Aplikasi Pengolahan Suara untuk Request Lagu

Achmad Basuki[1], Miftahul Huda[2], Tria Silvie Amalia[1]
[1] Jurusan Teknologi Informasi
[2] Jurusan Teknik Telekomunikasi
Politeknik Elektronika Negeri Surabaya - Institut Teknologi Sepuluh Nopember
Kampus ITS Keputih Sukolilo Surabaya 60111
Telp.(+62) 32-5947280, Fax(+62) 31-5946114
E-mail: basuki@eepis-its.edu, huda@eepis-its.edu

Abstrak

Ketika komputer multimedia telah banyak berubah untuk memastikan kemampuan dalam mengubah perintah dengan format analog seperti suara, musik dan video menjadi format digital begitu juga sebaliknya menyebabkan data-data digital semakin banyak digunakan, begitu juga dengan file musik. Sehingga muncul kesukaran dalam mencari file musik yang melimpah.

Pada penelitian ini dibuat sebuah sistem pengolahan suara manusia dengan jaringan saraf tiruan metode propagasi balik (back propagation) menggunakan personal computer. Sinyal suara analog mula-mula dicuplik menjadi sinyal digital dengan kecepatan cuplik 12000 Hz. Untuk mendapatkan fitur sinyal yang akan diproses pada jaringan saraf tiruan sinyal suara ditransformasikan ke domain frekuensi dengan Fast Fourier Transform (FFT) 256 point. Hasil FFT selanjutnya diproses dengan jaringan saraf tiruan back propagation untuk melakukan pengenalan. Seratus sampel suara dari sepuluh kata yang emnyebutkan judul lagu yang berbeda digunakan sebagai input pada proses pelatihan jaringan saraf tiruan. Hasil pengujian proses pengenalan suara menunjukkan keberhasilan 98 %.

***Kata Kunci**: jaringan saraf tiruan propagasi balik, backpropagation, pengolahan suara, FFT.*

1. Pendahuluan

Perkembangan sistem multimedia menyebabkan data-data digital makin banyak digunakan[1] memunculkan kesulitan dalam mencari data yang melimpah. Hal tersebut mendorong diciptakan kemudahan-kemudahan yang mampu melayani manusia berkomunikasi dengan komputer atau benda-benda digital layaknya manusia dengan manusia. Salah satu kemudahan yang diusahakan dalam penelitian ini adalah suatu aplikasi pengolahan suara untuk akses remote ke komputer yang digunakan untuk pemanggilan file musik menggunakan ucapan. Sehingga file musik dalam jumlah besar tidak menjadi penghalang dalam pencarian file tersebut serta dapat mempercepat pemanggilan file.

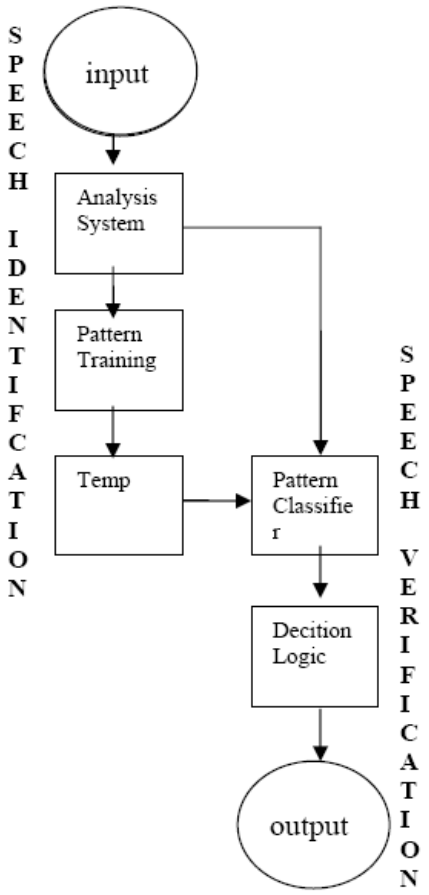
Aplikasi ini memudahkan kita dalam mencari file musik dengan bahasa manusia sehari-hari. Ucapan manusia sebagai masukan diubah menjadi perintah yang berfungsi untuk memanggil file musik.

