

# PENGENALAN UCAPAN VOKAL BAHASA INDONESIA DENGAN JARINGAN SARAF TIRUAN MENGGUNAKAN *LINEAR PREDICTIVE CODING*

I v a n a – L 2F0 98 632

e-mail : i\_van\_a@yahoo.com

Jurusan Teknik Elektro

Fakultas Teknik Universitas Diponegoro

## **Abstrak**

Suara merupakan alat komunikasi paling mendasar bagi manusia. Saat ini, suara tidak hanya digunakan untuk komunikasi antar manusia tetapi juga digunakan untuk berkomunikasi dengan perangkat-perangkat teknologi, karenanya topik pengenalan suara pun semakin berkembang. Dalam Tugas Akhir ini dilakukan pengenalan vokal Bahasa Indonesia dengan metode Linear Predictive Coding (LPC) untuk analisa sinyal, sedangkan untuk pengenalan pola digunakan Jaringan Saraf Tiruan (JST) dengan algoritma pelatihan Learning Vector Quantization (LVQ).

LPC merupakan suatu metode analisa sinyal yang menghasilkan sejumlah koefisien LPC sebagai representasi struktur formant. Jaringan Saraf Tiruan dengan pelatihan LVQ memiliki kemampuan mengklasifikasikan vektor masukan ke kelas target yang telah ditentukan sebelumnya. Mula-mula data suara disimpan dalam file wav, kemudian dianalisa dengan LPC. Koefisien LPC yang diperoleh dari analisa LPC selanjutnya menjadi vektor masukan JST untuk melatih jaringan. Hasil pelatihan berupa perubahan bobot jaringan sehingga diperoleh jaringan yang memiliki kemampuan pengklasifikasian. Jaringan tersebut kemudian diuji dengan mensimulasikannya pada data latihan dan data baru untuk menghasilkan persentase keberhasilan pengenalan. Eksperimen dilakukan dengan beberapa perubahan nilai parameter untuk memperoleh persentase pengenalan tertinggi.

Hasil pengujian menunjukkan bahwa persentase pengenalan tertinggi diperoleh pada orde LPC 10, panjang frame 10ms, dan jumlah epoch 1000. Pengujian data latihan memberikan hasil yang lebih baik dibandingkan data baru. Vokal /a/ dan /i/ merupakan yang paling tinggi tingkat pengenalannya dan terjadi beberapa pola kesalahan pengenalan yang terjadi berulang kali.

Kata kunci : pengenalan vokal, Linear Predictive Coding(LPC), Jaringan Saraf Tiruan, Learning Vector Quantization(LVQ).

## **I. PENDAHULUAN**

### **1.1 Latar Belakang**

Suara merupakan alat komunikasi yang paling dasar yang digunakan oleh manusia. Saat ini ketika teknologi semakin berkembang, maka manusia tidak hanya berinteraksi dengan manusia saja, tetapi seringkali juga harus berinteraksi dengan perangkat-perangkat teknologi.

Saat ini belum banyak perangkat teknologi yang mampu mengenali bahasa lisan manusia. Ini disebabkan oleh sulitnya perangkat teknologi menangkap pesan lisan, lalu menerjemahkannya dan akhirnya menjalankan perintah yang terkandung dalam pesan lisan tersebut. Untuk kepentingan itu, tentu dibutuhkan dibutuhkan perangkat tambahan yang mencakup teknologi pengenalan suara.

Pengenalan suara –dalam Tugas Akhir ini mengacu pada basis pelafalan Bahasa Indonesia– dapat dilakukan dengan menggunakan metode *Linear Predictive Coding* (LPC).

### **1.2 Tujuan**

Tujuan dari tugas akhir ini adalah

1. Menyusun program simulasi dengan program bantu MATLAB untuk mengenali suara vokal dalam lafal Bahasa Indonesia dengan Jaringan

Saraf Tiruan yang penciriannya menggunakan *Linear Predictive Coding* (LPC) dan menampilkannya dalam bentuk teks.

2. Membandingkan hasil pengenalan vokal Bahasa Indonesia dengan menggunakan Transformasi Fourier, Transformasi Wavelet Diskret dan *Linear Predictive Coding*.

### **Batasan Masalah**

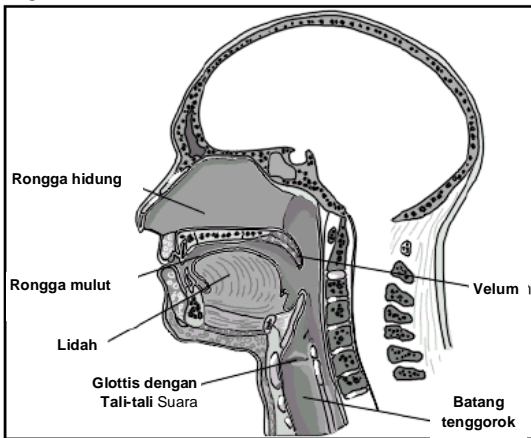
1. Data yang diolah adalah vokal dalam pelafalan Bahasa Indonesia.
2. Sampel suara diambil dari 20 orang responden yang masing-masing memiliki 25 file suara (5 sampel untuk tiap vokal).
3. Teknik pencirian suara dengan menggunakan *Linear Predictive Coding* (LPC).
4. Teknik pengenalannya melalui Jaringan Saraf Tiruan dengan algoritma pelatihan LVQ (*Learning Vector Quantization*).
5. Perbandingan terhadap Pengenalan Vokal dengan Transformasi Fourier dan Transformasi Wavelet Diskret dilakukan dari sudut pandang persentase pengenalan untuk data latihan, data baru dan data tunggal.

## II. DASAR TEORI

### 2.1 Teori Produksi Suara

Bunyi dapat terjadi karena gesekan atau benturan 2 benda atau lebih yang menimbulkan getaran udara dan diterima oleh telinga. Getaran udara ini dapat pula dihasilkan oleh alat ucap manusia seperti pita suara, lidah, dan bibir. Bunyi bahasa dibuat oleh manusia untuk mengungkapkan sesuatu.

Dalam pembentukan bunyi bahasa ada 3 faktor utama yang berperan yaitu sumber tenaga, alat ucap yang menimbulkan getaran, dan rongga pengubah getaran. Pada saat kita mengeluarkan nafas, paru-paru kita menghembuskan tenaga berupa arus udara yang mengalami perubahan pada pita suara yang terletak pada pangkal tenggorokan. Arus udara dari paru-paru itu dapat membuka kedua pita suara yang merapat sehingga mengakibatkan corak bunyi bahasa tertentu. Gerakan membuka dan menutup pita suara itu menyebabkan arus udara dan udara di sekitar pita suara berubah tekanannya atau bergetar. Perubahan bentuk saluran suara yang terdiri dari rongga faring, rongga mulut, dan rongga hidung menghasilkan bunyi bahasa yang berbeda-beda.



