



UNIVERSITAS INDONESIA

**ANALISA DAN IDENTIFIKASI BERBAGAI PENYAKIT
PARU-PARU DENGAN METODE HIDDEN MARKOV
MODEL**

SKRIPSI

EVI ANDRIANI
0706199281

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
DESEMBER 2009**



UNIVERSITAS INDONESIA

**ANALISA DAN IDENTIFIKASI BERBAGAI PENYAKIT
PARU-PARU DENGAN METODE HIDDEN MARKOV
MODEL**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

**EVI ANDRIANI
0706199281**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
DESEMBER 2009**

HALAMAN PERNYATAAN ORISINALITAS

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

Nama : Evi Andriani
NPM : 0706199281
Tanda Tangan :
Tanggal : 2009

PENGESAHAN

Seminar ini diajukan oleh :

Nama : Evi Andriani
NPM : 0706199481
Program Studi : Teknik Elektro
Judul Skripsi : **ANALISA DAN IDENTIFIKASI BERBAGAI
PENYAKIT PARU-PARU DENGAN METODE
HIDDEN MARKOV MODEL**

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

DEWAN PENGUJI

Pembimbing : **Ir. Purnomo Sidi Priambodo M.Sc., Ph.D.** (_____)
Penguji : **Prof. Dr. Ir. Harry Sudibyo M.Sc** (_____)
Penguji : **Dr. Ir. Arman D.Diponegoro** (_____)

Ditetapkan di : Depok

Tanggal : 29 Desember 2009

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur penulis ucapkan kehadirat Allah SWT karena dengan rahmat, ridho, dan kasih sayang-Nya, penulisan skripsi ini bisa selesai tepat pada waktunya. Salawat dan salam selalu penulis haturkan kepada baginda Rasulullah Muhammad SAW, karena berkat jasa beliau kita dapat hidup di zaman yang terang benderang ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Elektro Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan seminar ini, sangatlah sulit bagi saya untuk menyelesaikan seminar ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Ir. Purnomo Sidi Priambodo M.Sc., Ph.D, selaku dosen pembimbing I telah menyediakan waktu, tenaga dan pikiran untuk mengarahkan dalam penyusunan skripsi ini;
2. Dr. Ir. Arman D. Diponegoro, selaku dosen pembimbing II telah menyediakan waktu, tenaga dan pikiran untuk mengarahkan dalam penyusunan skripsi ini;
3. dr. Untung Sentosa M. Kes, dr. Resno Sp.PD dan dr. Yana Sp.PD di RS. Al Islam Bandung telah menyediakan waktu, tenaga dan pikiran untuk mengarahkan dalam penyusunan skripsi ini.
4. Kedua Orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral; dan
5. Teman dan sahabat yang telah banyak membantu saya dalam menyelesaikan skripsi ini.

Akhir kata, saya berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu pengetahuan.

Depok, 29 Desember 2009

Penulis

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI

TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

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

Nama : Evi Andriani
NPM : 0706199281
Program Studi : 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 :

ANALISA DAN IDENTIFIKASI BERBAGAI PENYAKIT PARU-PARU DENGAN METODE *HIDDEN MARKOV MODEL*

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 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 : 29 Desember 2009
Yang menyatakan,

(Evi Andriani)

ABSTRAK

Nama : Evi Andriani
Program Studi : Elektro
Judul : Analisa dan Identifikasi Berbagai Penyakit Paru-paru dengan Metode Hidden Markov Model

Hidup di kota yang memiliki tingkat populasi dan polusi yang sangat tinggi akan berdampak negatif bagi kesehatan manusia, khususnya pada sistem pernafasan. Gangguan pada sistem pernafasan biasanya dapat terdeteksi melalui suara tarikan dan hembusan nafas dari penderita. Beberapa contoh gangguan tersebut adalah *bronchial*, *cracle* dan *pleurisy*. Skripsi ini membahas tentang perancangan sistem identifikasi penyakit pernafasan atau paru-paru dengan metode Hidden Markov Model (HMM). Sistem ini terbagi menjadi dua proses utama, yaitu pembentukan database dan pengenalan penyakit paru-paru. Kedua proses ini dilakukan dengan cara yang hampir sama, yaitu tiap sampel akan mengalami proses pelabelan, pembuatan codebook dan pembentukan parameter HMM. Hanya saja, pengolahan sinyal suara pada proses pengenalan mengacu database yang telah lebih dulu diproses. Dimulai dengan pembentukan vektor-vektor data dengan teknik kuantisasi vektor (VQ), yang kemudian dicari suatu nilai *centroid* yang presisi untuk dijadikan *state* HMM dalam menentukan nilai-nilai parameter yang dibutuhkan. Berdasarkan parameter-parameter inilah, dapat dihitung suatu nilai probabilitas (*Log of Probability*) maksimum yang akan menunjukkan hasil keluarannya. Dari hasil perancangan sistem ini, akan dibandingkan akurasi sistem terhadap variasi nilai durasi sampel, jumlah sampel, dan ukuran *codebook*. Pada penelitian ini, ukuran *codebook* yang optimal adalah 32, jumlah database yang optimal sebesar 10 (sepuluh) buah. Sementara persentase akurasi sistem secara keseluruhan bervariasi antara 70% hingga 93,33%.

Kata kunci:
Penyakit paru-paru, HMM, *centroid*, VQ, LoP, ukuran *codebook*, jumlah sampel.

ABSTRACT

Name : Evi Andriani
Study Program : Electrical Engineering
Title : Analysis and Identification of Various Lung Diseases by
Methode of Hidden Markov Model

Living in high poluted and populated city will give negative effects for our health especially for our respiratory system. The failure of respiratory system can be recognized by its sound during inhale and exhale phases called abnormal sound. It consist of bronchial, cracle, and pleural. This thesis discusses about lung disease recognition based on its abnormal sound using HMM method. The system consists of two main processes: database construction and diseases recognition. Both of this processes is done with almost exact ways. Each sample is processed through labelling, codebook construction, and HMM parameter construction. The difference is that in recognizing process, sound signal will be compared to database which has been made in prior. The whole process is started with data vectors production by using vector quantization (VQ) which can be used to analyze precisely centroid positions. The centroid will define HMM states and parameters. A Log of Probability (LoP) will be calculated from the parameter values. The largest value of LoP will be declared as an output of the system. Output of each samples are compared to obtain system accuracy based on variation of sample duration, sample amount, and codebook size. The optimum codebook size in this research is 32 and optimum sample amount in database is 10. Overall, accuracy of the system is variating from 70% up to 93,33%.

Key words:

Lung disease, HMM, centroid, VQ, LoP, codebook size, sample amount.

DAFTAR ISI

JUDUL	i
HALAMAN JUDUL	ii
HALAMAN PERNYATAAN ORISINALITAS	iii
HALAMAN PENGESAHAN	iv
KATA PENGANTAR.....	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN	vi
ABSTRAK.....	vii
ABSTRACT	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xii
DAFTAR SINGKATAN.....	xiii
I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Tujuan Penulisan.....	2
1.3 Batasan Masalah.....	2
1.4 Metoda Penulisan	2
1.5 Sistematika Penulisan.....	3
II DASAR TEORI.....	4
2.1 Sistem Pernafasan Paru-paru	4
2.1.1 Bunyi Pernafasan	5
2.1.2 Sistem Pernafasan Suara Paru-paru	6
2.2 Jenis Penyakit Paru-paru	7
2.2.1 Pneumonia	7
2.2.2 Bronchitis.....	8
2.2.3 Emphysema	10
2.2.4 Wheeze.....	12
2.2.5 Pleurisy	13
2.3 Voice Recognition	15
2.4 Proses-proses dalam Sistem Pengenalan Getaran Suara Paru-paru	15
2.4.1 Sampling Sinyal Analog	16
2.4.2 Ekstraksi.....	18
2.4.3 Kuantisasi Vektor.....	22
2.5 Hidden Markov Model (HMM).....	26
2.5.1 Tipe-tipe Hidden Markov Model.....	26
2.5.2 Elemen-elemen Hidden Markov Model.....	27
III PERANCANGAN SISTEM SOFTWARE.....	32
3.1 Blok Diagram Sistem dan Fungsinya.....	33
3.2 Pembuatan Database	35
3.2.1 Pelabelan.....	36

3.2.2	Pembuatan Codebook	38
3.2.3	Pembentukan Parameter HMM	41
3.3	Proses Pengenalan	43
IV UJI COBA DAN ANALISIS SISTEM IDENTIFIKASI PENYAKIT PARU-PARU DENGAN METODE HMM..... 48		
4.1	Daftar Penyakit dan Jenis Penyakit.....	48
4.2	Hasil Uji Coba.....	48
4.3	Presentasi Akurasi	50
4.4	Analisis Hasil Percobaan	51
4.4.1	Analisis Pengaruh Variasi Ukuran <i>Codebook</i>	51
4.4.2	Analisis Pengaruh Variasi Jumlah <i>Database</i>	55
4.4.3	Analisis Pengaruh Hubungan Ukuran <i>Codebook</i> dan Jumlah <i>Database</i>	56
V	KESIMPULAN	58
	DAFTAR ACUAN	59
	DAFTAR PUSTAKA	61

DAFTAR GAMBAR

Gambar	Halaman
2.1 Organ Sistem Pernafasan	4
2.2 Organ Sistem Paru-paru, Bronchi dan Bronchial Tree	5
2.3 Lokasi Paru-paru Normal	7
2.4 Tipe Asma	10
2.5 Peradangan saluran udara dan bronkokonstriksi di asma	13
2.6 Diagram pembuatan model referensi dan testing getaran suara paru	16
2.7 Sampling diskrit suatu sinyal	17
2.8 Sinyal hasil sampel data	17
2.9 Blok Diagram MFCC processor	19
2.10 Grafik mel-frekuensi	21
2.11 Mel-spaced filterbank	21
2.12 Codebook dari suatu input vektor	23
2.13 Diagram konsep pembentukan codebook dengan VQ	24
2.14 Diagram alir dari LBG algorithm	26
2.15 HMM dengan 4 state	27
3.1 Tampilan Menu setelah Current Directory	33
3.2 Blok Diagram Sistem pengenalan getaran suara penyakit paru-paru ...	34
3.3 Diagram alir pembuatan database untuk deteksi suara penyakit paru-paru	35
3.4 Tampilan program pembentukan label	37
3.5 Tampilan program pembentukan codebook	39
3.6 Tampilan software pembentukan parameter HMM	42
3.7 Diagram alir proses pengenalan penyakit paru-paru	44
3.8 Tampilan program pengenalan penyakit paru-paru	46
4.1 Codebook 32 bit	52
4.2 Codebook 64 bit	53
4.3 Codebook 128 bit	54

DAFTAR TABEL

Tabel	Halaman
4.1 Nama File dan Jenis Suara Abnormal Paru-Paru	48
4.2 Hasil Uji Coba Suara Bronchial.....	49
4.3 Hasil Uji Coba Suara Cracle.....	49
4.4 Hasil Uji Coba Suara Pleural.....	50
4.5 Presentasi akurasi semua penyakit.....	50
4.6 Presentasi akurasi semua parameter.....	50

DAFTAR SINGKATAN

DAFTAR ISTILAH

1. **Centroid** : sebuah titik yang mewakili beberapa titik-titik sample (codeword) dalam satu cluster.
2. **Cluster** : suatu ruang dua dimensi dari suatu bidang yang bentuknya tergantung dari reknik yang digunakan.
3. **Codebook** : kumpulan dari sejumlah codeword dari beberapa gelombang.
4. **Codeword** : titik-titik sample dari besaran komponen-komponen spektrum frekuensi dari suatu gelombang yang diperoleh dari hasil transformasi Fourier.
5. **FFT** : *Fast Fourier Transform*, teknik atau cara untuk mengkonversi suatu gelombang dari domain waktu ke domain frekuensi.
6. **Frame blocking** : Sinyal kontinu dibagi-bagi dalam frame.
7. **LoP** : *Log of Probability*, besarnya kemungkinan munculnya suatu pengamatan dari suatu sederetan munculnya suatu kejadian-kejadian tertentu (probability of transition) dengan memperhitungkan kemungkinan munculnya kejadian yang mengikuti sebelumnya.
8. **Mel frequency cepstrums coefficients** : didasarkan pada variasi batas *bandwidth* frekuensi pendengaran manusia. MFCC memfilter secara linear pada frekuensi rendah dan secara logaritmik pada frekuensi tinggi untuk menangkap karakteristik penting dari sinyal suara.
9. **Iterasi** adalah banyaknya pengulangan yang dilakukan dalam pengambilan sample suara gempa
10. **Vector quantization** : proses pemetaan vektor dari ruang vektor yang besar menjadi sebuah wilayah yang terbatas.
11. **Windowing** : akan meminimalisasikan sinyal yang telah *diframe-framekan* sehingga sinyal akan nol pada permulaan dan akhir masing-masing *frame*. Sinyal yang baru tersebut seperti diberi *fade in* dan *fade out*.

BAB I

PENDAHULUAN

1.1 Latar belakang

Paru-paru merupakan salah satu organ terpenting dalam tubuh manusia. Paru-paru adalah salah satu organ pada sistem pernapasan yang berfungsi sebagai tempat bertukarnya oksigen dari udara yang menggantikan karbondioksida di dalam darah. Jika organ ini mengalami kerusakan sekecil apapun pada salah satu bagiannya, maka tubuh manusia juga akan terkena dampak yang fatal akibat kerusakan tersebut.

Efek perkembangan teknologi, selain membawa dampak positif bagi manusia, juga membawa dampak negatif khususnya bagi kesehatan manusia. Salah satunya berdampak pada sistem pernafasan pada manusia. Tingkat polusi udara yang semakin tinggi, khususnya di Indonesia, pola hidup yang tidak teratur, konsumsi makanan yang sembarangan, dan jenis-jenis ketidakteraturan lainnya, memicu terganggunya sistem pernafasan manusia yang dalam jangka waktu panjang dapat memicu timbulnya penyakit yang cukup banyak jenisnya. Hal inilah yang mendorong dilakukannya penelitian terhadap identifikasi berbagai jenis penyakit paru-paru tersebut.

Perkembangan teknologi khususnya di bidang *recognition system* yang semakin cepat dan canggih telah merambah dunia medis dalam pengaplikasiannya. Pada umumnya, terdapat beberapa metode pengenalan suara yang biasa digunakan, seperti *fuzzy logic*, *neural network*, dan *Hidden Markov Model* (HMM). Teknik *fuzzy logic* merupakan teknik yang paling sederhana, namun hasil yang didapatkan kurang akurat dibanding metode lainnya. Sedangkan dengan menggunakan teknik *neural network* diperlukan proses pembelajaran dan iterasi yang sangat banyak dan panjang. Dengan teknik HMM, walaupun proses pembelajarannya lebih kompleks dibanding dengan dua metode lainnya, tetapi jumlah sampel yang dibutuhkan tidak perlu banyak dan hasil yang diperoleh jauh lebih optimal. Oleh karena itulah, dalam penelitian ini digunakan HMM sebagai metode pengenalan. ^[1]

Akhirnya, penelitian ini diharapkan dapat mempermudah para ahli di bidang kedokteran untuk mengidentifikasi berbagai jenis penyakit paru-paru melalui suara pernafasan atau suara paru-paru sebagai objek yang diidentifikasi. Bahkan penelitian ini dapat dikembangkan lebih lanjut dalam pembuatan alat pendeteksi penyakit paru-paru yang lebih sederhana dan ekonomis sehingga dapat diaplikasikan langsung oleh masyarakat awam sekalipun.

1.2 Tujuan Penulisan

Tujuan penulisan skripsi ini adalah menganalisa perangkat lunak pengidentifikasi penyakit atau suara abnormal paru-paru dengan metode Hidden Markov Model (HMM). Selain itu, skripsi ini ditujukan kedepannya untuk membandingkan akurasi perangkat lunak sistem pendeteksi penyakit paru-paru terhadap variasi jumlah sampel dan ukuran *codebook*.

