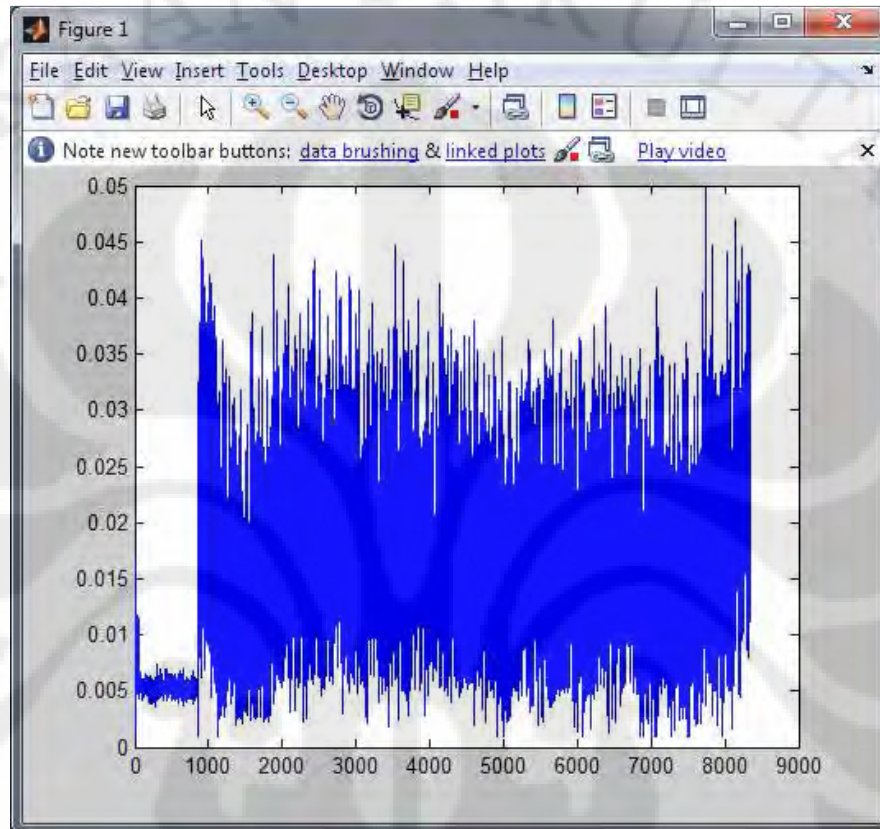
Gambar 4.2 Grafik Fungsi Error dengan Alpha 0.9

Pada gambar 4.2 dapat dilihat proses pembentukan grafik fungsi *error* untuk alpha yang diset sebesar 0.9 dengan epoch sebesar 10.000.



Gambar 4.3 Grafik Fungsi dengan alpha 0.6 untuk 2.000 epoch

Pada gambar 4.2 grafik yang ditampilkan hasil grafik untuk alpha sebesar 0.6 dengan batas 2.000 epoch, untuk proses kali ini waktu yang dibutuhkan jauh lebih sedikit. Dengan diperkecilnya nilai alpha, grafik *error* akan memiliki nilai yang kecil juga.



Gambar 4.4 Grafik Fungsi Error yang didapat dari kesalahan proses pembelajaran

Pada gambar 4.4 dapat dilihat garafik fungsi error yang tidak semestinya. Dari grafik dapat diketahui jika nilai *error* yang didapat dari proses pembelajaran nilainya semakin membesar. Ini bisa terjadi jika ada kesalahan dalam mencari nilai bobot pada lapisan keluaran maupun nilai bobot pada lapisan tersembunyi.

Proses pembelajaran pada sistem yang telah dirancang dengan menggunakan metode Jaringan Syaraf Tiruan dengan proses *Backpropagation* memerlukan waktu yang cukup lama. Terutama jika ingin mendapatkan hasil sampai dengan 10.000 epoch. Waktu yang diperlukan untuk proses pembelajaran adalah selama 45 menit, sedangkan untuk 2000 epoch waktu yang diperlukan sekitar 25 menit dan tergantung dari banyaknya data yang digunakan pada proses pembelajaran. Khusus untuk gambar 4.4 proses pembelajaran yang salah bisa memakan waktu sampai dengan 13 jam.