Gambar 1. Organ-organ produksi suara

Organ-organ pembentuk sinyal suara meliputi :

1. Bagian *Pulmonary Tract*; terdiri dari paru-paru dan batang tenggorokan.
2. Bagian *Larynx*; terdiri dari tali-tali vokal. Ruang antara pita suara disebut glottis.
3. Bagian *Vocal Tract*; terdiri dari pharynx, rongga mulut, dan rongga hidung.

Berdasarkan cara eksitasinya, suara dibagi menjadi 3, yaitu

1. Suara Voiced (*Voiced Sound*)  
Tekanan udara yang dibangkitkan dilewatkan pada *glottis* dengan pita suara yang tegang. Getaran pita suara akan mengubah aliran udara menjadi pulsa-pulsa yang *quasi-periodic*.
2. Suara Desah (*Fricative/Unvoiced Sound*)  
Suara desah diakibatkan penyempitan pada beberapa bagian *vocal tract*. Aliran udara yang berasal dari tenggorokan dipaksa melewati daerah

penyempitan tersebut dengan kecepatan yang cukup tinggi.

3. Suara letupan (*Plosive/Stop Sound*)

Adanya penutupan pada bagian *vocal tract* (biasanya di bagian depan) mengakibatkan aliran udara berkumpul di bagian belakang sehingga menimbulkan tekanan yang lebih besar. Bagian *vocal tract* yang tertutup ini dibuka secara tiba-tiba untuk menghasilkan eksitasi letupan

Vokal (*vowel*) adalah bunyi bahasa yang arus udaranya tidak mengalami rintangan dan kualitasnya ditentukan oleh 3 faktor: tinggi-rendahnya posisi lidah, bagian lidah yang dinaikkan, dan bentuk bibir pada pembentukan vokal tersebut. Pada saat vokal diucapkan, bagian depan, tengah atau belakang lidah dapat dinaikkan atau diturunkan bersama rahang. Dalam bahasa Indonesia terdapat 5 vokal yaitu /a/, /i/, /u/, /e/, dan /o/.

Tabel 1. Posisi lidah dalam vokal Bahasa Indonesia

Bagian Lidah	Depan	Tengah	Belakang
Tinggi	i		u
Sedang	e		o
Rendah		a	

### 2.2 Linear Predictive Coding (LPC)

Prinsip dasar dari pemodelan sinyal dengan menggunakan LPC adalah bahwa contoh sinyal ucapan  $s(n)$  pada waktu ke- $n$  dapat diperkirakan sebagai kombinasi linier dari  $p$  sampel sinyal ucapan sebelumnya yaitu :

$$s(n) \approx a_1 s(n-1) + a_2 s(n-2) + \dots + a_p s(n-p) \quad (1)$$

dengan koefisien  $a_1, a_2, \dots, a_p$  diasumsikan bernilai konstan selama *frame* analisis ucapan. Persamaan (1) kemudian diubah menjadi suatu persamaan yang mencakup suatu masa pembangkitan,  $G \cdot U(n)$ , menghasilkan

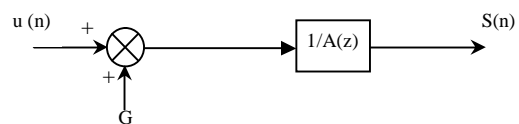
$$s(n) = \sum_{i=1}^p a_i \cdot s(n-i) + G \cdot u(n) \quad (2)$$

dengan  $u(n)$  adalah eksitasi ternormalisasi dan  $G$  adalah penguatan pembangkitan. Dengan mengekspresikannya dalam daerah  $z$ , diperoleh

$$S(z) = \sum_{i=1}^p a_i \cdot z^{-i} S(z) + G \cdot U(z) \quad (3)$$

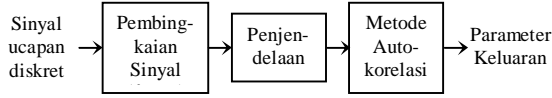
membawa kita pada fungsi alih

$$H(z) = \frac{S(z)}{G \cdot U(z)} = \frac{1}{\left(1 - \sum_{i=1}^p a_i \cdot z^{-i}\right)} = \frac{1}{A(z)} \quad (4)$$



Gambar 2. Model LPC

Analisis LPC pada dasarnya digunakan untuk mendapatkan koefisien LPC. Prosedur untuk mendapatkan koefisien LPC diperlihatkan pada Gambar 3.



Gambar 3. Blok Diagram Analisis LPC

Berdasarkan model pada Gambar 2, hubungan yang tepat antara  $s(n)$  dan  $u(n)$  adalah

$$s(n) = \sum_{k=1}^p a_k s(n-k) + G \cdot u(n) \quad (5)$$

Dasar permasalahan dalam suatu analisa prediktif linier (*linear predictive*) adalah menentukan kelompok koefisien LPC,  $\{a_k\}$ , secara langsung dari sinyal ucapan. Karakteristik spektral ucapan yang bervariasi sepanjang waktu menyebabkan koefisien LPC pada waktu yang diberikan,  $n$ , harus diestimasi dari suatu segmen pendek dari saat terjadinya sinyal ucapan sekitar waktu  $n$ .

Untuk menyelesaikan persamaan-persamaan yang harus dipecahkan untuk menentukan koefisien LPC, terlebih dulu ditetapkan ucapan jangka pendek dan segmen kesalahan pada waktu  $n$  sebagai

$$\begin{aligned} s_n(m) &= s(n+m) \\ e_n(m) &= e(n+m) \end{aligned} \quad (6)$$

$$\sum_m s_n(m-i)s_n(m) = \sum_{k=1}^p \hat{a}_k \sum_m s_n(m-i)s_n(m-k) \quad (7)$$

Dengan melihat bahwa jangka untuk bentuk  $\sum_m s_n(m-i)s_n(m-k)$  adalah jangka dari waktu pendek kovarians pada  $s_n(m)$ , yaitu

$$\phi_n(i,k) = \sum_m s_n(m-i)s_n(m-k) \quad (8)$$

Persamaan (7) dapat diekspresikan dalam bentuk singkat

$$\phi_n(i,0) = \sum_{k=1}^p \hat{a}_k \phi_n(i,k) \quad (9)$$

yang mana memberikan sekelompok persamaan  $p$  pada nilai  $p$  yang tidak diketahui.