1.3 Batasan Masalah

Pada penelitian ini, program bersifat *non-real time*. Data yang akan dikenali diolah terlebih dahulu dengan menggunakan *software* pengolah gelombang suara yang kemudian diproses program untuk diidentifikasi. Penyakit atau suara abnormal yang akan dideteksi hanya dibatasi pada penyakit atau suara abnormal paru-paru yang secara langsung dapat dikenali melalui suaranya. Adapun penyakit atau suara abnormal yang dimaksud adalah *cracle*, *bronchial*, dan *pleurisy*. Masing-masing penyakit terdiri dari 10 (sepuluh) buah *database* dan 10 buah sampel uji yang diambil secara tidak langsung dalam format digital. Sedangkan variasi ukuran *codebook* yang digunakan adalah 32, 64, dan 128.

1.4 Metode Penulisan

Metode-metode yang digunakan dalam penulisan skripsi ini antara lain:

(1) Studi kepustakaan

Mempelajari semua informasi mengenai paru-paru dan kelainannya serta mengenai proses pengenalan suara dan HMM dari buku, jurnal, artikel dan literatur lain.

(2) Pencarian data

Mencari data-data yang diperlukan, dalam hal ini berupa sampel suara paru-paru dalam bentuk digital yang diperoleh dari institusi-institusi terkait.

1.5 Sistematika Penulisan

Pada penulisan skripsi ini terdiri dari 5 (lima) bab, dimana masing-masing bab mempunyai kaitan satu sama lain yaitu :

BAB I PENDAHULUAN

Memberikan latar belakang tentang permasalahan, tujuan penulisan dan batasan masalah yang dibahas dalam tugas akhir ini.

BAB II DASAR TEORI

Memberikan tinjauan pustaka yang berkaitan dengan sistem dan bunyi pernafasan paru-paru, jenis-jenis penyakit paru-paru, auskultasi. Membahas teori dasar yang menunjang perancangan sistem termasuk diantaranya dasar-dasar pemrosesan sinyal khususnya dengan Teknik *Hidden Markov Model* (HMM).

BAB III MEMODIFIKASI SISTEM *SOFTWARE*

Membahas perancangan sistem *software* yang dibuat berisi flowchart tentang jalannya program.

BAB IV PENGUJIAN DAN ANALISIS

Berisi data-data pengamatan pengujian pada bagian-bagian tertentu dari keseluruhan rangkaian, serta pembahasan atau analisa data hasil pengujian, dengan melakukan perbandingan terhadap teori yang mendukung.

BAB V KESIMPULAN

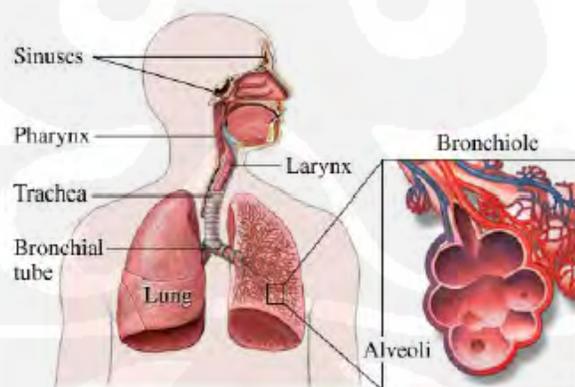
Berisikan kesimpulan dari dasar-dasar sistem, perancangan sistem dan pengujian.

BAB II DASAR TEORI

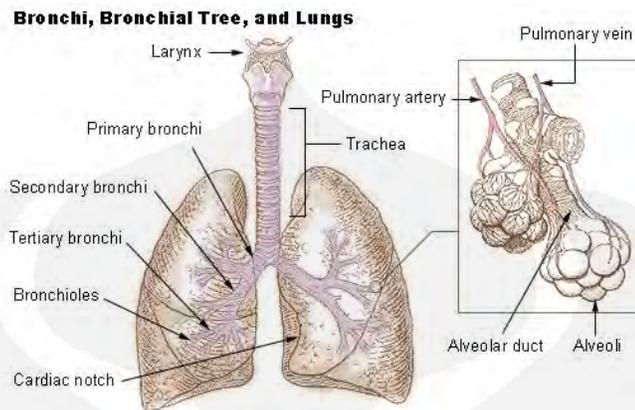
2.1 Sistem Pernafasan Paru-Paru

Paru-paru adalah salah satu organ pada sistem pernafasan yang berfungsi sebagai tempat bertukarnya oksigen dari udara yang menggantikan karbondioksida di dalam darah. Proses ini dinamakan sebagai respirasi dengan menggunakan bantuan haemoglobin sebagai pengikat oksigen. Setelah O_2 didalam darah diikat oleh haemoglobin, selanjutnya dialirkan ke seluruh tubuh. Dalam tubuh manusia O_2 digunakan sel-sel tubuh dalam proses pelepasan energi. Proses tersebut selain menghasilkan energi juga menghasilkan CO_2 yang harus dikeluarkan dari tubuh.^[2]

Sistem pernafasan berfungsi untuk menyediakan suplai O_2 dan mengeluarkan CO_2 dari dalam tubuh. proses pertukaran O_2 dan CO_2 terjadi pada saat manusia bernafas. Organ-organ yang menjadi bagian sistem pernafasan dapat dilihat pada gambar 2.1



Gambar 2.1 Organ Sistem Pernafasan^[3]



Gambar 2.2 Organ Sistem Paru-paru, Bronchi dan Bronchial Tree ^[3]

Pada sistem pernafasan, udara dari luar yang banyak mengandung O_2 masuk melalui hidung karena tekanan udara di paru-paru lebih rendah daripada tekanan lingkungan luar tubuh. Dari hidung udara masuk ke *trachea* lalu menuju paru-paru melalui *bronchial tube* dan *bronchiole*. Pertukaran O_2 dan CO_2 antara darah dan udara terjadi di alveoli yang dikelilingi banyak pembuluh darah. Udara di paru-paru yang kini banyak mengandung CO_2 dikeluarkan melalui organ pernafasan akibat tekanan paru-paru berubah menjadi lebih tinggi dibandingkan lingkungan luar tubuh. Perubahan tekanan pada paru-paru diatur oleh otot diafragma, otot dada dan tulang disekitar dada.

2.1.1 Bunyi Pernafasan

Aliran udara yang terjadi pada proses pernafasan di dalam paru-paru menghasilkan suara yang dapat digunakan untuk mengetahui kondisi paru-paru.

Secara umum, suara pernafasan dapat dibagi menjadi 3 jenis, yaitu :

- 1) Bunyi Nafas (*Breath Sounds*), adalah bunyi yang bangkit dari bernafas, di luar bunyi *adventitious*, yang didengar atau direkam pada dinding dada, *trachea*, atau mulut.
- 2) Bunyi Adventitious (*Adventitious Sounds*), adalah bunyi singkat yang muncul pada saat bernafas, baik kontinyu maupun tidak kontinyu, yang menandakan bahwa ada gangguan pada pulmonary. Contohnya :suara *crackles*, suara *squawks*, suara *wheezes*.

- 3) Bunyi Paru-Paru (*Lung Sounds*), adalah bunyi yang didengar atau terdeteksi di dinding dada atau dada, meliputi sebagian bunyi nafas dan sebagian bunyi *adventitious*.

2.1.2 Sistem Pernafasan Suara Paru-Paru

Sistem pernafasan dapat dikategorikan menjadi 2 saluran yaitu ^[4]:

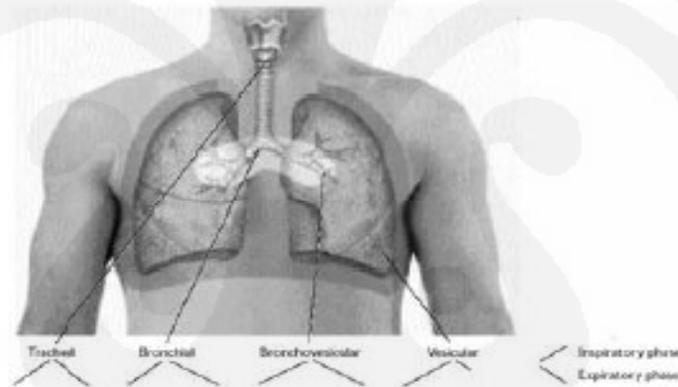
- 1) saluran atas
- 2) saluran bawah

Saluran pernafasan atas terdiri dari hidung, paranasal sinus, *pharynx*, dan *larynx*. Fungsi dari saluran ini adalah untuk menyaring, menghangatkan, dan melembabkan udara sebelum mencapai unit pertukaran gas. Saluran bawah pernafasan dimulai dari *trachea*, *bronchus* utama kanan yang terbagi menjadi 3 lobar atau bagian paru (atas, tengah dan bawah), *bronchus* kiri yang terbagi menjadi 2 lobar, *bronchioli*, dan berakhir di *alveoli*, dimana terjadi pertukaran gas.

Suara paru-paru terjadi karena adanya turbulensi udara saat udara memasuki saluran pernafasan selama proses pernafasan^[5]. Turbulensi ini terjadi karena udara mengalir dari saluran udara yang lebih lebar ke saluran udara yang lebih sempit atau sebaliknya. Pada saat inspirasi, udara mengalir dari saluran udara yang lebih luas ke saluran udara yang lebih sempit sehingga turbulensi yang terjadi lebih kuat sedangkan pada saat ekspirasi terjadi sebaliknya. Ini menyebabkan pada saat inspirasi suara yang terdengar lebih keras. Secara umum suara paru-paru dibagi menjadi 3 : suara normal, suara abnormal dan suara tambahan. Suara-suara tersebut dibagi dalam beberapa kategori berdasar *pitch*, intensitas, lokasi serta rasio inspirasi dan ekspirasi ^[5].

Suara paru-paru normal terbagi atas 4 kelompok *tracheal*, *bronchial*, *bronchovesikular* dan *vesikular*^[4]. Suara pernafasan *tracheal* sangat nyaring dan *pitch*-nya relatif tinggi. Inspirasi dan ekspirasi relatif sama panjang. Suara ini dapat didengar di atas *trachea* yang agak jarang dilakukan pada pemeriksaan rutin. Suara pernafasan vesikular merupakan suara pernafasan normal yang paling umum dan terdengar hampir di semua permukaan paru-paru. Suaranya lembut dan *pitch* rendah. Suara inspirasi lebih panjang dibanding suara ekspirasi. Suara

vesikular bisa terdengar lebih kasar dan sebagian terdengar lebih panjang apabila ada ventilasi yang cepat dan dalam (misal setelah berolah raga) atau pada anak-anak yang memiliki dinding dada yang lebih tipis. Suara vesikular juga bisa lebih lembut jika pasien lemah, tua, gemuk, atau sangat berotot. Suara *bronchial* sangat nyaring, *pitch* tinggi, dan suara terdengar dekat dengan stetoskop. Terdapat gap antara fasa inspirasi dan ekspirasi pada pernafasan, dan suara ekspirasi pada pernafasan, dan suara ekspirasi terdengar lebih lama dibanding suara inspirasi. Jika suara ini terdengar dimana-mana kecuali di manubrium, hal tersebut biasanya mengindikasikan terdapat daerah konsolidasi yang biasanya berisi udara tetapi berisi air. Terdapat suara pernafasan yang tingkat intensitas dan *pitch*-nya sedang. Inspirasi dan ekspirasinya sama panjang. Dengan suara bronchi, jika terdengar di mana-mana selain di batang utama *bronchus*, biasanya mengindikasikan daerah konsolidasi ^[5].



Gambar 2.3 Lokasi Paru-Paru Normal ^[4]

Selain itu terdapat juga suara paru-paru tambahan yang muncul karena adanya kelainan pada paru-paru yang disebabkan oleh penyakit.

2.2 Jenis Penyakit Paru-Paru

Ada 5 (lima) jenis penyakit paru-paru yang diuraikan berikut ini :

2.2.1 Pneumonia

Pneumonia adalah suatu infeksi dari satu atau dua paru-paru yang biasanya disebabkan oleh bakteri-bakteri, virus-virus, atau jamur. *Pneumonia* juga merupakan peradangan yang mengenai parenkim paru, distal dari

Universitas Indonesia

bronkiolus terminalis yang mencakup *bronkiolus respiratorius*, dan alveoli, serta menimbulkan konsolidasi jaringan paru dan gangguan pertukaran gas setempat^[5]. Orang-orang dengan *pneumonia* mungkin menjadi pendek napasnya. Satu-satunya serat-serat nyeri di paru adalah dipermukaan paru, di area yang dikenal sebagai ***pleura***. Nyeri dada mungkin berkembang jika aspek-aspek pleural bagian luar dari paru terlibat. Nyeri tajam dan memburuk ketika mengambil suatu napas yang dalam dikenal sebagai nyeri *pleuritic* juga dirujuk sebagai *pleurisy* yaitu peradangan dari lapisan sekeliling paru-paru (*pleura*). Untuk membuat diagnosis dari *pleurisy*, dokter memeriksa dada pada area nyeri dan seringkali dapat mendengar (dengan *stethoscope*) friksi (gesekan) yang dihasilkan oleh gosokan dari dua lapisan *pleura* yang meradang dengan setiap pernapasan dan bunyi ini diistilahkan sebagai ***pleural friction rub***.

- *Pneumonia Lobaris* ^[2] ialah peradangan jaringan paru akut yang berat yang disebabkan *pneumococcus*. Nama ini menunjukkan hanya satu lobus paru yang terkena.
- *Pneumonia Aspirasi* ^[6] merupakan proses terbawanya bahan yang ada di orofaring pada saat respirasi ke saluran bawah dan dapat merusak jaringan parenkim paru.
- *Pneumonia kronik* merupakan *pneumonia* yang berkembang dan berlangsung selama berminggu-minggu sampai berbulan-bulan.
- *Pneumoniathorax* adalah penumpukan dari udara yang bebas dalam dada diluar paru yang menyebabkan paru untuk mengempis. Pemeriksaan dada dengan sebuah *stethoscope* mengungkapakan suara-suara pernapasan yang berkurang atau tidak hadir diatas paru yang terpengaruh. Diagnosis dikonfirmasi oleh *x-ray* dada yaitu alat yang akurat dalam mendiagnosa jumlah-jumlah dari cairan yang terkumpul.

2.2.2 Bronchitis

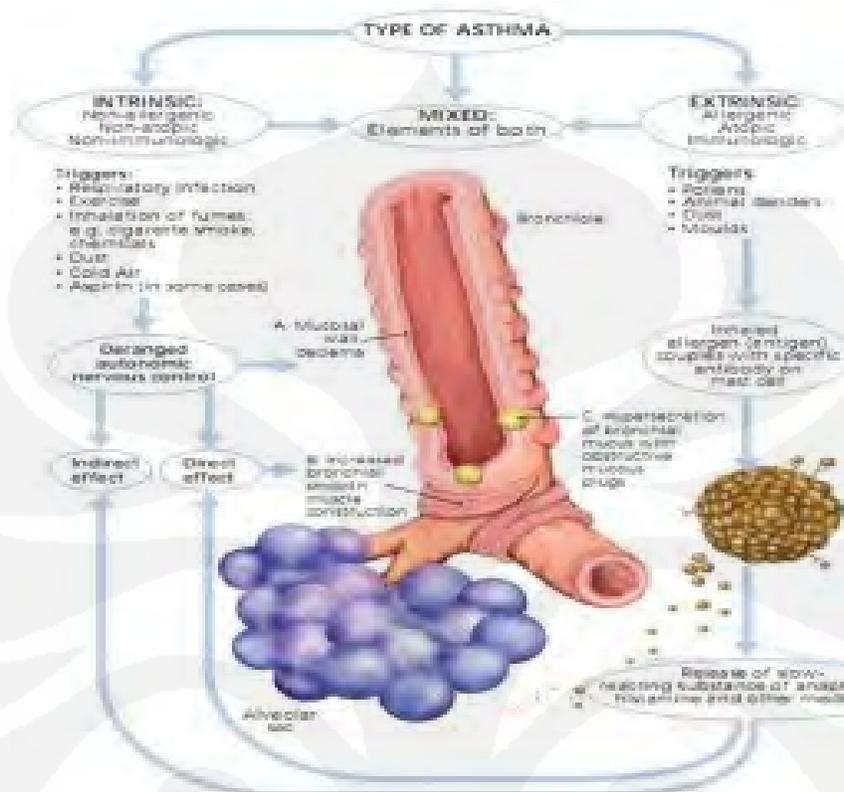
Bronchitis adalah penyakit pernapasan dimana selaput lendir pada saluran-saluran *bronchial* paru meradang. Ciri-cirinya adalah *chest cold* yang

berlangsung lebih dari dua minggu, demam yang lebih tinggi dari 102°F, batuk yang menghasilkan darah, sesak nafas atau menciut-ciut (*wheezing*). Orang-orang dengan **asma** juga mengalami peradangan lapisan dari tabung-tabung *bronchial* yang disebut *asthmatic bronchitis*. Dokter dapat melakukan tes pernapasan, yang disebut *spirometry*.