4.2. Pengujian

4.2.1. Pengujian dengan Menggunakan Kamera CCTV Inframerah

Untuk mengukur tingkat keakuratan kinerja dari sistem *Backpropagation* Jaringan Syaraf Tiruan yang telah dirancang, maka perlu dilakukan pengujian pengenalan terhadap beberapa sampel baru. Sampel baru ini hanya diambil dari 5 kelas yang dipilih berdasarkan kondisi ekspresi wajah yang variatif. Setiap kelas diambil 10 sample image baru untuk proses pengujian dan tidak diikutsertakan dalam proses pelatihan. Ini bertujuan untuk menghindari hasil pengujian dengan tingkat keberhasilan 100%. Dalam prosen pengujian, sample baru akan dilakukan proses pengenalan dengan sample yang telah mengalami proses pelatihan. Setelah dikenali, sample baru tersebut akan dikategorikam masuk kedalam kelas yang mana. Dalam proses pengujian ini, tidak selamanya sample yang diuji sesuai dengan kelas yang seharusnya.

Dalam proses pengenalan kesamaan sample baru dengan sample yang telah mengalami proses pembelajaran karena adanya kemiripan pola. Untuk hasil pengujian dengan menggunakan sample baru dapat dilihat pada Tabel 4.1, 4.2 dan 4.3.

Tabel 4.1 hasil pengujian denga alpha 0.2

No.	Nama	Hasil pengujian		Koreksi		Akurasi
		1	2	Benar	Salah	
1.	Thuso	Thuso Dedi Riki Thuso Thuso Thuso Thuso Thuso Thuso Thuso	Thuso Thuso Thuso Thuso Thuso Thuso Thuso Thuso Thuso Thuso	18	2	90%
2.	Adi	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	20	0	100 %

No.	Nama	Hasil pengujian		Koreksi		Akurasi
		1	2	Benar	Salah	
3	Riki	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	20	0	100%
4	Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	20	0	100%
5	Nandar	Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar	Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar	20	0	100%
Total						98%

Tabel 4.1 hasil pengujian dengan alpha 0.6

No.	Nama	Hasil pengujian		Koreksi		Akurasi
		1	2	Benar	Salah	
1.	Thuso	Thuso Thuso Thuso Thuso Adi Thuso Adi Thuso Thuso Thuso Thuso	Thuso Thuso Thuso Thuso Dedi Thuso Thuso Thuso Thuso Thuso Thuso	17	3	85%
2.	Adi	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	20	0	100%

No.	Nama	Hasil pengujian		Koreksi		Akurasi
		1	2	Benar	Salah	
3	Riki	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	20	0	100%
4	Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	20	0	100%
5	Nandar	Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar	Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar	20	0	100%
Total						97%

Tabel 4.1 hasil pengujian dengan Alpha 0.9

No.	Nama	Hasil pengujian		Koreksi		Persentase
		1	2	Benar	Salah	
1.	Thuso	Thuso Dedi Dedi Thuso Dedi Thuso Riki Thuso Thuso Thuso	Thuso Thuso Thuso Dedi Thuso Nandar Thuso Thuso Thuso	14	6	70%
2.	Adi	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	20	0	100 %

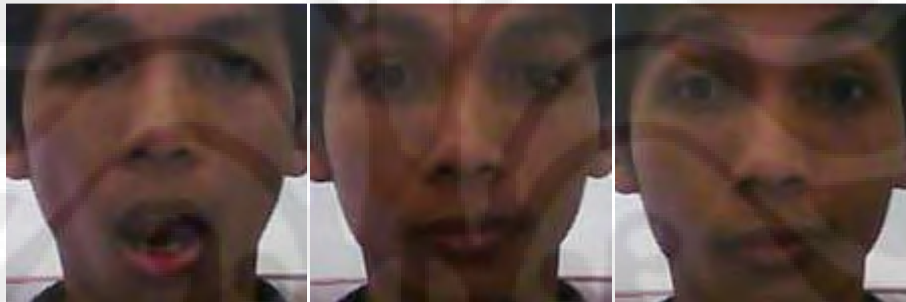
No.	Nama	Hasil pengujian		Koreksi		Akurasi
		1	2	Benar	Salah	
3	Riki	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	20	0	100%
4	Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	20	0	100%
5	Nandar	Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar	Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar Nandar	20	0	100%
Total						94%