Sinyal suara diproduksi oleh pergerakan pita suara dengan kontrol otak. Analisa suara berdasarkan pada analisa harmonik. Sinyal suara diolah sebagai berikut; setelah filter anti-aliasing, sinyal mikropon disampling dan dianalisa pada saat windowing dengan durasi yang digambarkan oleh waktu konstan[2]. Kemampuan telinga manusia yang mengesankan dalam mengenali kata yang sama dengan variasi panjang berbeda akan ditirukan dalam sistem ini dan dianalisa pada frot-end detection.

Tahap berikutnya adalah proses pembelajaran dengan neural network Dalam menyelesaikan permasalahan John-Paul Hosom, Ron Cole, dan Mark Fanty[4] menggunakan metode neural network untuk pengolahan suara dengan 130 node input untuk 5 frame data. Kemungkinan terburuk, telah ditemukan 82% benar. Hal tersebut telah ditunjukkan pula oleh Bourlard dan Wellekens (JEPIT 89) dan Richard dan Lippmann (Neural Perhitungan 1991). Dengan demikian neural network dapat digolongkan mampu memberi cukup data pelatihan dan menyembunyikan node.

2. Blok Diagram Sistem

Identifikasi pengucapan merupakan tahap mengidentifikasi suara menjadi pola – pola yang digunakan sebagai acuan. Selanjutnya acuan tersebut digunakan untuk mengenali suara yang dimasukkan. Proses identifikasi pengucapan dapat dilihat melalui diagram pada gambar 1.



Gambar 1. Blok diagram sistem

2.1. Identifikasi Pengucapan

Masukan pengucapan merupakan masukan ucapan yang dijadikan sampel melalui microphon. Suara direkam dengan frekuensi sampel 12000 Hz. Dengan durasi maksimum 1600 ms. Kemudian disimpan dalam bentuk file wav, sedangkan nilainya disimpan dalam file txt. Nilai ini diproses melalui beberapa tahap berikutnya, yaitu sistem analisa dan pola belajar. Masukan untuk tiap-tiap judul lagu sebanyak 10 kali. Setiap 10 judul lagu dijadikan satu folder khusus sebagai referensi dalam pola belajar. *Variant* tiap waktu bisa saja suara kita berubah. Untuk pembelajaran ini penulis menggunakan 10 sampel untuk tiap judul lagu. Karena semakin banyak sampel kemungkinan keberhasilan semakin tinggi. Penulis pernah menggunakan 3 sampel untuk tiap judul lagu, hasilnya hanya 30% suara dapat dikenali. Untuk lebih jelasnya dapat kita lihat perbedaan suara dalam domain waktu untuk kata ”aceh” dalam 10 sampel.

Sistem Analisa merupakan sistem yang disediakan untuk menganalisa hasil dari produksi suara dengan pengambilan polanya yang kemudian dilanjutkan pada proses berikutnya. Sistem Analisa terdiri atas :

1. Deteksi Awal-Akhir
2. Frame Blocking
3. Windowing
4. Fast Fourier Transform (FFT)

(1) Deteksi Awal-Akhir

Sebelum masuk deteksi awal akhir dilakukan normalisasi amplitudo terlebih dahulu untuk mengatasi jarak antara mulut dengan mikrofon. Normalisasi amplitudo dilakukan dengan cara membagi semua nilai input dengan nilai maksimum dari input itu sendiri. Sehingga untuk semua sinyal masukan memiliki nilai maksimum yang sama yaitu 1. Hal ini digunakan untuk mengatasi jarak dekat atau jauhnya mulut dengan mikrofon. Sedangkan deteksi awal-akhir digunakan pada proses untuk mendeteksi mulai sinyal ucapan awal dan berakhir ketika sudah tidak diucapkan. Sehingga tidak disalah artikan untuk tiap sinyal yang masuk. Nilai power digunakan untuk membedakan *voice* atau bukan. Standar deviasi untuk membandingkan nilai power. Untuk sinyal dengan nilai power diatas standar deviasi dapat diambil nilai pada indeks awal dan nilai pada indeks terakhir dari sinyal masukan.

Sebelum masuk pada *frame blocking* sinyal terlebih dahulu disamakan jumlah datanya, sehingga bisa didapat jumlah frame dan panjang masing-masing frame yang sama. Walaupun tahap ini akan memperpanjang data dan proses pada jaringan saraf tiruan, tahap ini mampu memperkecil kesalahan pengenalan kata. Terutama untuk kata yang memiliki jumlah suku kata berbeda.