Asma adalah suatu gangguan yang kompleks dari bronkial yang dikarakteristikan oleh periode bronkospasme (kontraksi spasme yang lama pada jalan nafas). (Polaski : 1996). Asma adalah gangguan pada jalan nafas bronkial yang dikarakteristikan dengan bronkospasme yang reversibel. (Joyce M. Black : 1996). Asma adalah penyakit jalan nafas obstruktif intermiten, reversibel dimana trakea dan bronkhi berespon secara hiperaktif terhadap stimulasi tertentu. (Smelzer Suzanne : 2001).

Dari ketiga pendapat tersebut dapat diketahui bahwa asma adalah suatu penyakit gangguan jalan nafas obstruktif intermiten yang bersifat reversibel, ditandai dengan adanya periode bronkospasme, peningkatan respon trakea dan bronkus terhadap berbagai rangsangan yang menyebabkan penyempitan jalan nafas^[7].

Asma bronkhial adalah suatu penyakit dengan ciri meningkatnya respon bronkus terhadap berbagai rangsangan dengan manifestasi adanya penyempitan jalan nafas yang luas dan derajatnya dapat berubah-ubah baik secara spontan maupun hasil dari pengobatan (The American Thoracic Society). Asma bronkhial adalah adanya gangguan parasimpatis (hiperaktivitas saraf kolinergik), gangguan Simpatis (blok pada reseptor beta adrenergic dan hiperaktifitas reseptor alfa adrenergik)^[8].

Gambar 2.4 Tipe Asma^[8]

2.2.3 Emphysema^[9]

Emphysema adalah penyakit paru yang berjangka panjang dan progresif dan terjadi ketika dinding-dinding alveoli rusak/hancur bersama dengan pembuluh-pembuluh darah kapiler yang mengalir didalamnya. Ini mengurangi total area didalam paru dimana darah dan udara dapat bersentuhan, membatasi potensial untuk pemindahan oksigen dan karbondioksida. Cirinya adalah ketika pasien menghembuskan napas, *alveoli* dan saluran-saluran udara kecil mengempis (*collapse*).

Emphysema adalah penyakit paru yang berjangka panjang dan progresif dan terjadi ketika dinding-dinding alveoli rusak/hancur bersama dengan pembuluh-pembuluh darah kapiler yang mengalir didalamnya. Ini mengurangi total area didalam paru dimana darah dan udara dapat bersentuhan, membatasi potensial untuk pemindahan oksigen dan karbon dioksida.

Pada *emphysema* yang dini, ada peradangan dari saluran-saluran udara kecil atau *bronchioles* yang berhubungan yang membatasi jumlah udara yang dapat mengalir ke alveoli. Pada *emphysema* yang lebih parah, ada juga kehilangan elastisitas pada dinding-dinding alveoli yang masih belum hancur. Ketika pasien menghembuskan napas, alveoli dan saluran-saluran udara kecil mengempis (*collapse*). Ini membuat udara lebih sulit untuk keluar dari paru-paru dan membuat udara baru bahkan lebih sulit untuk masuk.

Ketika lebih banyak paru yang rusak dan paru tidak dapat memelihara konsentrasi-konsentrasi oksigen dalam aliran darah, tubuh mengkompensasinya dengan secara berangsur-angsur meningkatkan kecepatan bernapas. Setelah sesaat, bahkan *hyperventilation* (*hyper* = lebih banyak + *ventilation* = bernapas) tidak dapat memelihara tingkat-tingkat oksigen yang memadai, dan arteri-arteri dalam paru mulai mengerut atau menyempit. Paru-paru harus bekerja lebih keras untuk mendorong darah ke dalam pembuluh-pembuluh darah yang menyempit ini, menyebabkan tekanan darah dalam arteri-arteri paru meningkat (*pulmonary hypertension*). Dari waktu ke waktu, kebutuhan kerja ekstra menyebabkan otot paru-paru membesar (*hypertrophy*) dan dapat menyebabkan gagal paru-paru.

Beberapa penyebab dari *emphysema* antara lain sebagai berikut :

- a) Penyebab utama dari *emphysema* adalah merokok, yang mengaktifkan sel-sel yang meradang dalam paru. Peradangan ini menyebabkan; 1) pembengkakan didalam *bronchioles*, dan 2) aktivasi dari enzim-enzim yang disebut *proteases* yang menyerang dan merusak jaringan paru (struktur-struktur dinding alveoli).
- b) Ada kecenderungan genetik pada *emphysema*. Kondisi yang relatif jarang yang dikenal sebagai kekurangan *alpha 1-antitrypsin* adalah

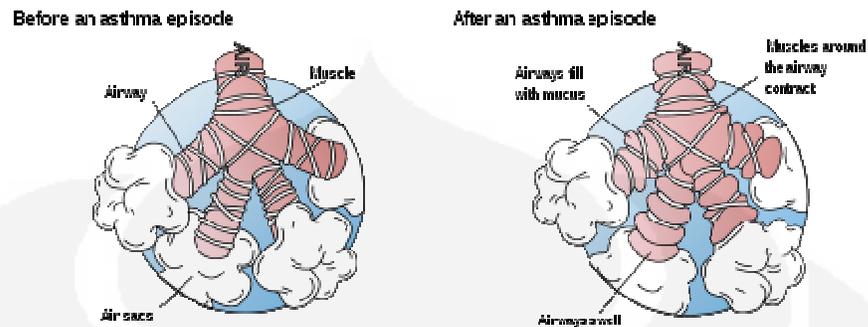
kekurangan genetik dari kimia yang melindungi paru dari kerusakan oleh proteases.

- c) *Emphysema* adalah juga komponen dari penuaan (*aging*). Ketika paru-paru menua, sifat-sifat elastisnya berkurang, dan tegangan-tegangan yang berkembang dapat berakibat pada area-area yang kecil dari emphysema.
- d) Penggunaan obat intravena dimana beberapa dari *additive-additive* yang bukan obat seperti tajin jagung dapat beracun pada jaringan paru.
- e) Kekurangan-Kekurangan imun dimana infeksi-infeksi seperti *Pneumocystis jiroveci* dapat menyebabkan perubahan-perubahan peradangan dalam paru.
- f) Penyakit-penyakit jaringan penghubung (*Ehlers-Danlos Syndrome, Marfan syndrome*) dimana jaringan elastis yang abnormal dalam tubuh dapat menyebabkan kegagalan alveoli

Di negara-negara belum berkembang, penyebab umum dari emphysema adalah polusi udara dalam rumah. Pada populasi-populasi ini, adalah sangat umum untuk mempunyai kompor-kompor dalam rumah di dapurnya. Asap dari memasak berakibat pada kerusakan paru-paru.

2.2.4 Wheeze^[10]

Wheeze merupakan suara seperti peluit, kasar dan kontinyu yang dihasilkan pada saluran pernafasan selama proses bernafas. saat terjadinya wheeze, beberapa bagian cabang saluran pernafasan menyempit sehingga kecepatan aliran udara di dalam saluran pernafasan meningkat. *Wheezing* biasanya dialami oleh orang-orang yang mengidap penyakit paru-paru. Penyebab utama kambuhnya *wheezing* adalah serangan asma.



Gambar 2.5 Peradangan saluran udara dan bronkokonstriksi di asma. Saluran udara menyempit sebagai akibat dari respon peradangan menyebabkan mengi^[10]

Wheeze terjadi pada bagian-bagian yang berbeda pada siklus pernafasan yang bergantung pada kedudukan penghambat saluran udara. Besarnya ukuran wheeze tersebut (*Wheeze rate*) diperkirakan berhubungan dengan tingkat ukuran penghambat pada saluran udara. Penyakit *Bronchiolar* biasanya menyebabkan *wheezing* yang terjadi pada fase *expiratory* dari proses pernafasan, dimana dalam fase ini hanya mencai maksimum 50 % dari keadaan normalnya. Wheezing juga dapat didengar pada fase *inspiratory*, yaitu dalam keadaan *stiff stenosis* (yang biasanya disebabkan oleh tumor) dan *hypersensitivity pneumonitis*.

2.2.5 Pleurisy^[11]

Pleurisy adalah peradangan dari lapisan sekeliling paru-paru (pleura). Ada dua pleura: satu yang melindungi paru (diistilahkan visceral pleura) dan yang lain melindungi dinding bagian dalam dari dada (parietal pleura). Dua lapisan-lapisan ini dilumasi oleh cairan pleural.

Pleurisy seringkali dihubungkan dengan akumulasi dari cairan ekstra dalam ruang antara dua lapisan dari pleura. Cairan ini dirujuk sebagai *pleural effusion*. Pleurisy juga dirujuk sebagai pleuritis.

Serat-serat nyeri dari paru berlokasi pada pleura. Ketika jaringan ini meradang, itu berakibat pada nyeri yang tajam pada dada yang memburuk

Universitas Indonesia

dengan napas, atau pleurisy. Gejala-gejala lain dari pleurisy dapat termasuk batuk, kepekaan dada, dan sesak napas.

Pleura tersusun dari dua lapisan dari jaringan lapisan yang tipis. Lapisan yang melindungi paru (visceral pleura) dan parietal pleura yang melindungi dinding dalam dari dada dilumasi oleh cairan pleural. Normalnya, disana ada kira-kira 10-20 ml cairan yang bening yang bekerja sebagai pelumas antara lapisan-lapisan ini. Cairan ini secara terus menerus diserap dan digantikan, terutama melalui lapisan bagian luar dari pleura. Tekanan didalam pleura adalah negatif (seperti dalam penghisapan) dan menjadi bahkan lebih negatif selama penghisapan (bernapas masuk). Tekanan menjadi kurang negatif selama penghembusan (bernapas keluar). Oleh karenanya, ruang diantara dua lapisan dari pleura selalu mempunyai tekanan negatif. Introduksi dari udara (tekanan positif) kedalam ruang (seperti dari luka pisau) akan berakibat pada mengempisnya paru.

Gejala yang paling umum dari pleurisy adalah nyeri yang umumnya diperburuk oleh penghisapan (menarik napas), Nyeri pada dada yang diperburuk oleh bernapas, dan Sesak Napas . Meskipun paru-paru sendiri tidak mengandung syaraf-syaraf nyeri apa saja, pleura mengandung berlimpah-limpah ujung-ujung syaraf. Ketika cairan ekstra berakumulasi dalam ruang antara lapisan-lapisan dari pleura, nyeri biasanya dalam bentuk pleurisy yang kurang parah. Dengan jumlah-jumlah akumulasi cairan yang sangat besar, ekspansi dari paru-paru dapat dibatasi, dan sesak napas dapat memburuk.

Pleurisy dapat disebabkan oleh apa saja dari kondisi-kondisi berikut:

- Infeksi-Infeksi: bakteri-bakteri (termasuk yang menyebabkan tuberculosis), jamur-jamur, parasit-parasit, atau virus-virus
- Kimia-Kimia Yang Terhisap Atau Senyawa-Senyawa Beracun: paparan pada beberapa agen-agen pembersih seperti ammonia
- Penyakit-Penyakit Vaskular Kolagen: lupus, rheumatoid arthritis

Universitas Indonesia

- Kanker-Kanker: contohnya, penyebaran dari kanker paru atau kanker payudara ke pleura
- Tumor-Tumor Dari Pleura: mesothelioma atau sarcoma
- Kemacetan: gagal paru-paru
- *Pulmonary embolism*: bekuan darah didalam pembuluh-pembuluh darah ke paru-paru. Bekuan-bekuan ini adakalanya dengan parah mengurangi darah dan oksigen ke bagian-bagian dari paru dan dapat berakibat pada kematian pada bagian itu dari jaringan paru (diistilahkan *lung infarction*). Ini juga dapat menyebabkan pleurisy.
- Rintangan dari Kanal-Kanal Limfa: sebagai akibat dari tumor-tumor paru yang berlokasi secara central
- Trauma: patah-patahan rusuk atau iritasi dari tabung-tabung dada yang digunakan untuk mengalirkan udara atau cairan dari rongga pleural pada dada
- Obat-Obat Tertentu: obat-obat yang dapat menyebabkan sindrom-sindrom seperti lupus (seperti Hydralazine, Procan, Dilantin, dan lain-lainnya)
- Proses-proses Perut: seperti pankreatitis, sirosis hati
- Lung infarction: kematian jaringan paru yang disebabkan oleh kekurangan oksigen dari suplai darah yang buruk

2.3 Voice Recognition

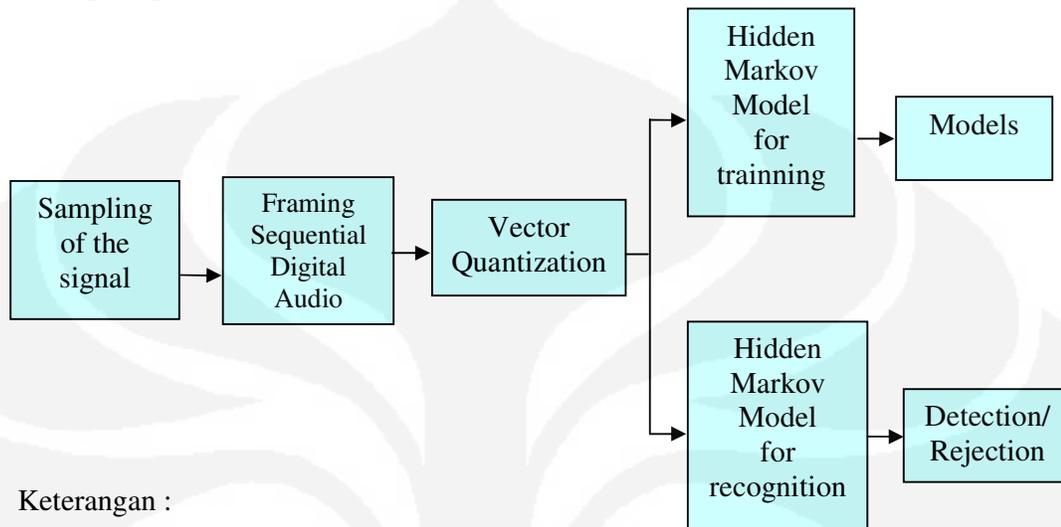
Voice recognition merupakan proses pengenalan secara otomatis suatu sinyal suara dengan membandingkan pola karakteristiknya dengan sinyal suara yang menjadi referensi atau acuan.

2.4 Proses-proses Dalam Sistem Pengenalan Getaran Suara Paru-Paru

Sistem pengenalan ini digunakan untuk mengenali jenis getaran dari suara paru-paru. Sistem ini dapat mengenali suara paru-paru dengan terlebih dahulu melakukan karakterisasi sinyal akustik yang akan dijadikan sebagai referensi. Karakterisasi getaran ini dilakukan dengan ekstraksi fitur sehingga dihasilkan vektor-vektor data yang akan digunakan pada pembuatan *codebook*. Gambar 2.7

Universitas Indonesia

menunjukkan proses-proses yang digunakan dalam sistem pengenalan getaran suara paru-paru.