Untuk menyelesaikan Persamaan (9) untuk mendapatkan koefisien LPC optimal ( $\hat{a}_k$ ), yaitu dengan menghitung  $\phi_n(i,k)$  untuk  $1 \leq i \leq p$  dan  $0 \leq k \leq p$ , dan kemudian menyelesaikan kelompok hasil persamaan simultan  $p$ . Salah satu metode untuk menyelesaikan persamaan tersebut yang akan digunakan pada Tugas Akhir ini adalah metode autokorelasi.

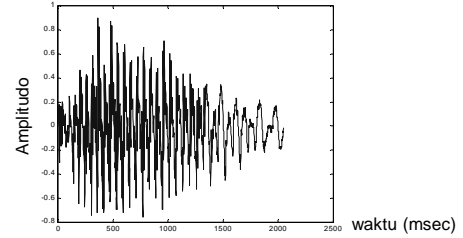
Pada metode autokorelasi, cara yang mudah untuk dapat membatasi limit penjumlahan  $m$  adalah dengan mengasumsikan bahwa nilai segmen sinyal ucapan  $s_n(m)$  adalah nol untuk nilai  $m$  di luar interval  $0 \leq m \leq N-1$ . Cara menerapkannya adalah dengan fungsi penjendelaan terhadap sinyal ucapan seperti persamaan di bawah:

$$s_n(m) = s(n+m) \cdot w(n), \quad 0 \leq m \leq N-1 \quad (10)$$

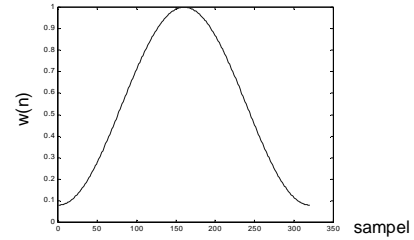
Dalam metode autokorelasi pada tahap penjendelaan digunakan jendela *hamming* sebagai berikut :

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad (11)$$

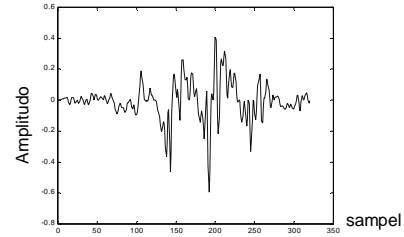
Tujuan penjendelaan sinyal ucapan pada Persamaan (10) adalah untuk meruncingkan atau memperhalus sinyal dekat  $m=0$  atau  $m=N-1$  sehingga dapat memperkecil sinyal kesalahan pada batas daerah tersebut. Gambar 4 memperlihatkan pengaruh penjendelaan terhadap sinyal suara heri-a1.wav



a. Masukan sinyal-suara



b. Hamming window



c. Sinyal keluaran setelah dijendelakan

Gambar 4. Pengaruh penjendelaan terhadap sinyal suara heri-a1.wav

Dengan demikian  $\phi_n(i,k)$  dapat dinotasikan kembali sebagai

$$\phi_n(i,k) = \sum_{m=0}^{N-1-p} s_n(m-i)s_n(m-k) \quad (12)$$

$$\text{atau} \quad \phi_n(i,k) = \sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m+i-k) \quad (13)$$

dengan nilai batas  $1 \leq i \leq p$  dan  $0 \leq k \leq p$ .

Dengan menganggap bahwa fungsi  $\phi_n(i,k)$  pada persamaan (13) hanya memiliki fungsi tunggal  $R_n(i-k)$  maka persamaan tersebut dapat dinotasikan kembali sebagai fungsi autokorelasi :

$$\phi_n(i,k) = R_n(|i-k|) = \sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m+i-k) \quad (14)$$

Jika fungsi ini memiliki sifat simetris maka  $R_n(k) = R_n(-k)$ , sehingga persamaan LPC pada persamaan (9) dapat dinotasikan sebagai :

$$\sum_{k=1}^p R_n(i-k) \hat{x}_k = R_n(i), \quad 1 \leq i \leq p \quad (15)$$

Persamaan (15) dapat dinyatakan dalam persamaan matriks sebagai berikut :

$$\begin{bmatrix} R_n(0) & R_n(1) & R_n(2) & \dots & R_n(p-1) \\ R_n(1) & R_n(0) & R_n(1) & \dots & R_n(p-2) \\ R_n(2) & R_n(1) & R_n(0) & \dots & R_n(p-3) \\ \dots & \dots & \dots & \dots & \dots \\ R_n(p-1) & R_n(p-2) & R_n(p-3) & \dots & R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \dots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ R_n(3) \\ \dots \\ R_n(p) \end{bmatrix} \quad (16)$$

Untuk mendapatkan harga koefisien LPC  $\alpha_k$  dapat dilakukan dengan menggunakan Persamaan (17)

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \dots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(0) & R_n(1) & R_n(2) & \dots & R_n(p-1) \\ R_n(1) & R_n(0) & R_n(1) & \dots & R_n(p-2) \\ R_n(2) & R_n(1) & R_n(0) & \dots & R_n(p-3) \\ \dots & \dots & \dots & \dots & \dots \\ R_n(p-1) & R_n(p-2) & R_n(p-3) & \dots & R_n(0) \end{bmatrix}^{-1} \begin{bmatrix} R_n(1) \\ R_n(2) \\ R_n(3) \\ \dots \\ R_n(p) \end{bmatrix} \quad (17)$$

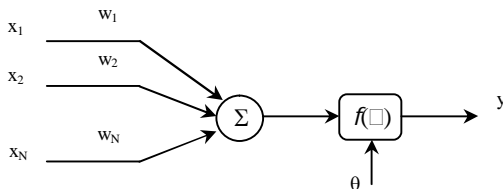
### 2.3 Jaringan Saraf Tiruan

JST bisa didefinisikan kurang lebih sebagai sistem komputasi yang didasarkan pada pemodelan saraf biologis (*neuron*) melalui pendekatan dari sifat-sifat komputasi biologis (*biological computation*). JST bisa dibayangkan berupa jaringan dengan elemen pemroses sederhana yang saling terhubung. Elemen pemroses berinteraksi melalui sambungan yang variabel, disebut bobot, dan bila diatur secara tepat dapat menghasilkan sifat yang diinginkan.