Setelah penyeragaman jumlah data tidak perlu penambahan data 0 pada proses *frame blocking* untuk jumlah data yang tidak sama. Karena sudah menemukan jumlah data yang sama tanpa mengubah data itu sendiri. Proses dilanjutkan dengan *frame blocking.*

(2) Frame Blocking

Frame blocking merupakan proses yang digunakan untuk membagi *voice* menjadi beberapa bagian. Untuk mempercepat proses komputasi. Sedangkan hasil dari frame blocking merupakan sinyal terpotong yang *discontinue*. Dalam penelitian ini ucapan di dibagi dalam 20 ms tiap frame, jadi ada 12000 Hz X 20 ms = 240 data sample untuk tiap frame. Masing-masing sinyal hasil dari *framing* adalah sinyal terpotong yang *discontinue*. Sinyal *discontinue* ini akan dilanjutkan dalam proses Windowing. Sinyal terpotong tersebut akan dilanjutkan dalam proses Windowing.

(3) Windowing

Sinyal terpotong yang *discontinue* tersebut dikalikan dengan fungsi window agar menjadi sinyal yang *continue*. Fungsi windowing yang digunakan dalam penelitian ini adalah window Hamming karena fungsi hamming dapat membuat data pada awal frame dan akhir frame mendekati nilai 0 dengan baik. Dengan demikian sinyal menjadi kontinyu. Hasil Windowing Frame pertama

(4) Fast Fourier Transform (FFT)

Proses *Fast Fourier Transform* (FFT) ini dilakukan setelah didapat sinyal kontinyu. FFT yang digunakan memakai 256 data. Sedangkan tiap frame ada 240 data sehingga data tersebut dilakukan penyaamaan data tiap

frame dengan menambahkan nilai nol pada akhir nilai *(zero padding)* untuk setiap frame blok. Original diferensial *power spectral*, masih terlihat adanya periodik selang *fundamental frequency*, yang menyebabkan tidak terlihatnya *frequency* pada *peak* di posisi nol.

Pengaruh inilah yang menyebabkan ketidakteraturan dari informasi periodik. Untuk menghilangkan pengaruh tersebut, nilai negatif yang terdapat disekitar posisi nol sampai awal *peak* pada frekuensi, dibuat menjadi nol. Dengan melakukan cara perbaikan penambahan nilai *peak* pada posisi frekuensi nol, serta penambahan nilai *peak* pada posisi frekuensi nol di spectral.

Data yang diambil dari hasil kepstrum hanya 32 data yang dapat mewakili tiap frame. Karena ada 80 frame, total keluaran dari FFT ini adalah 80 X 32 data = 2560 data. Proses kepstrum dapat melalui tahap-tahap sebagai berikut :

1. FFT 256
2. Log FFT 256
3. Invers FFT 256
4. FFT (liftering) 32

Selanjutnya data tersebut dijadikan calon masukan bagi jaringan saraf tiruan.

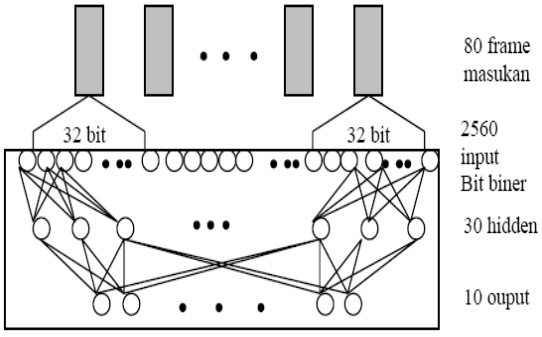
(5) Bit Biner

Pada proses ini data diubah dalam binary bit. Yang selanjutnya jadi masukan dalam neural network. Rumus untuk menjadikan data biner sebagai berikut :

$x_i / x_{max} >= 0.5$; maka $y_i = 1$ selain itu $y_i = 0$ (1)

(6) Pola Belajar

Pada proses ini digunakan Jaringan Saraf Tiruan *Multilayer Persepton* dengan metode pembelajaran *backpropagation*. Secara garis besar arsitektur jaringan saraf tiruan yang digunakan seperti gambar dibawah ini :



Gambar 5. Arsitektur Jaringan Saraf Tiruan

Input dari node pada neural network adalah 10 pattern X 80 frame X 16 bit biner = 12800 input biner yang aka diproses untuk setiap iterasi. Pada proses ini memakan waktu 8 jam hingga nilai MSE < 0.1. MSE adalah singkatan dari Mean Square Error yaitu nilai rata – rata kuadrat error yang telah di akar untuk setiap iterasi.