Keterangan :

Proses Ekstraksi :

framing, FFT, windowing, Vektor Quantization (VQ), cluster, codebook, state

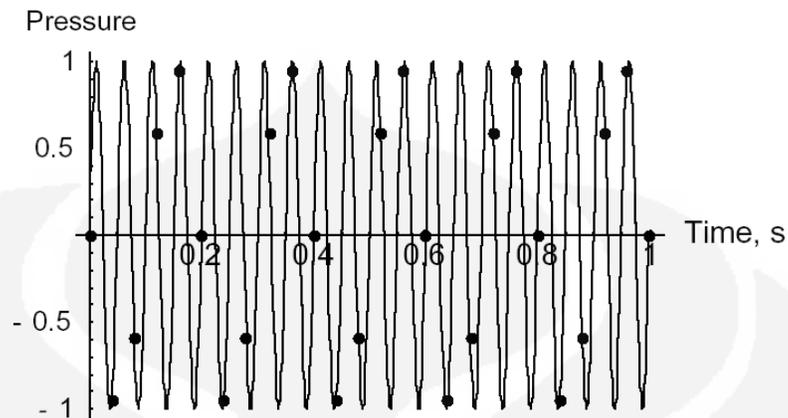
Gambar 2.6 Diagram pembuatan model referensi dan testing getaran suara paru-paru

2.4.1 Sampling Sinyal Analog

Sampling adalah suatu proses untuk membagi-bagi suatu sinyal kontinu dalam interval waktu yang telah ditentukan. Sampling ini dilakukan dengan mengubah sinyal analog menjadi sinyal digital dalam fungsi waktu. Pengubahan bentuk sinyal ini bertujuan untuk mempermudah memproses sinyal masukan yang berupa analog karena sinyal analog memiliki kepekaan terhadap noise yang rendah, sehingga sulit untuk memproses sinyal tersebut.

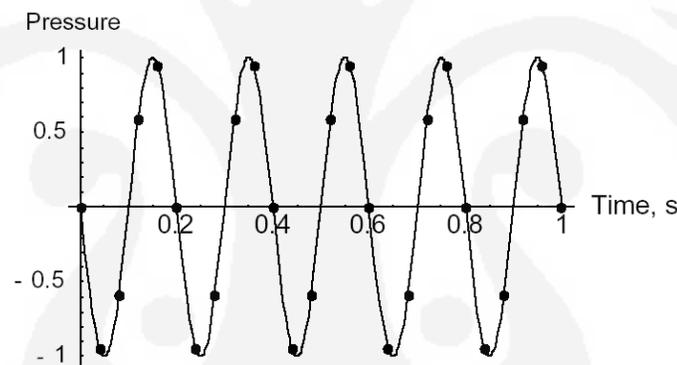
Proses *sampling* berbeda-beda untuk setiap suara. Bila *sampling* terhadap suatu sinyal suara tidak akurat maka dapat terjadi *misleading* atau hasil yang tidak sesuai dengan aslinya ^[12].

Aliasing merupakan hasil dari *sampling* secara diskrit pada suatu sinyal yang terlalu rendah sehingga memberikan resolusi yang rendah pula. Gambar 2.8 menunjukkan sampel sinyal 10 Hz yang nampak menjadi sinyal 5 Hz pada Gambar 2.8. Ini menunjukkan sebuah contoh terjadinya *aliasing*.



Gambar 2.7 Sampling diskrit suatu sinyal ^[13]

Titik-titik yang terlihat adalah sampel *point*



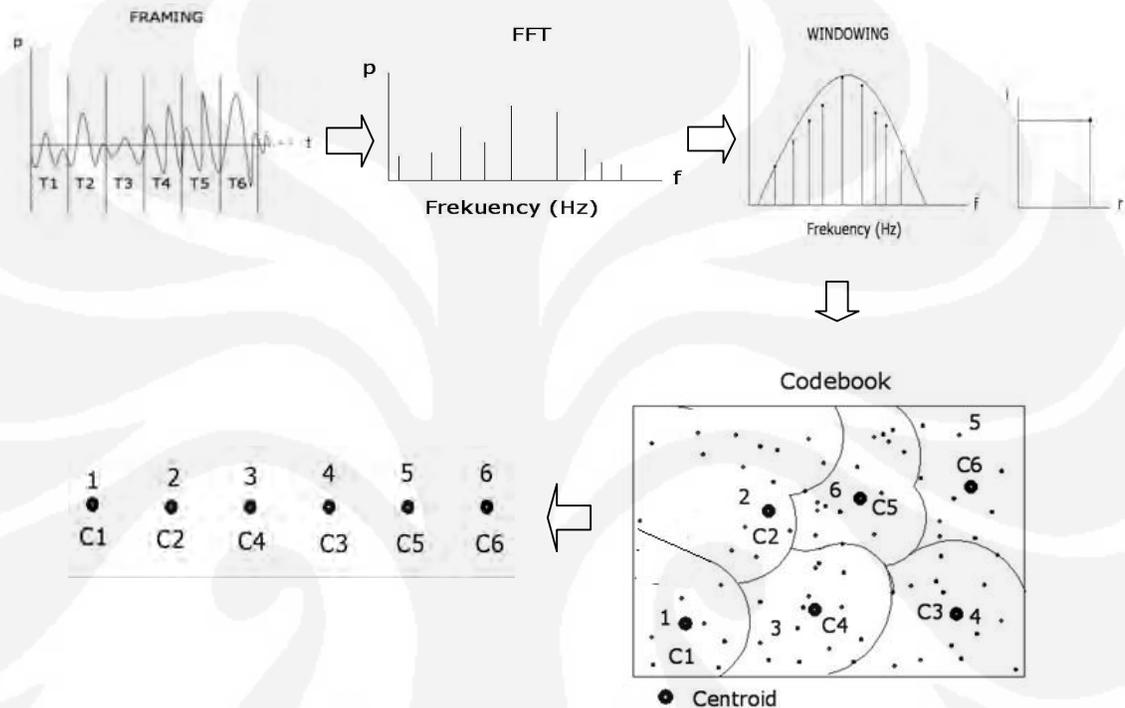
Gambar 2.8 Sinyal hasil sampel data ^[13]

Nyquist rate adalah rata-rata sampel minimum yang harus dipakai untuk mencegah timbulnya *aliasing* informasi frekuensi. Besarnya *Nyquist rate* harus dua kali frekuensi tertinggi dalam sinyal.

Dalam skripsi ini frekuensi sampling yang digunakan untuk merubah masukan menjadi sinyal digital adalah 8000 Hz disesuaikan dengan getaran suara paru-paru yang ketika pengambilan data menggunakan frekuensi sampling yang sama. Besarnya frekuensi sampling diatas jauh lebih tinggi dari dua kali frekuensi tertinggi dalam sinyal masukan.

2.4.2 Ekstraksi

Tujuan ekstraksi atau *feature extraction* ini adalah untuk mengubah bentuk getaran suara menjadi berbagai tipe parameter yang merepresentasikan sinyal suara untuk dianalisa.



Gambar 2.9 Feature Ectraxtion & Vector Quantization (VQ) dengan framing sebanyak 6

Dari gambar 2.9 di atas dapat dijelaskan bahwa sinyal-sinyal kontinyu suara paru-paru di bagi-bagi menjadi beberapa frame dalam interval waktu yang telah di tentukan sehingga diperoleh sampel poin (titik vektor). Setelah itu dilakukan windowing pada setiap frame yang tujuannya untuk membatasi. Dari proses ekstraksi akan menghasilkan titik-titik vektor dengan menggunakan *Fast Fourier Transform* (FFT) yang berfungsi mengubah masing-masing frame dari domain waktu ke domain frekuensi. Titik-titik itu kemudian dipetakan pada sebuah grafik dengan teknik kuantisasi vektor (VQ) dimana sumbu x merupakan komponen real dari masin-masing titik dan sumbu y merupakan komponen imajiner dari masing-masing titik. VQ merupakan proses pemetaan vektor dari

Universitas Indonesia

ruang vektor yang besar menjadi daerah yang terbatas (*cluster*). Setiap *cluster* ini diwakili oleh sebuah titik *centroid*. Satu kombinasi dari beberapa *centroid* ini merupakan perwakilan untuk satu sampel, yang disebut dengan *codeword*. Kumpulan semua *codeword* disebut dengan *codebook*. Titik-titik vektor tersebut dipetakan dan titik yang berdekatan di kuantisasikan kesatu vektor sehingga diperoleh beberapa titik vektor atau kumpulan *centroid*. Selanjutnya pemenggalan tersebut menjadi state yaitu merupakan salah satu parameter Hidden Markov Model (HMM).

Mel-Freq Cepstrum Coefficient (MFCC)

Metode yang digunakan untuk mendapatkan parameter yang dapat merepresentasikan sinyal suara ini adalah dengan *Linear Prediction Coding (LPC)*, *Mel-Frequency Cepstrum Coefficient (MFCC)* dan lainnya. Pada sistem pengenalan gelombang perubahan fase ini digunakan MFCC.

MFCC didasarkan pada variasi yang telah diketahui dari batas *bandwidth* pendengaran manusia dengan frekuensi. MFCC memfilter secara *linear* pada frekuensi rendah dan secara logaritmik pada frekuensi tinggi yang digunakan untuk menangkap karakteristik dari suatu sinyal suara. Ini diekspresikan dengan skala *Mel-frequency*. Pemetaan secara linear untuk frekuensi dibawah 1000 Hz dan logaritmik untuk frekuensi diatas 1000 Hz^[14]. Tujuan dari MFCC *processor* adalah untuk meniru karakteristik dari telinga manusia. Dibandingkan dengan bentuk getaran suara yang dikarakteristikan, hasil dari MFCC menunjukkan perbedaan yang kecil^[14].

Frame Blocking

Countinous speech diblok dalam *frame* dari N sampel, dengan *frame* yang berdekatan yang terpisah oleh M ($M < N$). *Frame* pertama terdiri dari N sampel pertama. *Frame* kedua mulai M sampel setelah *frame* pertama dan saling menyusul dengan $N - M$ sampel. Begitu pula dengan *frame* ketiga mulai dari sampel $2M$ setelah *frame* pertama (atau M sampel setelah *frame* kedua) dan menyusul $N - 2M$ sampel. Proses ini terus berlanjut sampai semua suara dihitung

dalam satu *frame* atau lebih. Harga khusus untuk N dan M adalah $N = 256$, yang mana sama dengan 30 ms *windowing*, sedangkan nilai $M = 100$ ^[14].

Windowing

Langkah selanjutnya adalah *windowing* masing-masing *individual frame* untuk meminimalisasikan diskontinuitas sinyal pada permulaan dan akhir dari masing-masing *frame*. *Windowing* ini untuk meminimalisasikan *spectral distortion* dengan menggunakan *window* untuk men-*taper* sinyal ke nol pada permulaan dan akhir masing-masing *frame*. Jika kita mendefinisikan *window* sebagai $w(n)$, $0 \leq n \leq N - 1$, dimana N adalah angka sampel pada masing-masing *frame*. Hasil *windowing* adalah signal yang dinyatakan dengan persamaan ^[14]

$$y_1(n) = x_1(n)w(n), \quad 0 \leq n \leq N - 1 \quad (2.1)$$

Pada program ini menggunakan *Hamming windowing* yang dinyatakan dalam persamaan

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \leq n \leq N - 1 \quad (2.2)$$

Fast Fourier Transform (FFT)

FFT mengubah masing-masing *frame* dari domain waktu ke domain frekuensi. FFT adalah *fast algorithm* untuk mengimplementasikan *Discrete Fourier Transform (DFT)* yang mana didefinisikan sebagai N sampel $\{x_n\}$, yaitu ^[7]

$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N} \quad (2.3)$$

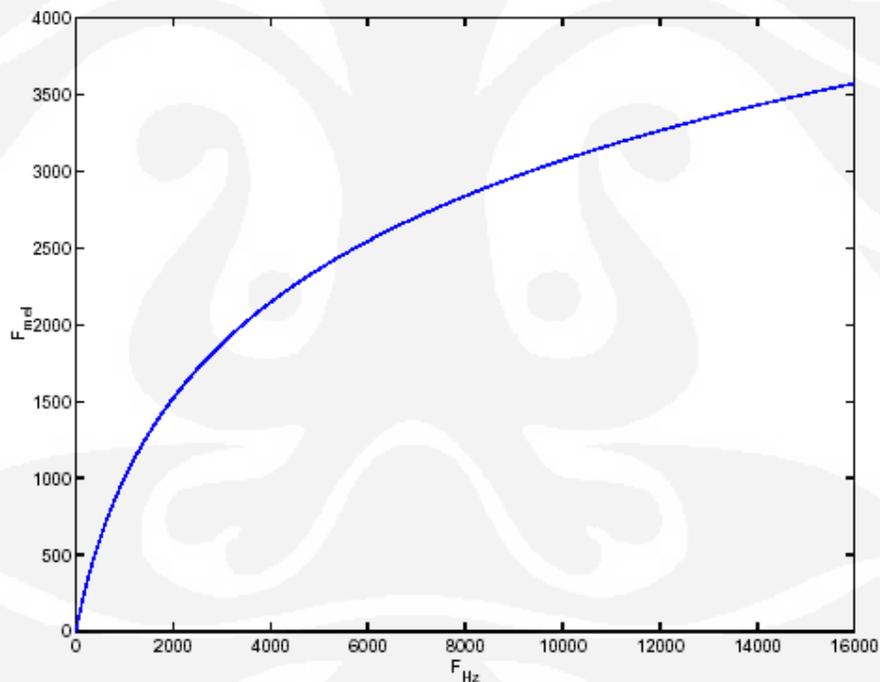
Hasil *sequence* $\{x_n\}$ diinterpretasikan sebagai berikut :

- frekuensi nol untuk harga $n = 0$
- frekuensi positif $0 < f < f_s/2$ untuk harga $1 \leq n \leq N/2 - 1$
- frekuensi negatif $-f_s/2 < f < 0$ untuk harga $N/2 + 1 \leq n \leq N - 1$

Mel-Freq Wrapping

Persepsi manusia terhadap isi frekuensi dari suatu suara tidak mengikuti skala linear. Untuk itu masing-masing *tone* dengan sebuah frekuensi tertentu diukur dengan skala Hz, sedangkan sebuah *subject pitch* diukur pada skala yang disebut *mel*. Skala mel-frekuensi adalah pemetaan frekuensi secara linear untuk frekuensi dibawah 1 kHz dan logaritmik untuk frekuensi diatas 1 kHz. Sebagai titik referensi, *pitch* dari 1 kHz, 40 dB diatas *perceptual hearing threshold*, didefinisikan sebagai 1000 *mels*. Oleh karena itu, dapat digunakan formula sebagai berikut untuk menghitung *mels* untuk frekuensi yang diberikan dalam Hz. Gambar 2.10 menunjukkan grafik skala mel-frekuensi.

$$mel(f) = 2595 * \log_{10}(1 + f / 700) \quad (2.4)$$

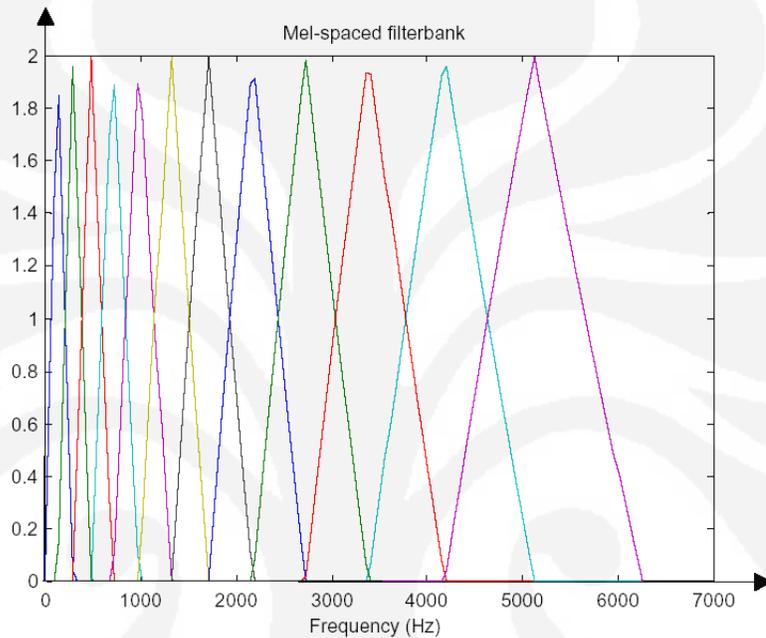


Gambar 2.10 Grafik mel-frekuensi versus frekuensi ^[8]

Gambar 2.11 merupakan *filterbank* yang diperoleh dengan menempatkan pusat frekuensi pada skala mel-frekuensi dan hasil pemetaan dikembalikan lagi sebagai lebar dari *triangular bandpass*. *Filter bank* yang diaplikasikan dalam

Universitas Indonesia

domain frekuensi menyederhanakan perhitungan untuk mengambil *triangle-shape window* pada *spectrum*. Keuntungan dari pemikiran *mel wrapping filter bank* adalah bahwa masing-masing *filter* dapat digambarkan sebagai sebuah *histogram* pada domain frekuensi.