Pada tahun 1943 Warren McCulloch dan Walter Pitts memperkenalkan model matematis sederhana dari neuron, yang disebut model McCulloch-Pitts, terdiri dari banyak masukan dan satu keluaran. Model dari neuron pada Gambar 5 mempunyai persamaan

$$y = f\left(\sum_{i=1}^N w_i x_i - \theta\right) \quad (18)$$

- Dengan  $x_i$  = sinyal masukan,  $i = 1, 2, 3, \dots, N$  ( $N$  = jumlah simpul masukan)  
 $w_i$  = Bobot hubungan atau sinapsis  
 $\theta$  = *Threshold* atau bias  
 $f(\bullet)$  = Fungsi aktivasi  
 $y$  = Sinyal keluaran dari neuron



Gambar 5. Model Mc-Culloch-Pitts untuk neuron

McCulloch dan Pitts tidak memberikan suatu metode agar neuron dapat mengatur sendiri atau mengadaptasi bobotnya dalam proses pembelajaran. Pada 1949, Hebb

memberikan persamaan matematika sederhana yang dapat mengubah bobot neuron secara adaptif yaitu :

$$\Delta w_i(n) = \mu y(n) x_i(n) \quad (19)$$

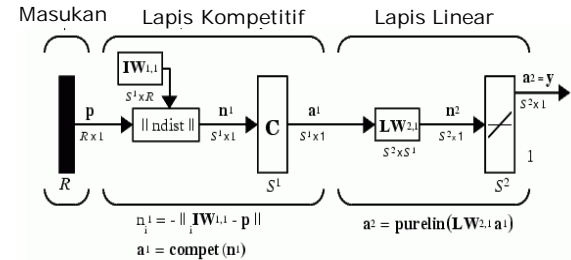
dengan  $\mu$  = tetapan laju pembelajaran positif pada semua n data (pasangan data masukan-keluaran).

Ide dasar JST adalah konsep belajar. Jaringan-jaringan belajar melakukan generalisasi karakteristik tingkah laku obyek. JST menyimpan ilmu pengetahuannya dalam nilai bobot sambungan (seperti *synapsis* dalam otak manusia) dan elemen-elemen (*neuron*) yang menghasilkan keluaran.

Terminologi belajar artinya membentuk pemetaan (fungsi) yang menggambarkan hubungan antara vektor masukan dan vektor keluaran. Untuk menyelesaikan permasalahan, JST memerlukan algoritma untuk belajar, yaitu bagaimana konfigurasi JST dapat dilatih untuk mempelajari data historis yang ada. Dengan pelatihan ini, pengetahuan yang terdapat pada data bisa diketahui dan direpresentasikan dalam bobot sambungannya.

### 2.3.1 Learning Vector Quantization (LVQ)

Jaringan LVQ mempunyai target yang akan dicapai. Lapisan kompetitif belajar mengenali dan mengklasifikasikan vektor-vektor masukan. Jika ada 2 vektor yang hampir sama, maka lapisan kompetitif akan menempatkan keduanya pada kelas yang sama. Dengan demikian LVQ belajar mengklasifikasikan vektor masukan ke kelas target yang ditentukan oleh pengguna. Arsitektur jaringan LVQ ditunjukkan pada Gambar 6.

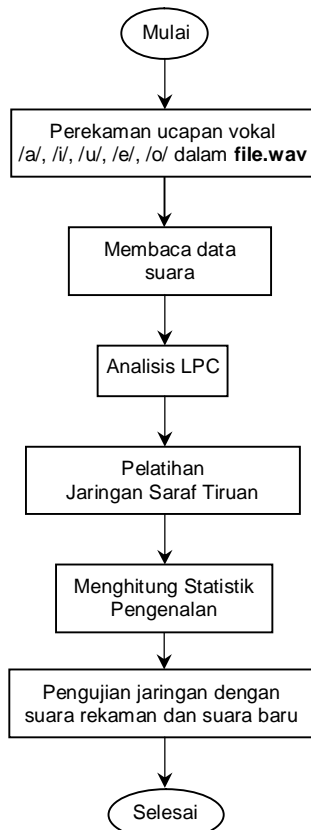


Gambar 6. Arsitektur jaringan LVQ

Jaringan LVQ terdiri dari 2 lapis yaitu lapis kompetitif dan lapis linear. Lapis kompetitif disebut juga *Self Organizing Map* (SOM). Disebut lapis kompetitif karena neuron-neuron berkompetisi dengan algoritma kompetisi yang akan menghasilkan neuron pemenang (*winning neuron*).

## III. PENERAPAN LINEAR PREDICTIVE CODING UNTUK PENGENALAN UCAPAN

Program simulasi untuk Tugas Akhir ini disusun melalui beberapa tahapan, yaitu perancangan, pembuatan, dan pengujian. Diagram alir rancangan program simulasi ditunjukkan pada Gambar 7.



Gambar 7 Diagram alir utama program simulasi

### Perekaman Ucapan Vokal

Ucapan vokal yang akan diolah dalam program simulasi terdiri dari 2 jenis, yaitu data suara terekam dan suara masukan baru.

Data suara terekam diperoleh dari data penelitian sejenis yang telah dilakukan sebelumnya yaitu Pengenalan Vokal Bahasa Indonesia dengan Jaringan Saraf Tiruan melalui Transformasi Wavelet Diskret (oleh Ignatius Leo May – L2F098630) dan Pengenalan Vokal Bahasa Indonesia dengan Jaringan Saraf Tiruan melalui Transformasi Fourier (oleh Doni Sitanggang – L2F098607). Perekaman dilakukan pada 20 orang responden, masing-masing responden memiliki 25 file suara (5 file suara untuk tiap vokal). Semua file suara disimpan dalam format wav.

Data suara baru berarti file suara yang direkam saat program dijalankan karena pengguna ingin mengujikan suara yang sama sekali baru. Untuk keperluan perekaman ini, sebelumnya ada beberapa parameter yang nilainya harus ditentukan terlebih dahulu meliputi frekuensi *sampling*/pencuplikan, durasi waktu perekaman, jumlah bit, dan nama file suara. Perekaman ini dilakukan oleh program MATLAB dan disimpan dalam format wav.

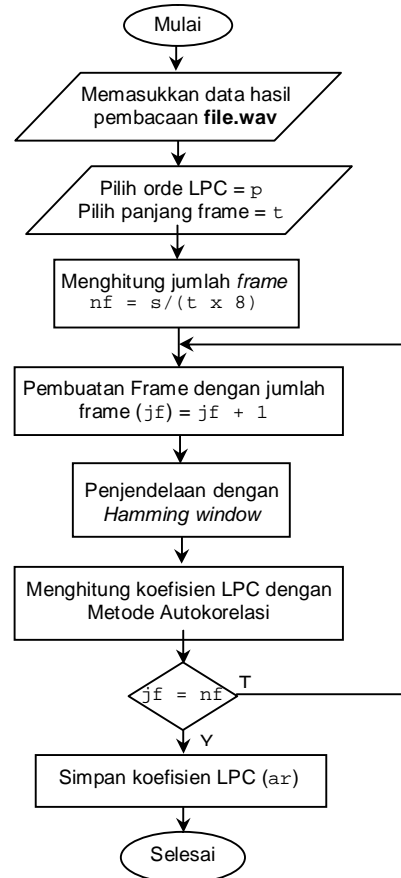
### Pembacaan File Suara

Hasil rekaman suara untuk selanjutnya dibaca oleh program MATLAB agar dapat diolah dengan

program simulasi. Panjang data yang diambil sebanyak 512 titik sampel, yang mana mengasumsikan bahwa lama ucapan tiap vokal sekitar 0,064 detik (512/8000).