4.2.2. Hasil Pengujian dengan Menggunakan Kamera *Visible*

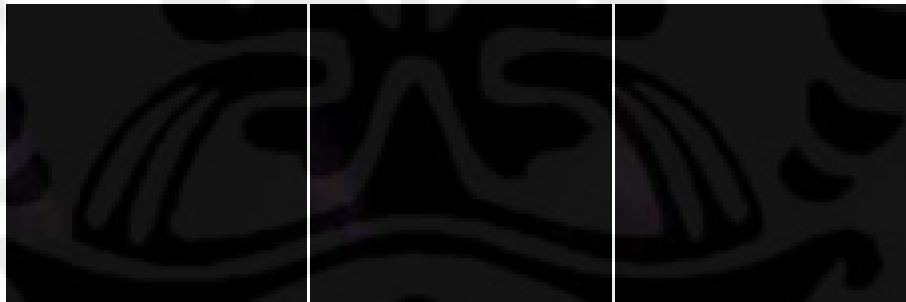
Untuk mengetahui hasil perbandingan antara kamera CCTV inframerah dengan kamera *visible* biasa dalam penerapan sistem pengenalan dengan metode Backpropagation Jaringan Saraf Tiruan, maka dilakukan pengujian dengan data sample wajah baru yang diambil dengan menggunakan kamera *visible* biasa. Untuk perbandingan ini sample yang akan diujikan diambil hanya berasal dari satu kelas yaitu kelas ke-3 dengan jumlah sample sebanyak 10 buah. Pengujian akan dilakukan dengan nilai alpha yang memiliki hasil pengujian terbaik, yaitu dengan alpha 0.2. Ada dua kondisi pengambilan sample dengan menggunakan kamera *visible*, pertama pada kondisi malah hari dengan pencahayaan lampu yang cukup terang, kedua pada kondisi malam hari tanpa cahaya.

Pengambilan sample dengan kamera *visible* dengan pencahayaan yang cukup memungkinkan kamera *visible* untuk dapat mengambil gambar dengan

kemampuan maksimal yang hanya dapat mengambil gambar dengan pencahayaan yang cukup. Untuk hasil tangkapan gambar dengan kamera *visible* dengan pencahayaan yang cukup dapat dilihat pada gambar 4.5. Sedangkan pengambilan sample dengan menggunakan kamera *visible* pada malam hari tanpa bantuan cahaya bisa dikatakan diluar kemampuan dari kamera *visible* dalam mengambil gambar. Untuk hasil tangkapan dengan kamera *visible* pada malam hari tanpa pencahayaan dapat dilihat pada gambar 4.6. Pada gambar 4.6 tersebut hanya menampilkan warna hitam, itu karena kamera *visible* memang tidak dapat mengambil gambar tanpa cahaya. Sehingga hasil yang didapat adalah warna hitam atau gambar gelap. Untuk perbandingan diambil juga data uji yang diambil dalam kondisi gelap dengan menggunakan kamera CCTV inframerah untuk kelas ke-3, gambarnya dapat dilihat pada gambar 4.7.