Sedangkan perhitungan error tiap pola dapat dilakukan perhitungan berikut[11].

$$EL = TL - YL \quad (2)$$

TL adalah nilai target yang ditentukan

YL adalah nilai keluaran hasil kalkulasi

EL adalah nilai *error* hasil kalkulasi antara ***YL*** dan ***TL***

Proses ini terdiri atas forward dan backward. Pada forward, data input akan dilanjutkan pada node hidden. Untuk mendapatkan node hidden jumlah dari input dikali bobot yang mula-mula diacak, pada iterasi berikutnya bobot ini di update pada proses backward. Hasil dari jumlah tersebut masukkan pada fungsi aktifasi. Untuk masalah berikut ini menggunakan fungsi aktifasi sigmoid biner. Fungsi sigmoid biner dirumuskan sebagai berikut :

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

Kelebihan proses ini adalah kemampuan dalam memperbaiki nilai errror hingga nilai error ynag diharapkan.

(7) Acuan

Karena pada penelitian ini menggunakan metode pembalajaran terawasi *backpropagation* maka output telah ditentukan terlebih dahulu. Setelah proses iterasi beberapa kali maka akan didapat nilai error min yang diinginkan. Yang disimpan disini adalah bobot terakhir untuk mengatasi permasalahan. Nilai bobot tersebut akan disimpan dan digunakan dalam proses verifikasi pengucapan. Nilai bobot terakhir tidak dapat ditampilkan didalam buku ini karena jumlah yang begitu besar. jumlah bobot input-hidden = 30X2560 = 76800. Jumlah bobot hidden-output = 30X10 = 300. Jumlah bias input = 30. Jumlah bias output = 10. Data disimpan dalam file.txt

2.2. Verifikasi Pengucapan

Tahap ini akan memasukkan nilai ucapan yang sebelumnya telah dianaliasa pada Sistem Analisa dengan menggabungkan dengan bobot yang telah baik dan hasilnya dicocokkan dengan template yang sesuai dengan input. Sehingga dihasilkan output berupa lagu yang sesuai dengan request. Tahap ini dapat digambarkan sebagai berikut :

1. Sistem Analisa
2. Klasifikasi Pola
3. Keputusan Logika

(1) Sistem Analisa

Sistem Analisa pada verifikasi pengucapan. Sama halnya dengan sistem analisa pada tahap identifikasi suara. Input adalah input dari hasil rekaman suara yang akan di uji sedangkan bobot didapat dari hasil pola belajar.

(2) Klasifikasi Pola

Proses ini menggunakan arsitektur jaringan saraf tiruan sama dengan pola belajar. Dalam proses ini masukan jaringan saraf tiruan adalah input ucapan dari hasil proses akan mendeteksi pola dari masukan suara

dengan cara mencocokkan dengan nilai hasil dari keluaran. Namun hanya melalui proses forward. Nilai bobot input-hidden, bobot hidden-output, bias hidden dan bias output didapat dari hasil pembelajaran.

(3) Keputusan Logika

Keluaran yang dihasilkan adalah user dapat mendengarkan hasil request lagu.

3. Hasil Pengujian

3.1. Pengambilan suara

Agar program berjalan sesuai dengan yang diharapkan, maka suara yang direkam haruslah dengan kemungkinan noise sangat kecil karena suara tersebut merupakan input utama yang akan sangat menentukan untuk proses selanjutnya. Syarat-syarat tersebut misalnya: dalam keadaan normal, tidak ada noice, soundcard baik, micropon baik, dsb. Hal yang paling disorot adalah masalah noise karena sangat mempengaruhi kualitas suara yang diambil dengan microphon. Misalkan mengambil sampel saat keadaan ramai dengan saat keadaan hening akan terlihat berbeda hasilnya. Bahkan noise juga bisa berasal dari desah nafas sendiri saat merekam suara, menghirup udara dengan keras-keras dan semua suara yang biasa datang bersamaan dengan voice.