### Analisis LPC untuk Mendapatkan Koefisien LPC

Hasil pembacaan file suara di atas selanjutnya dianalisis untuk memperoleh koefisien LPC. Metode yang digunakan dalam program simulasi ini untuk memperoleh nilai tersebut adalah metode autokorelasi.



Gambar 8. Diagram alir analisis LPC

Sinyal suara tidak bersifat stasioner, karena itu sinyal mula-mula harus di-frame dalam segmen-segmen pendek. Panjang frame yang umumnya digunakan untuk memperoleh kondisi kuasi-stasioner adalah 5 – 20 ms.

Setiap frame sinyal suara perlu dijendelakan untuk meminimalisasi diskontinuitas pada awal dan akhir frame, yaitu dengan meratakan sinyal menuju nol pada awal dan akhir frame. Pembingkaihan (*windowing*) dilakukan dengan menggunakan jendela *hamming* yang memiliki bentuk sebagai berikut :

$$w(n) = 0,54 - 0,46 \cos\left(\frac{2\pi n}{N-1}\right) \quad 0 \leq n \leq N-1$$

Penghitungan koefisien LPC dilakukan dengan metode autokorelasi yang dapat dirumuskan dengan persamaan matriks:

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \dots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(0) & R_n(1) & R_n(2) & \dots & R_n(p-1) \\ R_n(1) & R_n(0) & R_n(1) & \dots & R_n(p-2) \\ R_n(2) & R_n(1) & R_n(0) & \dots & R_n(p-3) \\ \dots & \dots & \dots & \dots & \dots \\ R_n(p-1) & R_n(p-2) & R_n(p-3) & \dots & R_n(0) \end{bmatrix}^{-1} \begin{bmatrix} R_n(1) \\ R_n(2) \\ R_n(3) \\ \dots \\ R_n(p) \end{bmatrix}$$

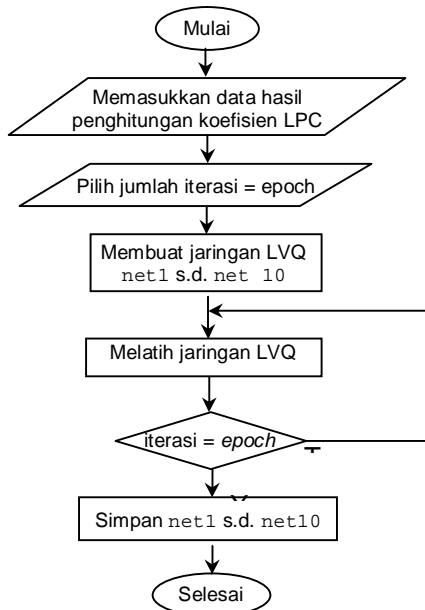
dengan :

$$R(k) = \sum_{n=0}^{N-1-k} s(n)s(n+k)$$

### Pelatihan Jaringan LVQ

Data hasil penghitungan koefisien LPC menjadi masukan untuk pelatihan jaringan agar diperoleh perubahan bobot pada neuron.

Pengklasifikasian vokal oleh JST menggunakan skema Pembagian Pasangan Biner (*Binary Pair Partitioned scheme*) sehingga 1 buah jaringan mengklasifikasikan 2 buah vokal. Dengan demikian digunakan 10 buah net atau jaringan yang masing-masing merupakan variasi dari 2 jenis vokal, yaitu /a/-/i/, /a/-/u/, /a/-/e/, /a/-/o/, /i/-/u/, /i/-/e/, /i/-/o/, /u/-/e/, /u/-/o/, dan /e/-/o/. Proses pelatihan jaringan ditunjukkan diagram alir pada Gambar 9.



Gambar 9. Diagram alir pelatihan jaringan LVQ

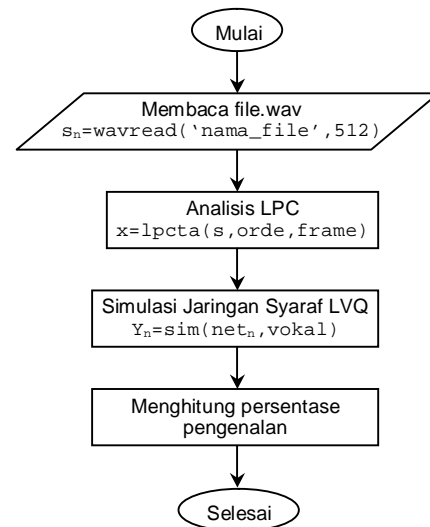
Pembuatan dan pelatihan jaringan memerlukan nilai variabel-variabel elemen masukan, jumlah *hidden neuron*, laju pembelajaran, algoritma pembelajaran, dan *epoch/iterasi*. Penentuan jumlah neuron dilakukan dengan cara eksperimen. Ada 2 cara yang dapat dipilih yaitu metode konstruktif/*evolutive* dan destruktif/*subtractive*. Pada metode konstruktif pertama-tama

dipilih jumlah neuron yang kecil kemudian diuji hasilnya, jika hasilnya tidak sesuai dengan yang diharapkan maka jumlah neuron ditambah lalu diuji lagi. Penambahan jumlah neuron dihentikan bila tidak lagi dihasilkan peningkatan hasil. Metode destruktif merupakan kebalikan dari metode konstruktif. Laju pembelajaran juga ditentukan dengan cara eksperimen, dimulai dari 0,05 sampai 0,9.