Gambar 2.11 Mel-spaced filterbank ^[8]

Cepstrum

Ini adalah langkah terakhir dalam feature extraction, *log mel spectrum* diubah kembali ke waktu. Hasilnya disebut *mel frequensi cepstrum coefficient* (MFCC). Representatif *spectral* dari *speech spectrum* memberikan representatif yang baik untuk *local spectral properties* dari sinyal suara untuk analisa *frame* yang diberikan. Karena *mel spectrum coefficient* (dan logaritmiknya) adalah angka real, kita dapat mengubahnya ke time domain menggunakan *Discrete Cosine Transform* (DCT). Oleh karena itu *mel power spectrum coefficient* tersebut merupakan hasil dari langkah terakhir yang dinotasikan dengan \tilde{S}_k , dimana $k = 1, 2, \dots, K$, maka MFCC, \tilde{c}_n dapat dihitung dengan persamaan ^[8]

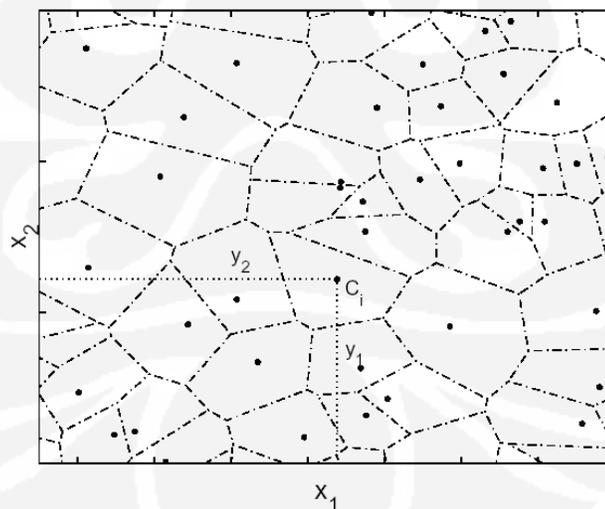
$$\tilde{c}_n = \sum_{k=1}^K (\log \tilde{S}_k) \cos \left[n \left(k - \frac{1}{2} \right) \frac{\pi}{K} \right], \quad k = 1, 2, \dots, K \quad (2.5)$$

2.4.3 Kuantisasi Vektor (VQ)

VQ adalah proses dari pemetaan vektor dari ruang vektor yang besar menjadi sebuah wilayah yang terbatas. Masing-masing wilayah ini disebut *cluster* dan dapat direpresentasikan dengan *centroid* yang disebut *codeword*. Koleksi dari semua *codeword* disebut *codebook* yang berhubungan untuk suara yang telah diketahui^[14].

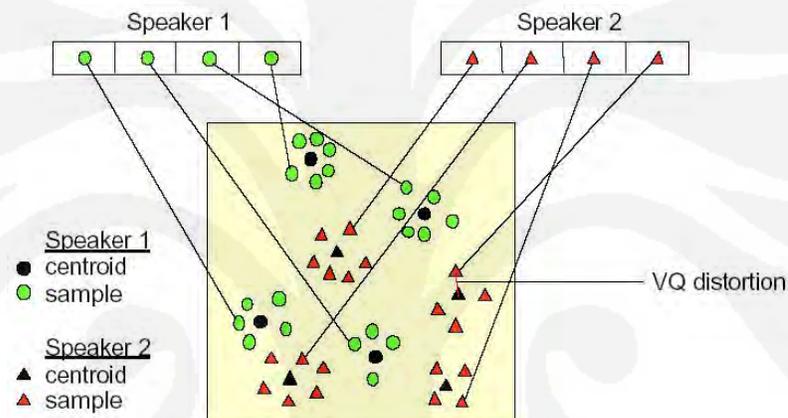
VQ diinterpretasikan dengan skalar kuantisasi. Sinyal input akan dikuantisasi menjadi *codebook* $C = \{y_k \mid k = 1, \dots, N\}$. Hampir keseluruhan sinyal input merupakan sebuah vektor yang harus dikodekan kedalam ruang multidimensi. Gambar 2.13 merupakan contoh ruang dua dimensi dari *codebook*. Gambar 2.12 menunjukkan partisi dari ruang multidimensi sebuah input vektor yang dibagi menjadi L wilayah yang dapat dinotasikan sebagai $P = \{C_1, C_2, \dots, C_L\}$ dimana^[10]

$$C_i = \{x \mid d(x, y_i) \leq d(x, y_j), j \neq i\} \quad (2.6)$$



Gambar 2.12 Codebook dari suatu input vector^[3]

Gambar 2.13 menunjukkan konseptual diagram untuk mengilustrasikan proses *recognition*. Pada Gambar 2.13 hanya digambarkan 2 suara dari 2 *speaker* dalam ruang akustik dua dimensi. Lingkaran menunjukkan vektor akustik dari suara 1, sedangkan segitiga adalah vektor akustik dari suara 2. Dalam tahap *training*, *VQ codebook* untuk masing-masing suara yang telah diketahui dibuat dengan mengumpulkan vektor akustik *training*-nya menjadi sebuah *cluster*. Hasil *codeword*-nya ditunjukkan pada Gambar 2.13 dengan lingkaran dan segitiga hitam untuk suara 1 dan 2. Jarak dari sebuah vektor ke *codeword* terdekat disebut *distortion* ^[14].



Gambar 2.13 Diagram konsep pembentukan codebook dengan vector quantization. Suara satu dan lainnya dapat dibedakan berdasarkan lokasi dari centroid-nya^[14]

Pada tahap *recognition*, sebuah input dari suara yang tidak dikenal akan dilakukan proses *vector-quantized* dengan menggunakan semua *trained codebook* dan selanjutnya dihitung total *VQ distortion*-nya. Total *distortion* yang paling kecil antara *codeword* dari salah satu suara dalam *database* dan *VQ codebook* dari suara input diambil sebagai hasil identifikasi.

Dalam pembentukan *codebook* untuk iterasi guna memperbaiki VQ digunakan *General Lloyd Algorithm (GLA)* atau yang sering disebut dengan *LBG Algorithm*. *LBG VQ algorithm* tersebut dapat diimplementasikan dengan prosedur rekursif sebagai berikut^[8]:

1. Mendesign suatu vektor *codebook* yang merupakan *centroid* dari keseluruhan vektor *training*.

2. Menjadikan ukuran *codebook* dua kali lipat dengan membagi masing-masing *current codebook* C_n menurut aturan

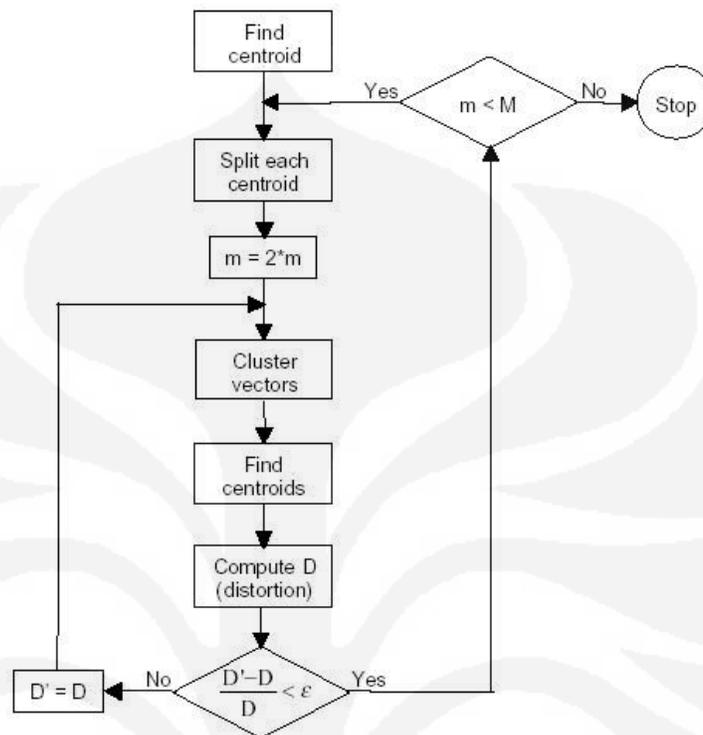
$$C_n^+ = C_n (1 + \varepsilon) \quad (2.7)$$

$$C_n^- = C_n (1 - \varepsilon) \quad (2.8)$$

dimana n bervariasi dari 1 sampai dengan *current size codebook* dan ε adalah parameter *splitting* ($\varepsilon = 0.01$).

3. *Nearest Neighbour Search*, yaitu mengelompokkan *training vector* yang mengumpul pada blok tertentu. Selanjutnya menentukan *codeword* dalam *current codebook* yang terdekat dan memberikan tanda vektor yaitu *cell* yang diasosiasikan dengan *codeword* yang terdekat.
4. *Centroid update*, yaitu menentukan *centroid* baru yang merupakan *codeword* yang baru pada masing-masing *cell* dengan menggunakan *training vector* pada *cell* tersebut.
5. Iterasi 1
mengulang step 3 dan 4 sampai jarak rata-rata dibawah *present threshold*.
6. Iterasi 2
mengulang step 2, 3, 4 sampai *codebook* berukuran M .

Gambar 2.14 menunjukkan diagram alur, langkah detail dari *LBG algorithm*. *Cluster vektor* menerapkan prosedur *nearest neighbour search* yang mana menandai masing-masing *training vektor* ke sebuah *cluster* yang diasosiasikan dengan *codeword* terdekat. ‘*Find centroid*’ merupakan prosedur meng-*update centroid* untuk menentukan *codeword* yang baru. ‘*Compute D (distortion)*’ berarti menjumlah jarak semua *training vektor* dalam *nearest neighbour search* terhadap *centroid* untuk menentukan besarnya *distortion*.



Gambar 2.14 Diagram alur dari LBG algorithm^[14]

2.5 Hidden markov Model (HMM)

Hidden Markov models (HMM) merupakan model dengan pendekatan statistik yang digunakan dalam berbagai implementasi pengenalan suara. *Time variance* dalam suatu bahasa dimodelkan sebagai proses *Markov* dengan *discrete state*. Masing-masing *state* menghasilkan observasi menurut karakteristik distribusi probabilitas dari *state* tersebut. Observasi dapat mengambil pada harga diskrit atau kontinyu. Observasi merepresentasikan durasi waktu yang tetap yang disebut *frame*. Pada model ini *state* tidak secara langsung dapat diamati, hal ini yang menjadikan model ini disebut sebagai *hidden Markov model*.

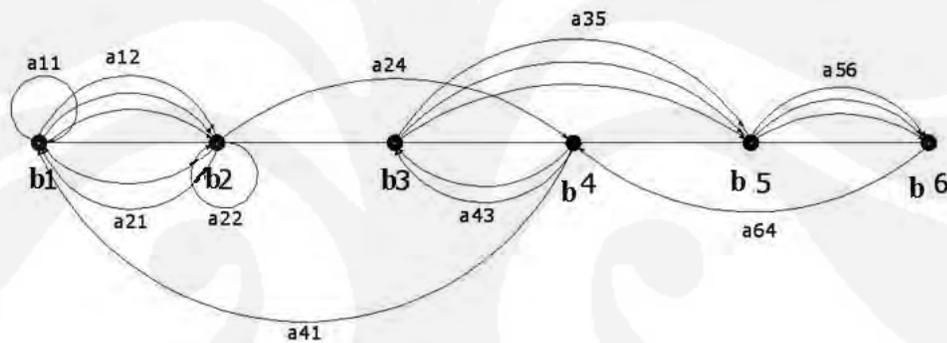
2.5.1 Tipe-tipe Hidden Markov Models

Salah satu cara untuk mengklasifikasikan HMM adalah dengan melihat bentuk matrix transisinya (A) dari rantai markov. Bentuk yang umum adalah bentuk *ergodic* atau bentuk yang setiap *state* saling terhubung (*fully connected HMM*). Seperti terlihat pada Gambar 2.15, a untuk $N = 6$ state model, model ini

mempunyai nilai a_{ij} antara 0 dan 1. Nilai 0 dan 1 tidak termasuk, jika tidak maka bentuk model *ergodic* tidak akan terwujud. Matriks transisi untuk *ergodic* model dapat dimisalkan seperti dibawah ini^[8].

$$A_{mn} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} \dots & a_{3n} \\ a_{n1} & a_{n2} & a_{n3} \dots & a_{nn} \end{bmatrix} \quad (2.9)$$

HIDDEN MARKOV MODEL



$a_{11} = 1/16$	$b_1 = 1/16$
$a_{12} = 3/16$	$b_2 = 2/16$
$a_{22} = 1/16$	$b_3 = 2/16$
$a_{24} = 1/16$	$b_4 = 3/16$
$a_{35} = 3/16$	$b_5 = 2/16$
$a_{56} = 3/16$	$b_6 = 1/16$
$a_{64} = 1/16$	
$a_{43} = 2/16$	
$a_{41} = 1/16$	
$a_{21} = 2/16$	

Gambar 2.15 HMM dengan 6 state^[15]

Untuk *voice recognition* atau *speech recognition* model yang tepat digunakan adalah model *left-right HMM* atau biasa disebut *Bakis Model*. Mengikuti kenyataan bahwa dalam suara aliran waktu terus bertambah, hal ini dapat diwakili oleh perpindahan *state* dari kiri ke kanan (*left-to-right*). Seperti terlihat pada Gambar 2.15^[14]. Pada a_{11} menunjukkan perpindahan *state* ada 1 sehingga nilai kemungkinan munculnya yaitu $1/16$, a_{12} menunjukkan perpindahan *state* ada 3 sehingga nilai kemungkinan munculnya yaitu $3/16$, dan begitu selanjutnya untuk semua matriks. Pada b_1 menunjukkan munculnya *state* menuju ke 1 ada 4 yaitu a_{11} , a_{12} , a_{21} , dan a_{41} sehingga nilai kemungkinan munculnya $4/16$, begitu selanjutnya untuk semua *state*.

2.5.2 Elemen-elemen Hidden Markov Model (HMM)

Hidden Markov Model (HMM) memiliki 3 (tiga) parameter utama^[1] yang harus dicari nilainya terlebih dahulu. Ketiga parameter tersebut adalah sebagai berikut.:

1. **Parameter A** disebut sebagai probabilitas transisi matriks A, merupakan probabilitas kedudukan suatu *state* terhadap semua *state* yang ada, termasuk kedudukan terhadap *state* itu sendiri. Penggunaan probabilitas transisi dapat ditunjukkan pada gambar berikut :

Parameter A pada HMM dinyatakan dalam sebuah matriks dengan ukuran $n \times n$ dimana n adalah jumlah *state* yang ada. Pada gambar terdiri dari 6 (enam) *state* sehingga setiap *state* memiliki 6 (enam) hubungan transisi, maka parameter A dapat dituliskan dalam bentuk matriks seperti pada persamaan berikut :

$$A = a_{66} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & a_{56} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} \end{bmatrix}$$

2. **Parameter B** disebut sebagai probabilitas *state*, merupakan probabilitas kemunculan suatu *state* dalam deretan seluruh *state* yang ada.

Parameter B dalam HMM dituliskan dalam bentuk matriks kolom dengan ukuran $n \times 1$ dimana n merupakan jumlah seluruh *state* yang ada. Misalnya terdapat n buah *state* dalam suatu kondisi, maka matriks B yang terbentuk ditunjukkan oleh persamaan berikut :

$$B = \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ b_n \end{bmatrix}$$

3. **Parameter π** disebut sebagai probabilitas awal, yaitu probabilitas kemunculan suatu *state* di awal.

Sama halnya dengan parameter B, parameter π juga dituliskan dalam bentuk matriks kolom dengan ukuran $M \times 1$ dimana M adalah jumlah *state*-nya. maka parameter π yang dihasilkan akan ditunjukkan seperti pada persamaan berikut :

$$\pi = \begin{bmatrix} c_1 \\ c_2 \\ \cdot \\ \cdot \\ c_n \end{bmatrix}$$

Keterangan :

n merupakan jumlah pemenggalan (*state*) dalam model.