Gambar 4.5 Gambar sample uji dengan kamera *visible* dengan pencahayaan



Gambar 4.6 Gambar sample uji dengan kamera *visible* tanpa pencahayaan



Gambar 4.7 Gambar sample uji dengan kamera CCTV inframerah

Setelah sample dengan dua kondisi yang disebutkan sebelumnya selesai diambil, maka dilakukan pengujian dengan database yang telah dilatih pada proses pelatihan. Untuk proses pengujian pertama dilakukan pengujian pada sample wajah yang diambil menggunakan kamera *visible* dengan pencahayaan. Hasil pengujian sample dengan pencahayaan dapat dilihat pada Tabel 4.4. Selanjutnya dilakukan pengujian dengan sample wajah yang diambil menggunakan kamera *visible* tanpa pencahayaan. Untuk hasil pengujian sample tanpa pencahayaan dapat dilihat pada Tabel 4.5. Untuk pembandingan dengan hasil kamera CCTV inframerah hasil pengujiannya dapat dilihat pada Tabel 4.6

Tabel 4.4 Hasil pengujian sample kamera *visible* dengan pencahayaan

Nama kelas	Hasil pengujian		Koreksi		Akurasi
	1	2	Benar	Salah	
Dedi	Thuso Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	19	1	95%

Tabel 4.5 Hasil pengujian sample kamera *visible* tanpa pencahayaan

Nama kelas	Hasil pengujian		Koreksi		Akurasi
	1	2	Benar	Salah	
Dedi	Riki Riki Riki Riki Riki Riki Riki Riki Riki Riki	Adi Adi Adi Adi Adi Adi Adi Adi Adi Adi	0	20	0%

Tabel 4.6 Hasil pengujian sample kamera CCTV inframerah

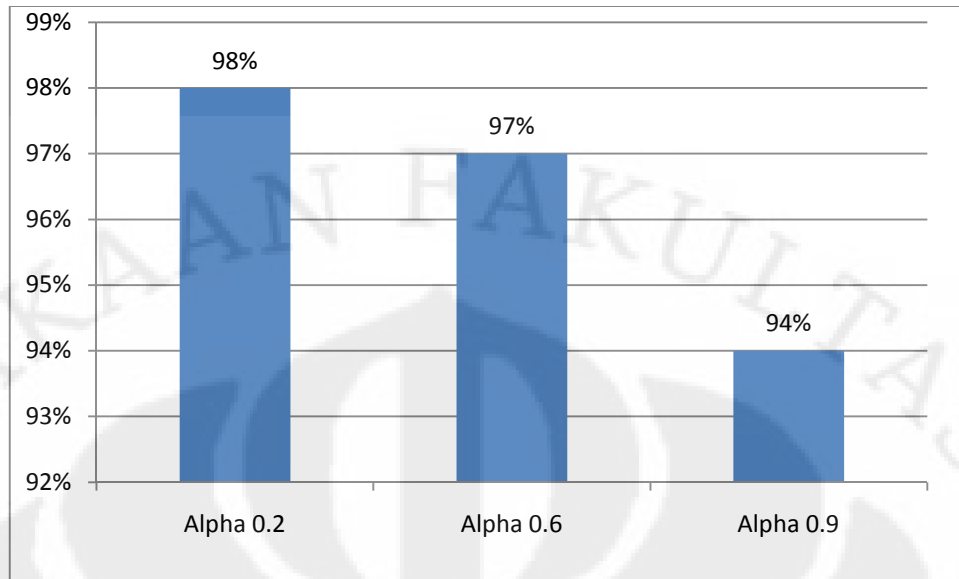
Nama kelas	Hasil pengujian		Koreksi		Akurasi
	1	2	Benar	Salah	
Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi Dedi	20	0	100%

4.3. Analisa

4.3.1. Analisa Pengujian dengan Menggunakan Kamera CCTV Inframerah

Dari hasil 3 pengujian yang ditampilkan pada Tabel 4.1, 4.2 dan 4.3. dimana dilakukan pengujian terhadap 5 kelas, dari tiap kelas data sample yang diujikan 10 untuk tiap-tiap kelas. Dalam proses pengujian pertama dilakukan pada setingan alpha 0.2 didapat untuk kelas pertama ada 2 kesalahan yang terjadi, dimana sample 2 dan 3 terjadi kesalahan pengenalan sedangkan pada kelas yang lain tidak terjadi kesalahan. Lalu dilakukan pengujian ke dua dan pada kelas pertama tidak terjadi kesalahan yang sebelumnya terdapat kesalahan pengenalan. Sedangkan pada empat kelas yang lain terjadi keberhasilan pengenalan untuk tiap sampelnya. Sehingga untuk alpha 0.2 terjadi tingkat kesalahan sebesar 98%.