### Pengujian Jaringan

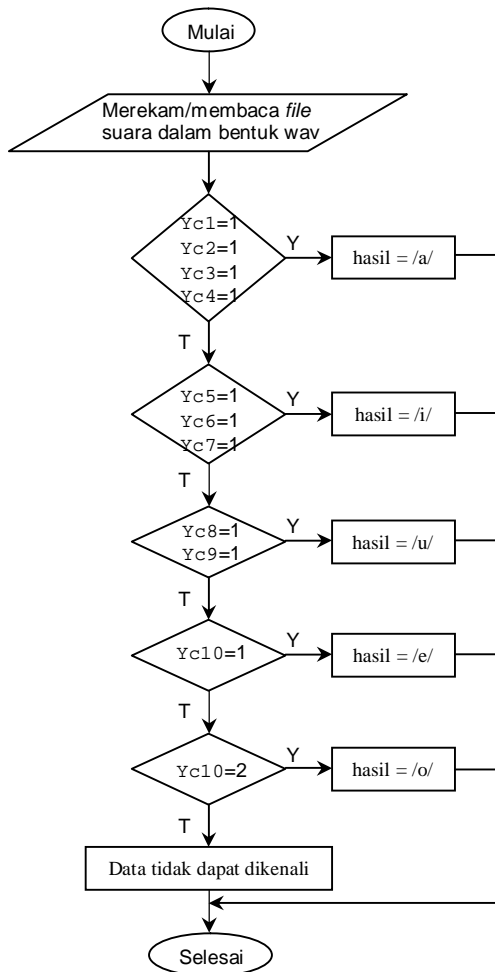
Dalam program simulasi, umumnya setiap pembacaan ucapan vokal, diawali dengan proses membunyikan *file* suara tersebut dan menampilkan grafik.

Data latihan yang akan diujikan di sini berupa 1 paket masukan yang terdiri dari 50 matriks koefisien LPC untuk masing-masing vokal. Matriks ini kemudian disimulasikan ke seluruh jaringan dari net1 sampai net10 untuk mendapatkan hasil pengenalan masing-masing vokal. Diagram alir untuk proses pengujian data latihan dan data baru ditunjukkan oleh Gambar 10.



Gambar 10. Diagram alir pengujian data latihan dan data baru

Masukan untuk pengujian data tunggal berupa matriks koefisien LPC dari suara baru yang baru direkam. Matriks koefisien kemudian disimulasikan pada semua net di jaringan yang memberikan persentase pengenalan terbaik. Gambar 11 menunjukkan diagram alir pengujian data tunggal yang akan memberikan hasil akhir berupa tampilan dalam bentuk teks vokal hasil pengenalan terhadap suara masukan.



Gambar 11. Diagram alir pengujian data tunggal

#### IV. ANALISA DAN PENGUJIAN

Pada program ini terdapat beberapa parameter yang dapat divariasikan, seperti yang ditunjukkan Tabel 2.

Tabel 2. Variasi nilai parameter

Parameter	Variasi
Orde LPC	8
	10
	14
	16
Panjang <i>Frame</i>	10 ms
	20 ms
Iterasi / <i>epoch</i>	500
	1000
	2000

Dengan variasi dari ketiga jenis parameter di atas, maka akan diperoleh 24 kombinasi jaringan.

#### 4.1 Hasil Pengujian Program Simulasi

Variasi 24 kombinasi nilai parameter diuji satu persatu hingga menghasilkan 24 set jaringan yang memiliki persentase keberhasilan tertentu, baik untuk data latihan maupun data baru.

Tabel 3. Hasil pengenalan data latihan untuk orde 8, panjang *frame* 10ms dan *epoch* 500

No	Nama net	Klasifikasi vokal	Persentase Pengenalan	
			a_a	a_i
1	net1	a dan i	a_a = 100	a_i = 0
			i_a = 0	i_i = 100
2	net2	a dan u	a_a = 92	a_u = 8
			u_a = 18	u_u = 94
3	net3	a dan e	a_a = 100	a_e = 0
			e_a = 0	e_e = 100
4	net4	a dan o	a_a = 100	a_o = 0
			o_a = 20	o_o = 80
5	net5	i dan u	i_i = 100	i_u = 0
			u_i = 0	u_u = 100
6	net6	i dan e	i_i = 72	i_e = 28
			e_i = 18	e_e = 82
7	net7	i dan o	i_i = 0	i_o = 100
			o_i = 0	o_o = 100
8	net8	u dan e	u_u = 100	u_e = 0
			e_u = 0	e_e = 100
9	net9	u dan o	u_u = 66	u_o = 34
			o_u = 36	o_o = 64
10	net10	e dan o	e_e = 100	e_o = 0
			o_e = 0	o_o = 100

Pada Tabel 3, variabel a\_a berarti vokal masukan /a/ yang dikenali sebagai vokal /a/, a\_i berarti vokal masukan /a/ yang dikenali sebagai vokal /i/, dan seterusnya. Selanjutnya pengenalan untuk vokal /a/ dihitung dengan rata-rata nilai variabel a\_a dihitung dengan

$$\text{Rata-rata pengenalan vokal /a/} = \frac{100 + 92 + 100 + 100}{4} = 99.5$$

Tabel 4. Statistik pengenalan untuk orde 8, panjang *frame* 10ms dan *epoch* 500

Vokal	/a/	/i/	/u/	/e/	/o/
/a/	<b>99.5</b>	0	0	0	0.5
/i/	1	<b>72.5</b>	0	1.5	25
/u/	0.5	0	<b>92.5</b>	0	7
/e/	0	2	0	<b>98</b>	0
/o/	8.5	0	7.5	0	<b>84</b>

Total persentase pengenalan dihitung dengan :

$$\begin{aligned} \% \text{ pengenalan} &= \frac{a_a + i_i + u_u + e_e + o_o}{5} \\ &= \frac{99.5 + 72.5 + 92.5 + 98 + 84}{5} \\ &= 89.3\% \end{aligned}$$

Pengujian untuk setiap jaringan dilakukan 3 kali. Rata-rata hasil pengenalan dari ketiga pengujian ditunjukkan pada Tabel 5.

Tabel 5. Rata-rata persentase pengenalan untuk data latihan dan data baru

Orde LPC	Epoch Panjang Frame	DATA LATIHAN			DATA BARU		
		500	1000	2000	500	1000	2000
8	10ms	90.1	89.6	89.5	<b>86.9</b>	85.5	85.3
	20ms	89.5	88.8	89.4	84.7	86	85.9
10	10ms	90.8	<b>91.6</b>	90.5	85	85.1	84.7
	20ms	91.5	91	90.7	85.4	84.4	84.7
14	10ms	90.4	90.4	90.4	84	84.9	84.8
	20ms	90.6	90.6	90.4	85	84.6	84.5
16	10ms	90.8	91	89.4	84.9	84.9	83.7
	20ms	90.9	90.8	90.3	85.1	82.9	84.3

Menurut hasil yang ditunjukkan Tabel 5, maka pengenalan terbaik untuk data latihan diperoleh untuk orde LPC 10, panjang *frame* 10ms, dan *epoch* 1000. Sedangkan untuk data baru pengenalan terbaik diperoleh pada orde LPC 8, panjang *frame* 10ms dan *epoch* 500. Perbedaan antara hasil pengenalan data latihan dan data baru disebabkan data baru memiliki karakteristik sinyal yang berbeda dengan data latihan sehingga akan memberikan koefisien LPC yang berbeda pula. Akibatnya jaringan tidak dapat mengklasifikasikan vokal masukan dengan tepat. Masalah ini dapat diatasi dengan menambah data yang dilatihkan pada jaringan sehingga referensi ciri vokal yang dimiliki jaringan bertambah banyak.