Elemen π , A, dan B merupakan parameter-parameter markov dalam HMM yang tidak diketahui atau tersembunyi (*hidden*). Ketiga parameter tersebut digabungkan menjadi sebuah parameter HMM dan dapat dituliskan dalam bentuk $\lambda = (A, B, \pi)$.

Jika diberikan suatu model $P(O|\lambda)$ dengan probabilitas urutan observasi $O = O_1, O_2, \dots, O_T$, maka untuk mengetahui nilai probabilitas observasinya diperlukan suatu urutan *state* yang tetap, misalkan

$$Q = q_1 q_2 \dots q_T \dots \dots \dots \quad (2.16)$$

di mana q_t adalah *initial-state*. Maka probabilitas urutan observasi O untuk urutan *state* persamaan (2.27) adalah

$$P(O|Q, \lambda) = \prod_{t=1}^T P(Q_t|q_t, \lambda) \dots\dots\dots (2.17)$$

$$\text{Sehingga diperoleh } P(O|Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \dots b_{q_T}(O_T) \dots\dots\dots (2.18)$$

Probabilitas dari urutan *state* Q maka dapat ditulis sebagai berikut

$$P(Q|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \dots a_{q_{T-1} q_T} \dots\dots\dots (2.19)$$

Probabilitas gabungan O dan Q yang merupakan probabilitas saat O dan Q muncul bersamaan adalah hasil perkalian dari keduanya atau dapat ditulis sebagai berikut :

$$P(O, Q|\lambda) = P(O|Q, \lambda) P(Q, \lambda) \dots\dots\dots (2.20)$$

Probabilitas observasi O diperoleh dengan menjumlahkan probabilitas gabungan dari semua kemungkinan urutan *state* q , yaitu :

$$P(O|\lambda) = \sum_{all Q} P(O|Q, \lambda) P(Q|\lambda) \dots\dots\dots (2.21)$$

atau dapat juga dituliskan sebagai berikut :

$$P(O|\lambda) = \sum_{q_1 q_2 \dots q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \dots a_{q_{T-1} q_T} b_{q_T}(O_T) \dots\dots\dots (2.22)$$

$$\text{Log of Probability (LoP)} = \text{Log } P(O|\lambda) \dots\dots\dots (2.23)$$

Jika terdapat keadaan dimana :

State 1 : waveform segment 1 (W_1)

State 2 : waveform segment 2 (W_2)

State 3 : waveform segment 3 (W_3)

State 4 : waveform segment 4 (W_4)

State 5 : waveform segment 5 (W_5)

Lop =

Maka, probabilitas dari observasi HMM :

$$\text{Suara 1} \rightarrow (w_1, w_2, w_2, w_1, w_1) = c_1 * a_{12} * b_2 * a_{22} * b_2 * a_{21} * b_1 * a_{11} * b_1$$

$$\text{Suara 2} \rightarrow (w_1, w_2, w_1, w_3, w_1) = c_1 * a_{12} * b_2 * a_{21} * b_1 * a_{13} * b_3 * a_{31} * b_1$$

⋮

$$\text{Suara x} \rightarrow (w_4, w_5, w_4, w_5, w_4) = c_4 * a_{45} * b_5 * a_{54} * b_4 * a_{45} * b_5 * a_{54} * b_4$$

proses yang terjadi adalah :

1. Gelombang yang telah terbagi menjadi gelombang-gelombang kecil pada *frame blocking* akan dikenali melalui *codebook* yang dimiliki. Pada proses pencocokan dengan *codebook* akan dihitung jarak dari tiap gelombang dengan *centroid-centroid*. Jarak terdekat akan menentukan urutan kode observasi.
2. Gelombang yang telah dikenali berdasarkan *codebook* akan membentuk suatu *state*. Dari *state* ini akan dicari nilai masing-masing parameter HMM-nya, yang perhitungannya dicocokkan dengan nilai pada parameter HMM *database*.

Dari contoh di atas, dapat diketahui bahwa suara 1 terbentuk dari gelombang w_1 , gelombang w_2 , gelombang w_2 , gelombang w_1 , dan gelombang w_1 . Tiap suara dibentuk oleh susunan gelombang yang berbeda-beda. Susunan-susunan gelombang tersebut memiliki probabilitas transisi yang bergantung terhadap perubahan gelombangnya.

BAB III

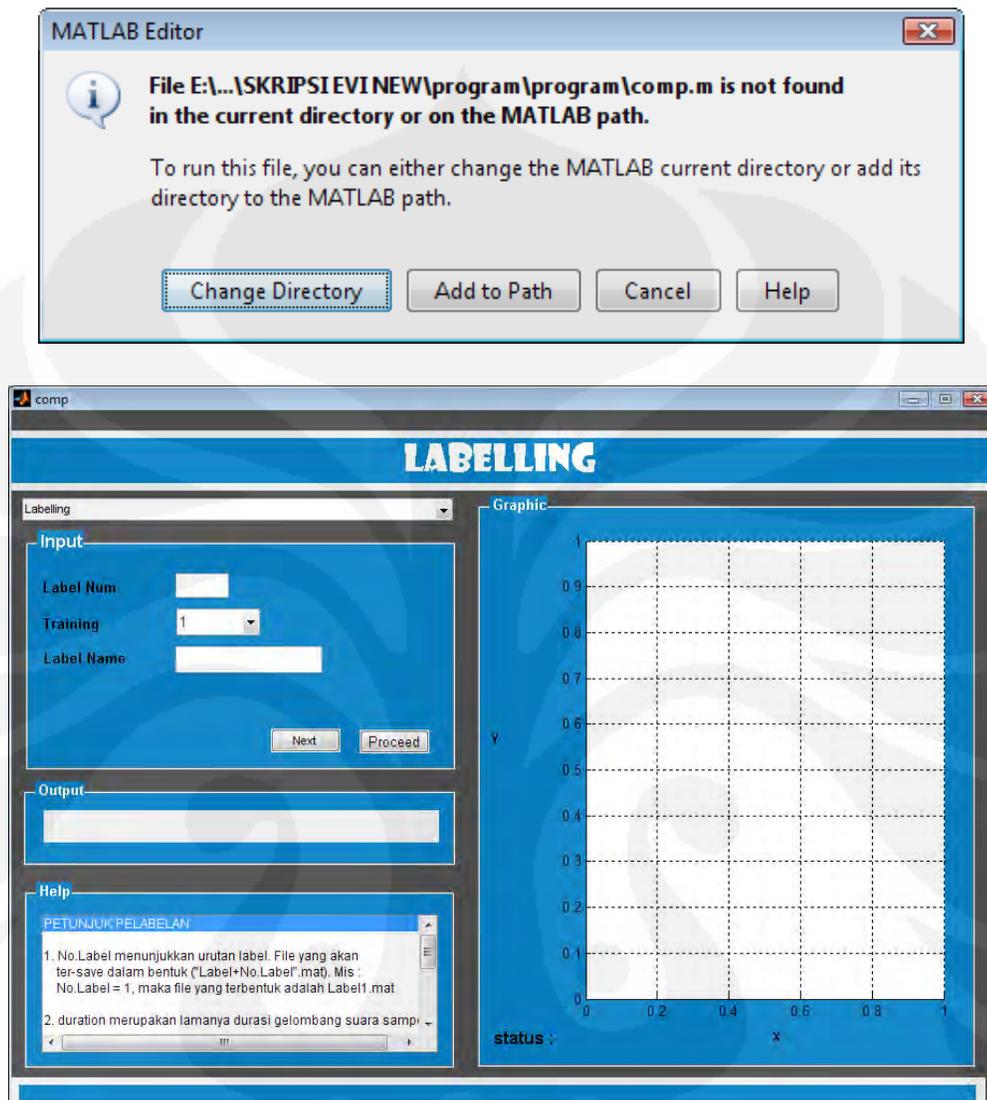
MEMODIFIKASI SISTEM *SOFTWARE* YANG EKSIS DAN ANALISIS GETARAN SUARA PENYAKIT PARU-PARU

Untuk menampilkan nama karakter dari suara penyakit paru-paru yang akan diidentifikasi atau dikenal (*recognized*), dilakukan dengan menggunakan teknik *Hidden Markov Model* (HMM). Sebelum proses pengenalan dilakukan, terlebih dahulu dibentuk basis data (*database*) untuk menyimpan *codeword* dan parameter HMM dari seluruh sampel suara paru-paru sebanyak 10 buah sampel dan 10 buah data uji. Parameter-parameter di dalam basis data selanjutnya dilakukan digunakan sebagai pembanding dalam proses pengenalan.

Perangkat lunak (*software*) pengenalan getaran suara penyakit paru-paru ini dilakukan dengan menggunakan komputer dengan spesifikasi sebagai berikut :

Sistem Operasi	: Windows Vista Home Premium (64-bit)
Prosesor	: AMD Turion X2 Dual Core 2 GHz
VGA Card	: ATI Radeon 3100 Graphics 1918 MB (400MHz)
Software	: Matlab 2008
Memory	: 4096 MB 800MHz SDRAM

Perangkat lunak ini sebelumnya telah dikembangkan oleh saudara Ahmad Mujadid Amin dengan judul “Simulasi dan Analisis Program Speech To Text Menggunakan Metode Hidden Markov Model” sehingga dalam perancangan ini dicari input suara penyakit paru-paru dan mengubah frekuensi sampling menjadi 8000 Hz. Dalam perancangan sistem pengenalan penyakit paru-paru ini, terdapat dua proses utama yang harus ditempuh, yaitu pembuatan *database* dan pengenalan suara (denyut paru-paru). Perangkat lunak ini dibuat dengan menggunakan GUI untuk mempermudah penggunaan dimana didalamnya terdapat *function-function* untuk melakukan perhitungan-perhitungan. Untuk memulai memanggil GUI terlebih dahulu harus merubah *Current Directory* pada *Matlab* sesuai dengan letak folder dimana program ini berada, dan menuliskan ‘menu’ pada *Command Window Matlab*. Setelah itu akan terlihat tampilan seperti Gambar 3.1



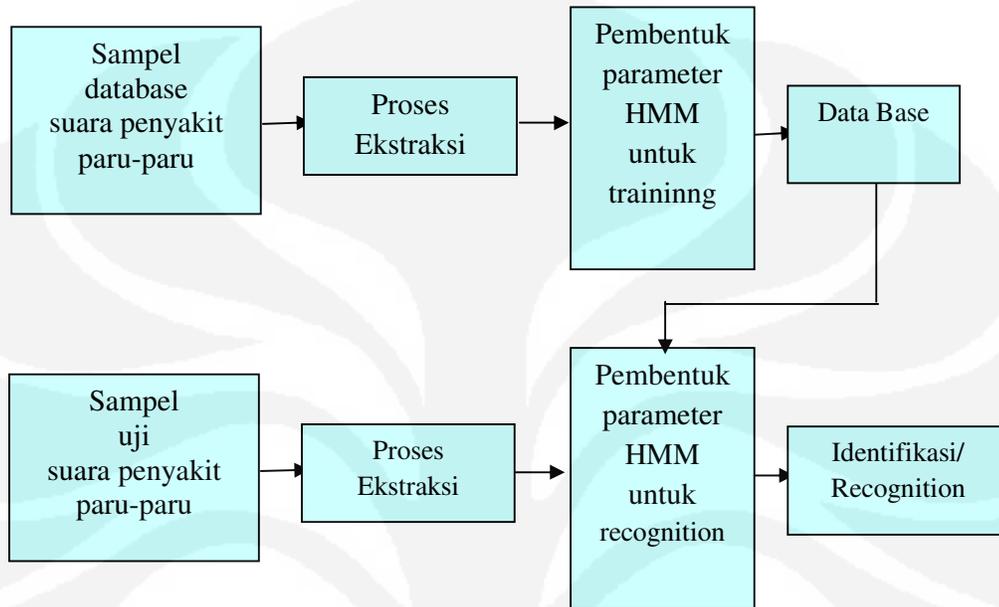
Gambar 3.1 Tampilan Menu Setelah Current Directory

3.1 Blok Diagram Sistem dan Fungsinya

Sistem terdiri atas beberapa bagian yang dapat digambarkan menjadi blok diagram seperti yang terlihat pada gambar 3.1.

Dalam blok diagram dibawah terbagi menjadi 2 bagian, yakni bagian atas yang merupakan process training *Hidden Markov Model* dan bagian bawah yang merupakan proses identifikasi sinyal. Proses training ini dilakukan untuk

menghasilkan database pola-pola hidden markov dari tiap-tiap label yang kemudian digunakan sebagai referensi pada proses identifikasi sinyal.

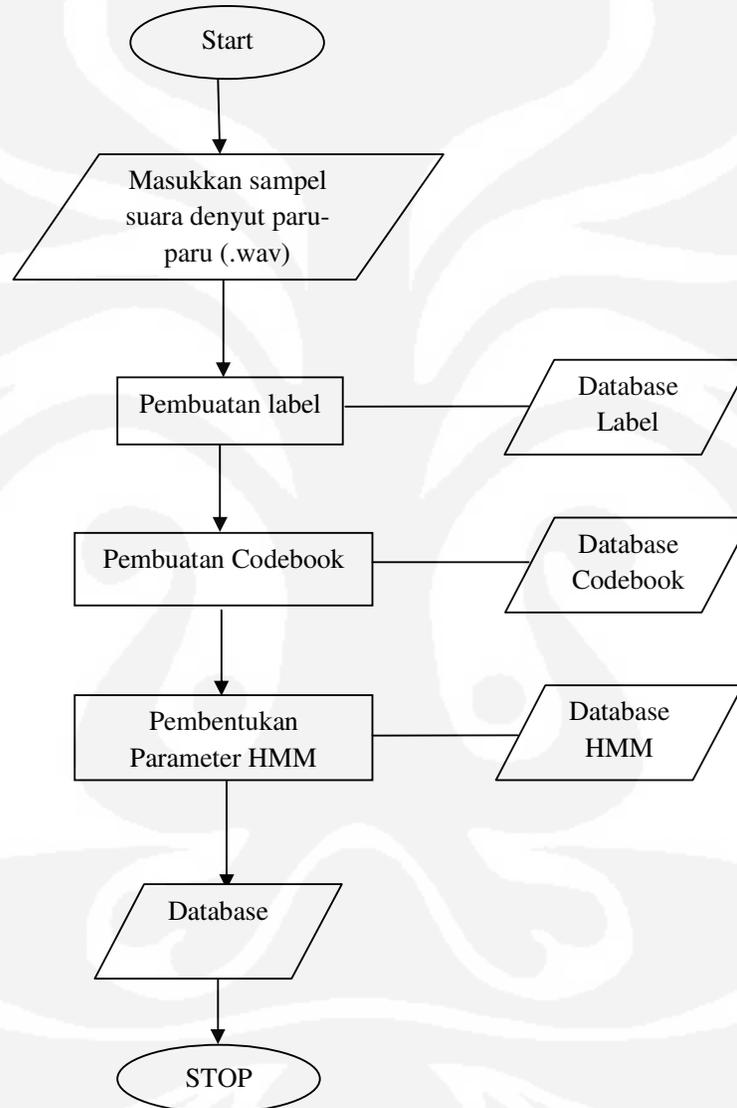


Gambar 3.2 Blok diagram sistem pengenalan getaran suara penyakit paru-paru

Proses training ini dimulai dengan memberikan beberapa sampel data gelombang yang kemudian diolah melalui proses ekstraksi yaitu dengan cara memenggal gelombang dengan interval waktu tertentu, lalu diubah ke domain frekuensi dengan menggunakan fungsi FFT yang tersedia pada *library matlab*. Matrik dari ekstraksi kemudian di cari pola-pola sinyal yang khas dari tiap label menggunakan teknik *Hidden Markov Model*. Hasil tersebut kemudian disimpan dalam database. Setelah proses training selesai maka akan didapatkan karakteristik suara penyakit paru-paru, jika terdapat suara penyakit paru-paru yang akan dideteksi maka dari gelombang suara tersebut akan dilakukan proses *ekstraksi* kembali seperti awal, setelah proses ekstraksi maka akan dilakukan *recognition* dengan teknik hidden markov yang mengacu pada karakteristik dalam *database*. Kemudian akan didapatkan hasil apakah jenis dari gelombang suara penyakit paru-paru tersebut.