Pada pengujian kedua, dilakukan dengan setingan alpha sebesar 0.6 pada kelas pertama dalam pengujian pertama terjadi 2 kesalahan pada sample 5, dan 7 sedangkan pada kelas yang lainnya masih terjadi keberhasilan pengenalan. pada pengujian kedua terjadi 1 kesalahan pada kelas pertama yaitu pada sample 5. Sedangkan untuk kelas yang lain masih tidak terjadi kesalahan pengenalan, sehingga total tingkat kesalahan pada setingan alpha 0.6 sebesar 97%. Untuk selanjutnya dilakukan pengujian dengan setingan alpha sebesar 0.9 dan didapat hasil pada pengujian pertama terjadi 4 kesalahan pada kelas pertama yaitu pada sample 2, 3, 5 dan 7. Sedangkan pada pengujian ke dua terdapat 2 kesalahan pada kelas pertama yaitu pada sample 5 dan 7. Sedangkan pada kelas yang lain masih berhasil dilakukan pengujian, sehingga pada pengujian terakhir dengan alpha 0.9 memiliki tingkat keberhasilan sebesar 94%. Hasil grafik proses pengenalan dapat dilihat pada gambar 4.8.



Gambar 4.8 Grafik persentase tingkat keakurasian pengujian

Kesalahan yang terjadi pada proses pengujian tersebut disebabkan nilai error yang didapat dari hasil pembelajaran masih terlalu besar antar sample yang diujikan dengan nilai bobot yang dihasilkan dari proses training. Error yang terlalu besar ini bisa disebabkan oleh perbedaan sample wajah yang ekspresinya terlalu ekstrim pada saat pengambilan, sehingga sulit untuk dikenali. Sedangkan adanya sample yang tidak terdapat kesalahan karena sample wajah tidak terdapat perbedaan ekstrim dalam ekspresi wajah ketika dilakukan pengambilan sample.

Dari 3 pengujian yang dilakukan didapat hasil keakuratan dari sistem Jaringan Syaraf Buatan ini memiliki tingkat keakurasian untuk alpha 0.2 sebesar 98%, untuk alpha 0.6 sebesar 97% dan untuk alpha 0.9 sebesar 94%. Jika dibandingkan dengan prinsip Jaringan Syaraf Tiruan yang dapat mendekati hasil 100% karena kemampuannya dalam beradaptasi dengan data yang dihasilkan, maka hasil pada proses pengujian kali ini sudah berhasil. Yang cukup berpengaruh dalam masalah keakurasian hasil yang didapat pada proses pengujian adalah pada proses pengambilan sample wajah, juga terjadi perbedaan dalam masalah jarak pada pengambilan sample wajah, sudut tangkap kamera terhadap wajah yang tidak sama satu sama lain, posisi wajah tiap sample yang tidak tegak lurus terhadap kamera. Proses pengambilan sample ini juga sangat mempengaruhi nilai error yang besar dalam proses pengenalan.

4.3.2. Analisa Pengujian dengan Menggunakan Kamera *Visible*

Dari dua kali pengujian yang dilakukan dengan pengujian pertama dilakukan terhadap sample yang diambil menggunakan kamera *visible* dengan pencahayaan. Hasil yang didapat pada pengujian 1, terjadi kesalahan pengenalan terhadap sample 1 sedangkan pada pengujian 2 tidak terdapat kesalahan pengenalan. Hasil yang didapat ini tidak jauh berbeda dengan pengujian yang dilakukan dengan menggunakan kamera CCTV inframerah, dimana pada kelas ke-3 pengujian dengan kamera CCTV didapat keberhasilan 100% dalam proses pengenalan. Sedangkan hasil pengenalan dengan sample dari kamera *visible* didapat keberhasilan sebesar 95% dan hanya memiliki perbedaan sebesar 5% dalam hasil pengenalan