Jaringan atau net dengan nilai parameter yang memberikan persentase pengenalan tertinggi, yaitu jaringan dengan orde LPC 10, panjang *frame* 10ms, dan *epoch* 1000, kemudian digunakan untuk simulasi pengenalan data tunggal. Masing-masing responden baik pada data latihan maupun data baru memiliki 5 buah *file* suara untuk tiap vokal, sehingga hasil pengenalan data tunggal terhadap responden untuk masing-masing vokal dapat dihitung dengan

$$\frac{\text{jumlah pengenalan terhadap vokal tertentu (/a/, /i/, /u/, /e/, /o/)}}{5} \times 100\%$$

Tingkat pengenalan total untuk masing-masing vokal pada data latihan dapat dihitung dengan persamaan berikut :

$$\frac{\text{hasil pengenalan untuk masing - masing vokal}}{\text{jumlah responden}} \times 100\%$$

Dengan menggunakan persamaan di atas, maka hasil pengenalan total untuk masing-masing vokal pada data latihan ditunjukkan Tabel 6.

Tabel 6. Pengenalan untuk masing-masing vokal data latihan

Vokal	Pengenalan (%)
/a/	90
/i/	92
/u/	84
/e/	82
/o/	80

Tabel 7 menunjukkan hasil pengenalan total untuk masing-masing vokal pada data baru.

Tabel 7. Pengenalan untuk masing-masing vokal data baru

Vokal	Pengenalan (%)
/a/	92
/i/	96
/u/	50
/e/	48
/o/	26

Berdasarkan hasil analisa di atas, maka dirumuskan hal-hal yang perlu diperhatikan dalam pengenalan ucapan vokal dengan analisis LPC sebagai berikut :

- Orde LPC  
Analisis LPC akan menghasilkan suatu matriks koefisien LPC yang memiliki jumlah baris sesuai dengan sinyal suara dan jumlah kolom sesuai dengan orde LPC. Semakin besar orde LPC yang dipilih, maka semakin besar pula ukuran matriks koefisien yang dihasilkan. Orde LPC 10 memberikan hasil pengenalan yang tertinggi.
- Panjang *Frame*  
Panjang *frame* ditentukan untuk membagi-bagi sinyal dalam segmen pendek yang kuasi-stasioner. Semakin kecil panjang *frame* berarti semakin banyak jumlah *frame* untuk sinyal suara tersebut, dan akan memberikan hasil pengenalan yang lebih baik.
- Kualitas data latihan dan data baru  
Data latihan dan data baru yang diambil dalam kondisi ruang yang tidak ideal mengakibatkan banyak noise yang ikut terekam dalam *file* suara.
- Karakteristik suara manusia  
Setiap manusia memiliki warna suara dan cara pengucapan vokal yang berbeda-beda. Selain itu posisi dan bentuk dari artikulatornya pun tidak sama, karenanya pada analisis LPC yang didasarkan pada struktur formant, terjadi kesalahan pengenalan. Kesalahan yang umum terjadi adalah vokal /u/ dikenali sebagai /o/, vokal /e/ sebagai /i/, dan vokal /o/ sebagai /u/.
- Cara perekaman sinyal suara  
Perekaman sinyal suara yang tepat juga ikut mempengaruhi ketepatan pengenalan. Berdasarkan eksperimen, cara perekaman yang paling baik adalah : vokal diucapkan secara wajar dan tidak dibuat-buat, setelah menekan tombol untuk merekam, langsung memasukkan suara sebab jeda waktu vakum akan mengakibatkan kesalahan pengenalan, suara tidak terlalu keras dan tidak terlalu lemah.

#### 4.2 Perbandingan Metode Analisis LPC dengan Metode Transformasi Fourier Cepat dan Transformasi Wavelet Diskret

Prinsip dasar dari analisis LPC adalah menghasilkan struktur *formant* (*spectral envelope*) dari sinyal suara, yang mengacu pada sumber dan *vocal tract*. *Vocal tract* terdiri dari faring dan rongga mulut. Bentuk *spectral envelope* digolongkan berdasarkan

Tabel 8. Perbandingan pengenalan suara dengan metode FFT, DWT dan LPC

Dasar perbandingan	FFT (Fast Fourier Transform)	DWT (Discrete Wavelet Transform)	LPC (Linear Predictive Coding)
Parameter	Jumlah titik uji dan <i>epoch</i>	Wavelet induk, <i>level</i> dekomposisi dan <i>epoch</i>	Orde LPC, panjang <i>frame</i> dan <i>epoch</i>
Langkah pemrosesan	<ul style="list-style-type: none"> <li>Menentukan jumlah titik uji</li> <li>Transformasikan dalam kawasan frekuensi</li> </ul>	<ul style="list-style-type: none"> <li>Memilih wavelet induk dan <i>level</i> dekomposisi</li> <li>Menentukan struktur wavelet</li> <li>Menghitung koefisien aproksimasi</li> </ul>	<ul style="list-style-type: none"> <li>Menentukan orde LPC dan panjang <i>frame</i></li> <li>Menghitung koefisien LPC</li> </ul>
Jumlah variasi	9	27	24
Pengenalan terbaik	Data latih : 95% Data baru : 87,8%	Data latih : 75,4% Data baru : 51,9%	Data latih : 91,6% Data baru : 86,9%
Pengenalan data tunggal	<ul style="list-style-type: none"> <li>Data latih /a/ : 98% /i/ : 74% /u/ : 80% /e/ : 66% /o/ : 20%</li> <li>Data baru /a/ : 92% /i/ : 74% /u/ : 38% /e/ : 52% /o/ : 20%</li> </ul>	<ul style="list-style-type: none"> <li>Data latih /a/ : 44% /i/ : 18% /u/ : 56% /e/ : 68% /o/ : 64%</li> <li>Data baru /a/ : 24% /i/ : 0% /u/ : 4% /e/ : 44% /o/ : 38%</li> </ul>	<ul style="list-style-type: none"> <li>Data latih /a/ : 90% /i/ : 92% /u/ : 84% /e/ : 82% /o/ : 80%</li> <li>Data baru /a/ : 92% /i/ : 96% /u/ : 50% /e/ : 48% /o/ : 26%</li> </ul>

sejumlah puncak yang disebut *formant*, sementara *formant* sendiri merupakan model resonansi dari *vocal tract*.