3.2 Pembuatan Database

Masing-masing sampel suara paru-paru merupakan sinyal masukan. Sampel tersebut akan mengalami tiga proses utama secara berurutan, yaitu pembuatan label, pembuatan *codebook*, dan pembentukan parameter-parameter HMM. Secara umum akan digambarkan melalui *flowchart* pada gambar 3.2



Gambar 3.3 Diagram alir pembuatan database untuk deteksi suara penyakit paru-paru

3.2.1 Pelabelan

Pada proses ini dilakukan pembuatan label untuk masing-masing jenis penyakit. Tiap-tiap sampel suara penyakit akan dikelompokkan dalam satu label sesuai dengan jenis penyakitnya dan label tersebut diberi nama sesuai dengan nama penyakit yang dimaksud, sehingga jumlah label sama dengan jumlah penyakit. Nama label inilah yang nantinya akan menjadi keluaran akhir pada simulasi ini. *File-file* sampel suara tersebut akan mengalami proses *sampling* dengan frekuensi sampling sebesar 8 kHz. Berikut ini adalah algoritma proses pembuatan label :

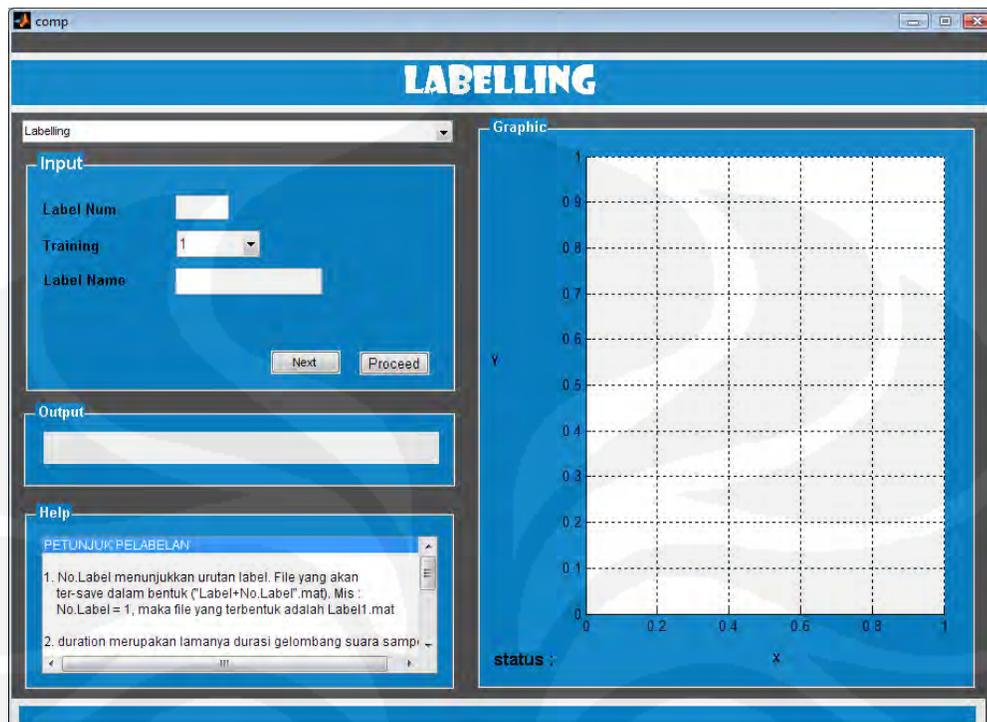
```

Untuk i = 1 sampai banyaknya penyakit
    Masukkan nama penyakit;
    Masukkan jumlah sampel yang didaftarkan ;
    Masukkan nilai durasi sinyal sampel;
    Baca file sampel (.wav);
    Nama label [i] = nama penyakit ;
Kembali
  
```

Gambar 3.4 dibawah ini adalah tampilan program pelabelan :

Program ini terdiri dari empat jenis inputan, yaitu : *Label Num*, *Duration*, *Training*, dan *Label Name*.

1. *Label Num* adalah urutan label yang akan dibuat, dan nantinya file yang akan tersimpan dalam bentuk format “(Label+Label Num).mat”.
2. *Training* menunjukkan banyaknya sampel yang akan diproses untuk setiap labelnya. Besarnya dapat ditentukan sesuai dengan keinginan user.
3. *Label Name* menunjukkan nama file sampel suara yang akan diproses. *Label Name* ini juga yang nantinya akan menjadi keluaran akhir dari proses simulasi ini.



Gambar 3.4 Tampilan Program Pembuatan Label

Sebagai contoh, *Label Num* = 1, *Duration* = *Training* = 5 dan *Label Name* = cracle. Maka, file sampel suara yang akan diproses sebanyak 5 buah secara berurutan dan harus diberi nama “cracle1.wav” – “cracle5.wav”, sedangkan file label untuk cracle yang akan terbentuk bernama Label1.mat.

Berikut contoh perintah program dalam Pelabelan yaitu :

```
function make_labels(rep,time)
    h=guidata(gcbo);
    index_label=get(h.index_label,'String');
    index_label=str2num(index_label);
    nama_label=get(h.nama_label,'String');
    etiqueta=nama_label;
    makelabel(['label' int2str(index_label)],etiqueta,rep,time);

function makelabel(filename,etiqueta,times,time)
    %fs=11025;
    fs=8000;
    label=zeros(time*8000,times);
    number=1;
    while (number<=times)
        nama_file = [etiqueta int2str(number)];
        etiqueta2 = nama_file;%nol1...nol5
```

```

%eval(['load' ' ' etiqueta2]);
%speech = eval(etiqueta2);
% if exist( [ etiqueta2 '.wav' ] )
    [speech]=wavread(etiqueta2,time*8000);
% else
%     se = guidata(showError);
%     errorText = [ etiqueta2 '.wav does not exist!!!' ];
%     set( se.errorText, 'string', errorText );
%     return;
% end
label(:,number)=speech;
number=number+1;
end
eval(['save' ' ' filename ' ' 'label etiqueta'])
h=guidata(gcbo);
index_label=get(h.index_label,'String');
index_label=str2num(index_label);
index_label = index_label + 1;
set(h.index_label,'String',index_label);

```

Kemudian pembacaan file gelombang ini dilakukan dengan menggunakan *function* matlab yaitu *wavread*. Contoh perintah yang digunakan untuk pembacaan file ini adalah

```
[speech]=wavread(etiqueta2,time*8000);
```

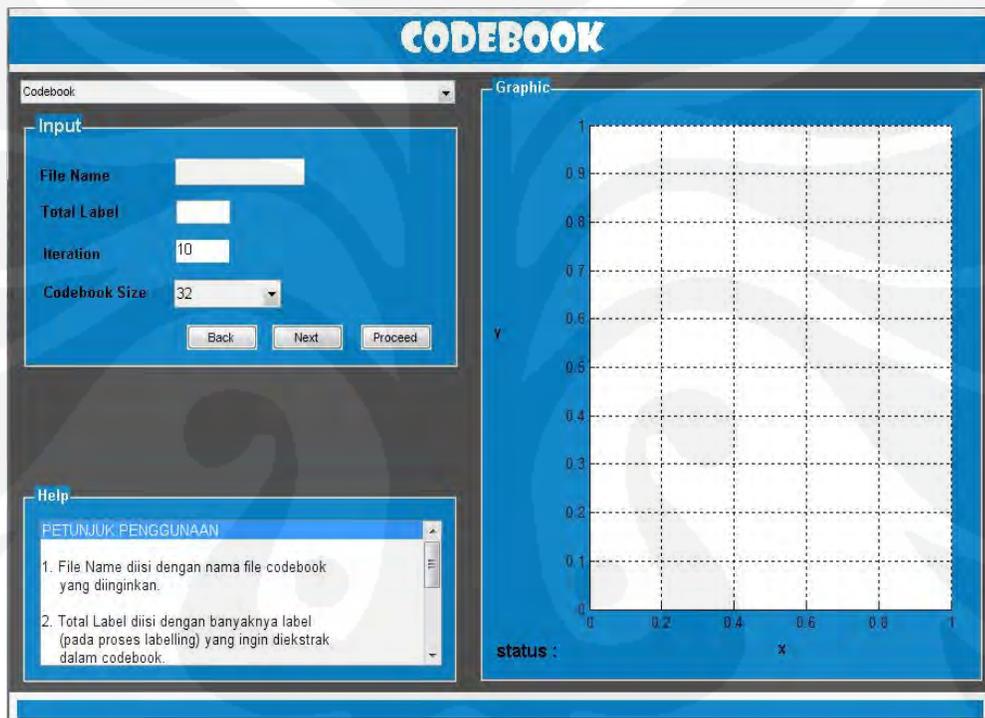
Setelah melakukan eksekusi program, maka keluaran yang dihasilkan pada proses pembuatan label ini adalah berupa matriks yang disimpan dalam format “.mat”.

3.2.2 Pembuatan *Codebook*

Pembuatan codebook ini merupakan tahap selanjutnya dari pembuatan *database*. Pada tahap ini, mulai dilakukan proses pengolahan sinyal suara untuk sampel-sampel didalam tiap-tiap label yang telah di bentuk. Proses ini dimulai dengan ekstraksi (MFCC) sampel-sampel suara paru-paru (telah dijelaskan pada bab sebelumnya), yang akan menghasilkan titik-titik vektor melalui proses FFT. Titik-titik ini kemudian dipetakan pada suatu grafik dengan teknik kuantisasi vektor (VQ). VQ merupakan proses pemetaan vektor dari ruang vektor yang besar menjadi daerah yang terbatas (*cluster*). Setiap *cluster* ini direpresentasikan oleh sebuah titik *centroid*. Satu kombinasi dari beberapa *centroid* ini merupakan perwakilan untuk satu sampel, yang disebut dengan *codeword*. Kumpulan semua

codeword disebut dengan *codebook*.

Titik-titik vektor tersebut dipetakan dan titik-titik yang berdekatan dikuantisasikan ke satu titik vektor sehingga diperoleh beberapa titik vektor atau kumpulan *centroid*. Kemudian *codeword* dari setiap sampel penyakit yang diperoleh dari proses tersebut akan disimpan menjadi sebuah *codebook*. Tiap-tiap *codeword* ini yang nantinya akan menjadi masukan bagi proses pembentukan parameter HMM.



Gambar 3.5 Tampilan program pembuatan codebook

Algoritma pembuatan *codebook* secara umum dapat dituliskan sebagai berikut :

- definisikan besar vektor
- untuk $i = 1$ sampai jumlah sampel
 - ekstraksi sampel [i];
 - hitung FFT untuk setiap sample [i];
 - sample point [i] = nilai FFT;
 - plot grafik;
- kembali

```

definisikan ukuran codebook dan iterasi;
untuk j = 1 sampai jumlah cluster
    hitung centroid berulang sebanyak iterasi;
    simpan centroid [j] berdasarkan urutan labelnya;
kembali

```

Pada program ini terdapat empat buah inputan, yaitu :

1. *File name* diisi dengan nama file yang diinginkan yang nantinya akan tersimpan dalam format “.mat”.
2. *Codebook size* merupakan ukuran *codebook* yang akan digunakan. Pada program ini tersedia ukuran *codebook* 32, 64 dan 128.
3. *Iteration* merupakan banyaknya proses pengulangan yang dilakukan dalam menentukan *centroid* agar mendapatkan *centroid* yang cukup presisi. Semakin besar jumlah iterasinya, maka akan semakin presisi letak *centroid* yang didapatkan, namun dengan mengambil iterasi yang sangat tinggi proses pembuatan *codebook* akan berjalan sangat lambat, oleh karena itu iterasi yang dilakukan juga tidak perlu terlalu besar. Dalam skripsi ini ditentukan *default* untuk besarnya iterasi adalah 10 dengan harapan letak *centroid* yang diperoleh cukup presisi dan waktu proses lebih cepat.
4. *Total label* dimasukkan sesuai dengan jumlah label yang telah dibuat sebelumnya pada proses pelabelan.

Setelah memasukkan *input-input* yang diperlukan dan mengeksekusi program, maka akan dihasilkan keluaran berupa grafik dan *file* matriks berekstensi “.mat”. Grafik ini merupakan tampilan pemetaan titik-titik vektor untuk semua sampel yang telah terdaftar pada proses pelabelan dan posisi *codeword* yang dicari.

Contoh program pembuatan codebook adalah:

```

function F=extraction(speech,length_frame,overlap,fs)
nmax_frame=floor(length(speech)/length_frame);
sp=[zeros(1,overlap) speech' zeros(1,overlap)]';
frame=0;
for i=overlap+1:length_frame:nmax_frame*length_frame,
    frame=frame+1;

```

```

sframe=sp((i-overlap):(i+length_frame-1+overlap));

[spec, logSpec, f, Pcoeff1, F1(frame, :), F2(frame, :)] = mfcc3(sframe.*ham
ming(length(sframe)), fs);
end;
F=F2;

function Cfinal=VQ_training(speech,M,iteration);

F=extraction(speech,100,50,8000);
save feat_default F
[a,b]=size(F);
Cfinal=split2(F,floor(log(M)/log(2)),iteration);
Code=Cfinal;

```

File *codebook* yang terbentuk akan disimpan pada file yang mempunyai format **.mat* sesuai dengan variasi ukuran *codebook* dan *iterasi*.

```
save codebook Code
```

3.2.3 Pembentukan Parameter HMM

Proses ini bertujuan untuk mencari parameter-parameter HMM yang dibutuhkan dalam proses pengenalan, yaitu $\lambda = (A, B, \pi)$. Untuk mendapatkan parameter-parameter tersebut, dibutuhkan suatu masukan yang dikenal sebagai *state* dalam HMM. Keluaran dari proses pembuatan *codebook* yang berupa nilai-nilai (posisi) *centroid*, merupakan *state* bagi proses ini. *Centroid* ini akan membentuk suatu urutan yang mewakili urutan penggalan masing-masing sampel. Urutan *centroid* inilah yang dijadikan urutan *state* dalam pembentukan parameter HMM. Selanjutnya adalah melakukan proses pembelajaran HMM yaitu perhitungan *log of probability* (LoP) pada tiap-tiap label. Berikut ini adalah algoritma proses pembentukan parameter HMM :

```

Untuk i = 1 sampai banyaknya label
    hitung jumlah centroid;
    state = jumlah centroid;
    hitung nilai probabilitas transisi;
    hitung nilai probabilitas kemunculan state;
    hitung nilai probabilitas observasi;

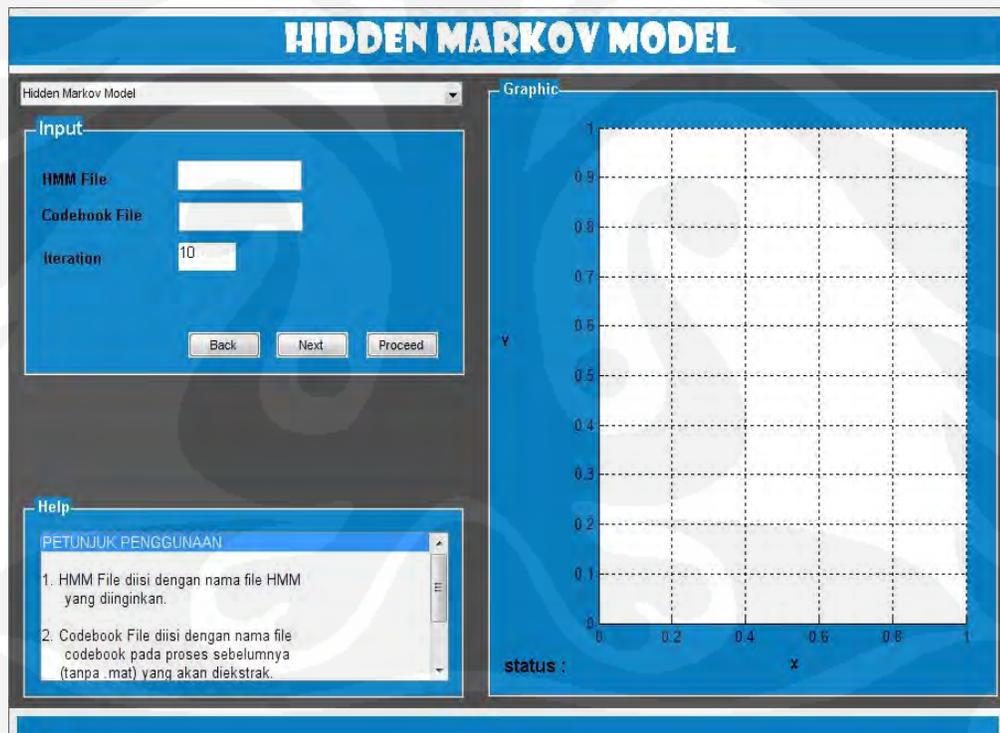
kembali
hitung nilai Log of probability tiap-tiap label;

```

Sedangkan tampilan program untuk proses pembentukan parameter HMM dapat dilihat pada gambar 3.6.