Sedangkan dalam pengujian dengan sample dari kamera *visible* yang diambil tanpa pencahayaan. Pada pengujian 1 terjadi kesalahan dalam pengenalan kelas atau seluruh sample, dimana yang dikenali justru dari kelas ke-4. Hal ini disebabkan oleh nilai bobot keluaran dari sample uji yang dihasilkan mendekati dengan nilai target untuk kelas ke-4. Begitu juga dengan pengujian 2, yang dikenali adalah kelas ke-2 dengan nilai bobot keluaran dari sample yang diuji memang mendekati nilai target dari kelas ke-2. Hal ini menyebabkan pengujian dengan hasil sample yang diambil menggunakan kamera *visible* tanpa pencahayaan proses pengujiannya didapat hasil 0% atau bisa dikatakan proses pengujian gagal total. Hal ini disebabkan kamera *visible* yang memang tidak dapat mengambil gambar tanpa pencahayaan, sehingga sample yang diambil berwarna gelap dan tidak memperlihatkan gambar yang bagus. Hal inilah yang menyebabkan pemilihan penggunaan kamera CCTV inframerah lebih unggul dari pada kamera *visible* ketika tanpa pencahayaan atau kondisi gelap.

BAB 5

KESIMPULAN

Setelah melakukan melakukan pengujian sistem pada bab 4, dapat diambil kesimpulan bahwa:

1. Besarnya epoch berpengaruh lamanya waktu yang diperlukan untuk proses training.
2. Kesalahan dalam proses training akan berpengaruh pada nilai error yang didapat.
3. Nilai error yang ditampilkan pada grafik error tergantung pada settingan nilai alpha, dengan nilai yang besar nilai error juga semakin besar dan dengan nilai yang kecil maka error yang didapat semakin kecil
4. Faktor penentuan nilai alpha berpengaruh pada hasil pengenalan, dengan nilai alpha 0.2 keberhasilan pengujian sebesar 98%. Dengan nilai alpha sebesar 0.6 keberhasilan pengujian sebesar 97% dan dengan nilai alpha keberhasilan pengujian sebesar 94%.
5. Perbedaan pengujian sample kamera CCTV inframerah dengan kamera *visible* yang diambil pada kondisi dengan pencahayaan terdapat selisih hasil pengenalan sebesar 5%
6. Perbedaan pengujian sample kamera CCTV inframerah dengan kamera *visible* yang diambil pada kondisi tanpa pencahayaan didapat hasil pengenalan 100% untuk kamera CCTV inframerah dan 0% untuk kamera *visible*.
7. Penggunaan kamera CCTV inframerah lebih baik dari pada kamera *visible* terutama untuk digunakan pada kondisi tanpa cahaya

DAFTAR ACUAN

- [1] Dipri A, A. 2009. Skripsi: Pendeteksi Sistem Pengenalan Penyakit Jantung Dengan Metode Jaringan Syaraf Tiruan
- [2] Eli Yani. 2005. *Pengantar Jaringan Syaraf Tiruan*. MateriKuliah.Com
- [3] Kusumoputro, Benyamin. 2001. Bahan Belajar Mahasiswa, Jaringan Neural Buatan. Fakultas Ilmu Komputer Universitas Indonesia
- [4] Zhao, Shuyan and Rolf-Rainer Grigat. July 9-11, 2005 An Automatic Recognition System In The Near Infrared Spectrum. Leipzig, Germany.
- [5] http://en.wikipedia.org/wiki/Covariance_matrix diakses pada 8 April 2010
- [6] <http://id.wikipedia.org/wiki/Inframerah> diakses pada 8 April 2010
- [7] http://id.wikipedia.org/wiki/Jaringan_saraf_tiruan diakses pada 8 April 2010
- [8] http://id.wikipedia.org/wiki/Night_vision diakses pada 8 April 2010
- [9] <http://id.wikipedia.org/wiki/Neuron> diakses pada 10 Mei 2010