Transformasi Fourier Cepat pada dasarnya mengubah sinyal dari domain waktu menjadi domain frekuensi sehingga diperoleh spektrum gelombang (distribusi frekuensi) dari sinyal. Sedangkan Transformasi Wavelet Diskret merupakan representasi sinyal dalam ruang vektor yang berenergi terbatas, merupakan fungsi band-pass dan merupakan hasil translasi dan dilasi. Pada transformasi wavelet diskret digunakan skala dan posisi berdasarkan pangkat dua.

Dalam percobaan yang dilakukan sebelum Tugas Akhir ini dibuat, telah dicoba untuk melakukan pengenalan ucapan vokal dengan menggunakan Transformasi Fourier Cepat dan Transformasi Wavelet Diskret, karena itu dipandang perlu untuk melakukan perbandingan terhadap ketiga eksperimen tersebut. Hasil perbandingan ditampilkan dalam Tabel 8. Berdasarkan Tabel 8 maka dapat disimpulkan bahwa pengenalan ucapan vokal dengan Transformasi Fourier memiliki persentase pengenalan yang paling tinggi untuk data dalam bentuk paket matriks, sedangkan analisis LPC memiliki persentase pengenalan yang paling baik dan merata untuk pengenalan data tunggal.

## V. PENUTUP

### 5.1 Kesimpulan

Kesimpulan yang dapat diambil dari pembuatan dan pengujian program simulasi pengenalan ucapan ini adalah sebagai berikut :

1. Analisis LPC dapat digunakan untuk proses pengenalan ucapan vokal yang berperan untuk memberikan struktur *formant* dari suatu sinyal suara.
2. Orde LPC yang digunakan dalam analisis LPC menentukan hasil pengenalan jaringan, dimana orde yang memberikan hasil terbaik adalah LPC-10.
3. Panjang *frame* yang lebih kecil (10ms) yang digunakan dalam analisis LPC akan memberikan hasil yang lebih baik dibandingkan panjang *frame* yang lebih besar (20ms).
4. Tingkat pengenalan data latihan lebih tinggi dibandingkan data baru.
5. Perbandingan hasil pengenalan dengan analisis LPC terhadap metode Transformasi Fourier Cepat dan Transformasi Wavelet Diskret menunjukkan bahwa Transformasi Fourier memiliki persentase pengenalan yang paling tinggi untuk data dalam bentuk paket matriks, sedangkan analisis LPC

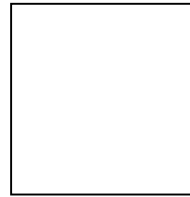
memiliki persentase pengenalan yang paling baik dan merata untuk pengenalan data tunggal.

## 5.2 Saran

1. Penelitian dapat dikembangkan hingga ke pengenalan suku kata, kata atau bahkan kalimat.
2. Analisis LPC dalam program ini hanya sampai pada tingkat koefisien LPC, untuk pengembangan lebih lanjut dapat dilakukan hingga penghitungan koefisien *cepstral*.
3. Penambahan data latihan dengan responden yang memiliki berbagai variasi warna suara untuk semakin melengkapi referensi jaringan.
4. Peningkatan arsitektur jaringan LVQ dengan penambahan jumlah *hidden neuron*, laju pembelajaran dan jumlah *epoch*.

## DAFTAR PUSTAKA

1. Bradbury, Jeremy, *Linear Predictive Coding*, paper, 2002
2. C. De Stefano, C. Sansone and M. Vento, *Comparing Generalization and Recognition Capability of Learning Vector Quantization and Multi-Layer Perceptron Architectures*, University of Naples, Napoli.
3. Christman, Michael, *Spoken Word Recognition Using Cepstral Analysis*, Virginia Tech, 2001.
4. Gold, Ben, and Nelson Morgan, *Speech and Audio Signal Processing : Processing and Perception of Speech and Music*, John Wiley & Sons, Inc., New York, 1999.
5. Hanselman, Duane, and Bruce Littlefield, *MATLAB : Bahasa Komputasi Teknis*, Penerbit ANDI, Yogyakarta, 2000.
6. Himawan, Roziq, *Teknologi Analisa Data : Aplikasinya pada Pengujian Tidak Merusak dengan Metode Ultrasonic*, IECI Japan Refreshing Seminar, 2001.
7. Ikuta, Hisatoshi, Yoshihisa Ishida, Takashi Honda, and Yasuhiko Arai, *Spoken Word Recognition Using LVQ Based on DP Matching*, Meiji University, Yokohama.
8. Indrajaya, Yunes, *Pemampatan Isyarat Tutur Menggunakan Penyandian Prediktif Linear*, Jurusan Teknik Elektro Universitas Diponegoro, Tugas Akhir, 2002.
9. May, Ignatius Leo, *Pengenalan Vokal Bahasa Indonesia dengan Jaringan Saraf Tiruan melalui Transformasi Wavelet Diskret*, Jurusan Teknik Elektro Universitas Diponegoro, Tugas Akhir, 2002.
10. Proakis, John. G, and Dimitris G. Manolakis, *Pemrosesan Sinyal Digital : Prinsip-prinsip, Algoritma, dan Aplikasi*, PT Prenhallindo, Jakarta, 1995.
11. Rabiner, Lawrence and Biing-Hwang Juang, *Fundamentals of Speech Recognition*, PTR Prentice-Hall Inc., New Jersey, 1993.
12. Sitanggang, Doni, *Pengenalan Vokal Bahasa Indonesia dengan Jaringan Saraf Tiruan melalui Transformasi Fourier*, Jurusan Teknik Elektro Universitas Diponegoro, Tugas Akhir, 2002.
13. Spanias, Andreas, *Matlab-Based Project on The Linear Predictive Coding Algorithm Used in Cellular Phones*, 2002.
14. The Connexions Project, *Speech Processing : Theory of LPC Analysis and Synthesis*, Creative Common Attribution, 2002.



**Ivana – L 2F0 98 632**, lahir di Medan, 3 September 1980. Saat ini sedang menjalani proses untuk menyelesaikan studi Strata 1 di Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang.

Mengetahui/Mengesahkan,

Pembimbing I

Pembimbing II

**Wahyudi, ST, MT**  
NIP. 132 086 662

**Achmad Hidayatno, ST, MT**  
NIP. 132 137 933