Selain itu, sample suara juga memegang peranan penting dalam proses ini. Jika sample suara banyak mengandung noise maka hasil yang didapat tidak sesuai dengan yang diinginkan. Namun bila noise yang terekam stabil maka proses front end detection cukup bisa mengatasi. Proses pengambilan suara ini menjadi penting karena hal ini akan berpengaruh besar dalam pemrosesan sinyal suara pada setiap bagian proses, sehingga akan lebih baik bila user mengambil suara ditempat yang tenang dan suara diusahakan normal sehingga dapat memperkesil noise yang bersamaan datang dengan suara sendiri. Oleh sebab itu perekaman suara harus dilakukan dengan baik karena vital untuk menentukan baik tidaknya suara yang diambil.

Tabel 1. Kombinasi Jaringan Saraf Tiruan

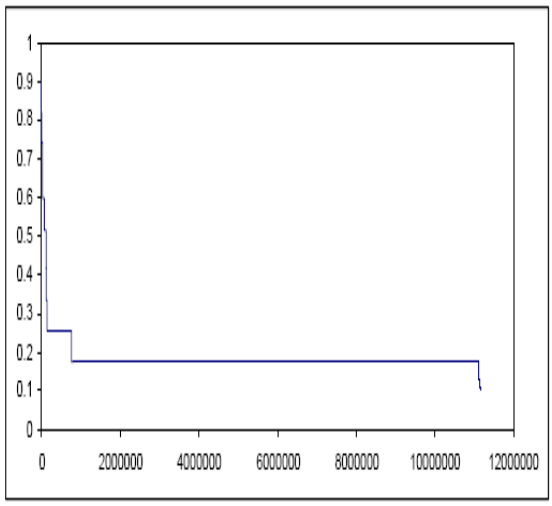
PERCOBAAN	PANJANG DATA AWAL	FFT	JUMLAH DATA LEARNING	TARGET ERROR	LEARNING RATE	JUMLAH NODE			ITERASI	WAKTU	GRAFIK ERROR	HASIL UJICOBA
						input	hidden	output				
1	tdak seragam	1024	2 X 10	0.1	0.5	32 X 15 X 10	30	10	1000	3jam	tdk konvergen	0%
2	tdak seragam	512	2 X 10	0.1	0.5	32 X 80 X 10	60	10	56802	32jam	tdk konvergen	8%
3	seragam	256	3 X 10	0.1	0.8	16 X 10 X 10	30	10	2149	12jam	tdk konvergen	10%
4	tdak seragam	256	3 X 10	0.1	0.5	16 X 80 X 10	30	10	65586	40jam	konvergen	30%
5	seragam	256	10 X 10	0.1	0.5	32 X 80 X 10	30	10	11185449	70jam	konvergen	98%

3.2 Analisa Sistem

(1) Hasil Sistem Jaringan Saraf Tiruan

Ada beberapa cara yang digunakan dalam proses learning dalam penelitian ini.

Penulis hanya mengkombinasikan jumlah node hidden, jumlah node input, *learning rate* dan beberapa variabel lain dengan satu hidden layer. Penelitian ini tidak menutup kemungkinan lain untuk memperbaiki sisi arsitektur jaringan maupun nilai koreksi yang lain. Tiga kombinasi yang telah dilakukan penulis dalam jaringn saraf tiruan dapat dilihat dalam tabel 1.
Dalam 5 kombinasi *learning* yang dilakukan hasil terbaik pada percobaan kelima.
Pembahasan berikutnya mengikuti percobaan *learning* kelima.



Gambar 6. Grafik penurunan error

Pada iterasi ke-11185449 nilai error telah mencapai 0.09934019 dan proses learning dihentikan. Waktu learning yang dibutuhkan $\pm$ 70 jam.

3.2. Pengujian Sistem Jaringan Saraf Tiruan

(1) Training Data Set

Pada tahap awal uji pengenalan dilakukan terhadap sinyal suara yang sama persis dengan yang telah ditrainingkan (*training data set*) dan didapat hasil bahwa error yang
terjadi sebesar 0% atau dengan kata lain keakuratan sistem untuk mengenali pola *training data set* mencapai 100%.

Tabel 2. Error Rate Pada Pengujian Training Data Set

Judul Lagu	% error 10 folder
kangen	0%
hitam	0%
denting	0%
dimensi	0%
pertama	0%
Judul Lagu	**% error 10 folder**
hilang	0%
aceh	0%
cindai	0%
duniaku	0%
pudar	0%
RATA2	0%

(2) Blind Data Set

Pengujian terhadap sinyal suara secara langsung dari *microphone* suara penulis untuk 10 pengucapan judul lagu masing-masing 10 kali (*blind data set*). Dari proses pengujian ini didapat error rata-rata sebesar 2 % atau dengan kata lain keakuratan sistem untuk pengenalan pola *blind data set* mencapai 98 %.

Tabel 3. Error Rate Pada Pengujian Blind Data Set

Judul Lagu	% error 10 folder
kangen	0%
hitam	10%
denting	0%
dimensi	0%
pertama	0%
hilang	10%
aceh	0%
cindai	0%
duniaku	0%
pudar	0%
RATA2	2%

3. Kesimpulan

Berikut adalah beberapa kesimpulan yang dapat diambil dari percobaan dan pengujian sebagai berikut:

(1) Untuk perbaikan fitur suara agar dapat menjadi panjang frame yang sama dengan tanpa mengubah fitur *voice*, sebaiknya fungsi ini dapat diletakkan setelah deteksi awal-akhir. Dengan interpolasi fitur dapat meminimalkan zerro padding sebelum masuk proses FFT. Fungsi yang dapat digunakan adalah fungsi interpolasi.

(2) Dari lima kali kombinasi jaringan saraf tiruan yang dilakukan pembuat aplikasi ini menemukan persentase kesalahan terkecil untuk pengenalan pengucapan dengan kombinasi berikut : melatih dengan data sbb : FRAME 80, INPUT 2560, HIDDEN 30, OUTPUT 10, LEARNING RATE 0.5, TARGET ERROR 0.1. Dengan data tersebut keakuratan sistem pengenalan suara untuk pengenalan *training data set* mencapai 100 % dan untuk pengenalan *blind data set* mencapai 98 %.

(3) Kesalahan pengenalan yang terjadi dapat diakibatkan adanya perbedaan yang terlalu besar antara sinyal suara yang hendak dikenali dengan sinyal suara yang dilatihkan, hal ini dapat diatasi dengan menambahkan/memperbanyak berbagai variasi pola kata pada saat pelatihan sehingga sistem jaringan lebih diperkaya pengetahuannya. Termasuk kondisi suara *learning* dan *testing* normal atau tidak.

(4) Terbuka penelitian lanjutan untuk memodifikasi arsitektur jaringan saraf tiruan dengan nilai-nilai terbaik, memperbesar jumlah perbendaharaan kata, dan penggunaan metode jaringan saraf tiruan yang lainnya sehingga pengenalan kata lebih akurat dan mampu bersifat independent.

Daftar Pustaka

[1] Cik Nor Anita Fairos binti Ismail, *Pengenalan Dan Definasi Multimedia*, Digital Audio and Video

[2] Chris J Wellekens, Communications Multimedia *Introduction to Speech Recognition, Using Neural Networks,* Institut Eurecom F-06904 Sophia-Antipolis

[3] Michael C.Mozer Professor, *Neural Network Speech Processing for Toys and Consumer Electronics,*http://www.cs.colorado.edu/~mozer/papers/speech.html, di download tanggal 30 Desember 2005.

[4] John-Paul Hosom, Ron Cole, and Mark Fanty, *Speech Recognition Using Neural Networks at the Center for Spoken Language Understanding,* Center for Spoken Language Understanding (cslu) Oregon Graduate Institute of Science and Technology, July 6, 1999.

[5] Dr. Arry Akhmad, Teknologi Pemrosesan Bahasa Alami sebagai Teknologi Kunci untuk Meningkatkan Cara Interaksi antara Manusia dengan Mesin, sidang Terbuka ITB, 23 Agustus 2004.

[6] http://www.speech.cs.cmu.edu/comp.speech/
[7] www.dacs.dtic.mil/techs/neural2.html, didownload tanggal 30 Desember 2005.
[8] http://students.if.itb.ac.id/~if19029/ann_al, didownload tanggal 31 Januari 2006.
[9] Arry Akhmad Arman, Proses Pembentukan dan Karakteristik Sinyal Ucapan, Dosen dan Peneliti di Departemen Teknik Elektro ITB
[10] Andry Haidar 23203058, *Keamanan Sistem Lanjut,*
[11] Bima Sena Bayu Dewantara, *Pelatihan Digital Signal Processing,* PENS – ITS, 26 – 27 Nopember 2004
[12] Rahardjo Budi, Pengenalan Pola Berbasis Neural Network