Pada software tersebut terdapat tiga buah masukan, yaitu :

1. *HMM File* diisi dengan nama file HMM yang diinginkan.
2. *Codebook File* diisi dengan nama file *codebook* yang dihasilkan oleh proses sebelumnya yang akan diekstrak.
3. *Iteration* diisi dengan jumlah iterasi yang diinginkan, direkomendasikan sebanyak 10 kali.



Gambar 3.6 Tampilan *software* pembentukan parameter HMM

Setelah pengeksekusian program, maka akan menghasilkan keluaran berupa grafik yang menunjukkan nilai-nilai LoP untuk masing-masing label. Nilai-nilai

ini akan tersimpan dalam bentuk matriks dengan nama *file* yang telah diisi pada *HMM File*.

Tahap-tahap *training* HMM yang dilakukan program setelah tombol *Next* ditekan dapat dijelaskan sebagai berikut:

1. *Meload* file format **.mat* dari *label* dan *codebook* untuk variasi variabel yang sesuai

```
load codebook Code;
```

2. Inisialisasi matrik A, B and Π dengan nilai acak

```
[A,B,p0]=generate_random_model(N,M);
```

3. Dengan menggunakan algoritma *Baum Welch*, dilakukan pelatihan untuk menentukan nilai HMM yang sebenarnya dengan memasukan semua data gelombang suara penyakit paru-paru beberapa kali sampai nilai matriks tidak berubah.

```
[A_new,B_new,p0_new]=baum(A,B,p0,O,iteration);
```

4. Menghitung probabilitas observasi HMM untuk setiap gelombang perubahan fase yang diteliti.

```
P(i,k)=HMM_probability(c);
```

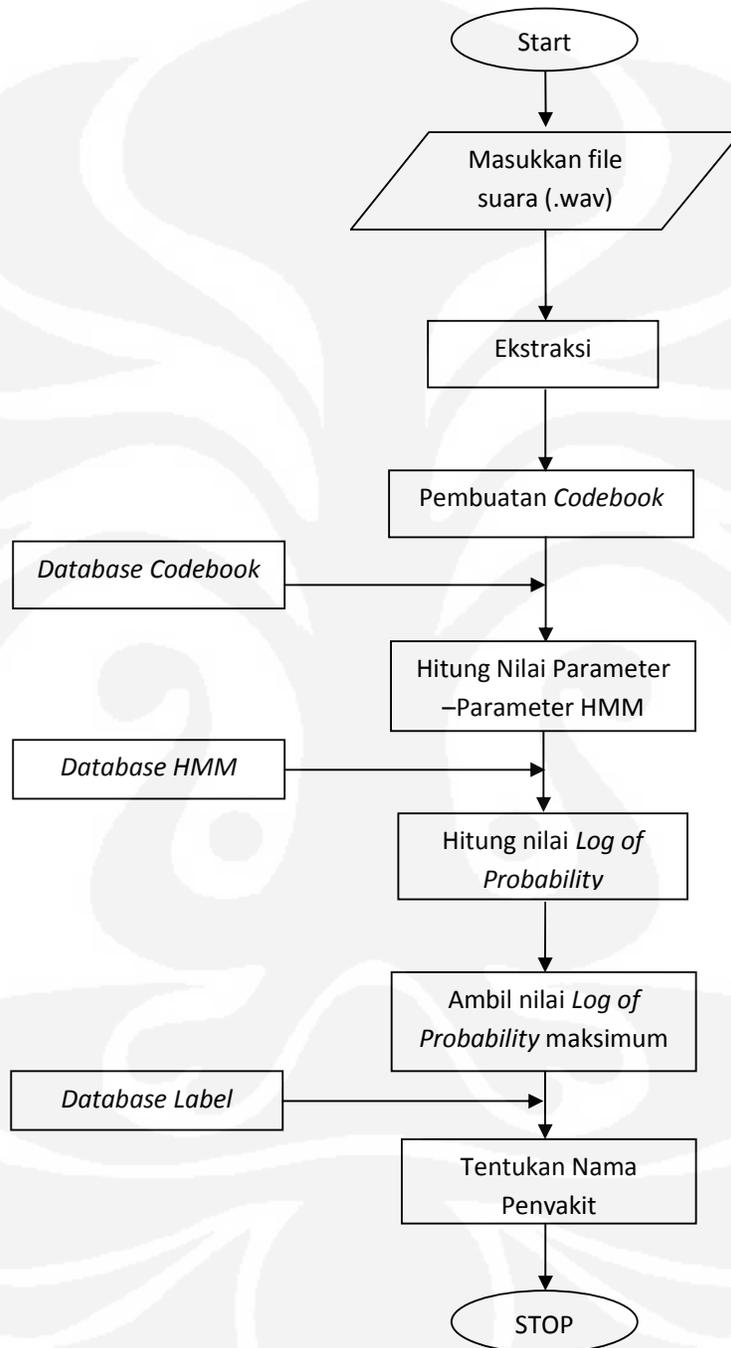
5. Membuat file format **.mat* untuk menyimpan nilai probabilitas HMM setiap gelombang suara penyakit paru-paru yang diteliti sesuai dengan variasi variabel yang sesuai.

```
eval(['save' ' ' model ' ' 'A1 A2 A3 B1 B2 B3 p01 p02 p03  
available N M'])
```

3.3 Proses Pengenalan (*Recognition*)

Proses pengenalan merupakan bagian terpenting dari seluruh simulasi pengenalan penyakit paru-paru ini. Beberapa langkah-langkah yang dilakukan pada proses ini mirip dengan langkah-langkah yang terdapat pada proses pengenalan *database* yang telah dilakukan sebelumnya, hanya saja pada bagian

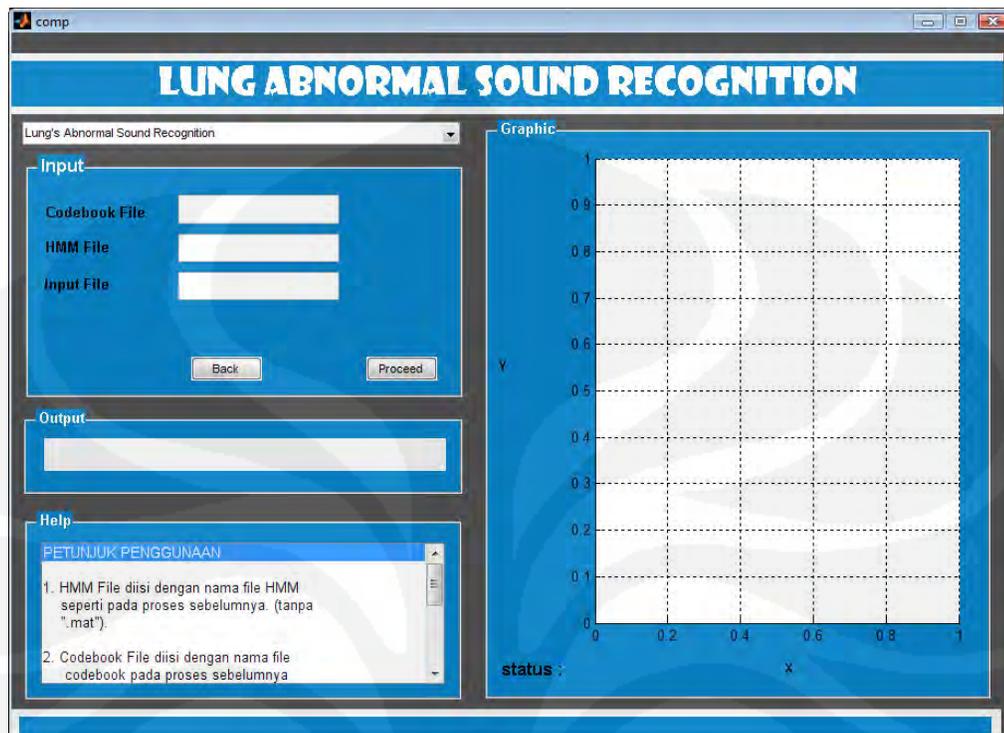
pengenalan tidak lagi dilakukan proses pembelajaran. Gambar 3.7 menunjukkan diagram alir proses pengenalan.



Gambar 3.7 Diagram alir proses pengenalan penyakit paru-paru

File suara paru-paru baru (dalam format “.wav”) yang ingin diidentifikasi, dibaca sebagai masukan dari program. *File* input ini akan diekstraksi menjadi potongan-potongan kecil (frame) dan selanjutnya akan dikonversi ke dalam domain frekuensi dengan transformasi FFT yang menghasilkan vektor-vektor data. Vektor-vektor data ini seolah-olah dipetakan pada *codebook* yang sama dengan *database codebook* yang telah dilatih sebelumnya karena pada *database*, letak *centroid*-nya telah presisi. Oleh karena itu, apabila vektor-vektor data sampel baru ini dilakukan proses demikian, maka dapat ditentukan vektor-vektor data ini lebih dekat ke *centroid* mana pada *database*. Sehingga dapat ditentukan letak *centroid* atau *codeword*-nya. Setelah nilai dan posisi *centroid* pada sampel diketahui, maka dapat ditentukan kombinasi urutan *centroid* sebagai urutan *state* yang nantinya akan digunakan untuk menentukan parameter-parameter HMM.

Dari beberapa parameter HMM ini (yang telah dijelaskan dalam Bab 2), akan ditentukan nilai *Log of Probability* dari sampel terhadap label-label pada *database*, sehingga akan didapat beberapa nilai *Log of Probability* sebanyak jumlah labelnya. Nilai LoP yang paling tinggi merupakan karakteristik yang mewakili sampel yang kemudian menentukan jenis penyakit sebagai hasil keluaran dari program.



Gambar 3.8 Tampilan program pengenalan penyakit paru-paru

Pada tampilan proses pengenalan ini, terdapat tiga jenis masukan, yaitu :

1. *HMM File* diisi dengan nama *file* HMM hasil dari proses sebelumnya.
2. *Codebook File* diisi dengan nama *file* *codebook* hasil dari proses sebelumnya.
3. *Input File* diisi dengan nama *file* sampel yang akan di ujicoba (dalam bentuk ".wav").

Untuk mengeksekusi program ini, tekan *proceed* dan keluaran dari program akan muncul pada layar. Keluaran inilah berupa nama penyakit yang merupakan keluaran akhir dari simulasi pengenalan penyakit paru-paru berdasarkan metode HMM.

Algoritma Proses Pengenalan

```

mulai
  baca file sampel paru-paru (.wav);
  untuk i = 1 sampai jumlah penggal
    ekstraksi sampel paru-paru (.wav);
    hitung FFT sebanyak h;
    cari centroid sampeluji berdasarkan database ;

```

definisikan urutan centroid sebagai *state* HMM;
untuk h = 1 sampai jumlah_label
 Hitung parameter-parameter HMM berdasarkan database;
 Hitung log of probability (LoP) untuk semua label;
 LoP [jumlah_label] = tertinggi
 Kembali
Kembali
Ambil nilai LoP tertinggi untuk satu label ;
Nama penyakit = Nama Label;
selesai

Pembacaan file gelombang ini sama dengan pembacaan file pada proses *make label*. Pembacaan file juga dilakukan dengan menggunakan *function* matlab yaitu *wavread*. Perintah yang digunakan juga sama yaitu :

```
[speech]=wavread(file_inputan,time*8000);
```

Proses *recognition* ini akan terlaksana jika tombol *proceed* ditekan.

BAB IV
**UJI COBA DAN ANALISIS SISTEM IDENTIFIKASI PENYAKIT PARU-
PARU DENGAN METODE HIDDEN MARKOV MODEL**

4.1 Daftar Penyakit dan Jenis Percobaan

Pada penelitian ini, uji coba dilakukan pada suara abnormal dari paru-paru yang mengindikasikan penyakit paru-paru tertentu. Suara tersebut terdiri dari 3 (tiga) jenis, yang dapat dilihat pada tabel 4.1.

Tabel 4.1 Nama File dan Jenis Suara Abnormal Paru-Paru

No.	Jenis Penyakit	Nama File
1.	<i>Bronchial</i>	bronchial11 – bronchial 20
2.	<i>cracle</i>	cracle11 – cracle20
3.	<i>Pleuralisy</i>	pleural11 – pleural20

Suara penyakit paru-paru diatas didapat dari praktik di RS. Al-Islam Bandung bersama dokter bagian Spesialis Penyakit Dalam dan mahasiswi kedokteran Universitas Indonesia.

Uji coba pada penelitian ini dilakukan dengan memvariasikan dua buah parameter yaitu ukuran *codebook*, dan jumlah *database*. Secara lengkap proses uji coba yang dilakukan meliputi :

1. Uji coba sampel dengan ukuran *codebook* 32 dan jumlah *database* 5 buah
2. Uji coba sampel dengan ukuran *codebook* 32 dan jumlah *database* 10 buah
3. Uji coba sampel dengan ukuran *codebook* 64 dan jumlah *database* 5 buah
4. Uji coba sampel dengan ukuran *codebook* 64 dan jumlah *database* 10 buah
5. Uji coba sampel dengan ukuran *codebook* 128 dan jumlah *database* 5 buah
6. Uji coba sampel dengan ukuran *codebook* 128 dan jumlah *database* 10 buah

4.2 Hasil Uji Coba

Setelah melakukan pengujian terhadap sampel-sampel yang telah ditentukan beserta variasi parameter-parameter yang telah dijelaskan sebelumnya, maka

didapatkan data hasil pengujian tersebut. Warna merah menunjukkan program “salah” mendeteksi penyakit dan warna hitam menunjukkan program “benar” mendeteksi penyakit. Berikut ini adalah tabel hasil uji coba untuk masing-masing penyakit.

Tabel 4.2 Hasil Uji Coba Suara Bronchial

No.	<i>Codebook</i>	32		64		128	
	Repetisi Nama File	5	10	5	10	5	10
1.	bronchial11	bronchial	bronchial	bronchial	bronchial	bronchial	bronchial
2.	bronchial 12	bronchial	bronchial	bronchial	bronchial	bronchial	bronchial
3.	bronchial 13	bronchial	bronchial	bronchial	bronchial	bronchial	bronchial
4.	bronchial 14	cracle	bronchial	cracle	bronchial	cracle	bronchial
5.	bronchial 15	cracle	bronchial	cracle	bronchial	cracle	bronchial
6.	bronchial 16	bronchial	bronchial	bronchial	bronchial	bronchial	bronchial
7.	bronchial 17	bronchial	bronchial	bronchial	bronchial	bronchial	bronchial
8.	bronchial 18	cracle	bronchial	cracle	bronchial	cracle	bronchial
9.	bronchial 19	bronchial	bronchial	bronchial	bronchial	bronchial	bronchial
10.	bronchial 20	cracle	bronchial	cracle	bronchial	cracle	bronchial

Tabel 4.3 Hasil Uji Coba Suara Cracle

No.	<i>Codebook</i>	32		64		128	
	Repetisi Nama File	5	10	5	10	5	10
1.	cracle 11	cracle	cracle	cracle	cracle	cracle	cracle
2.	cracle 12	cracle	cracle	cracle	cracle	cracle	cracle
3.	cracle 13	pleural	cracle	pleural	cracle	pleural	cracle
4.	cracle 14	cracle	cracle	cracle	cracle	cracle	cracle
5.	cracle 15	cracle	cracle	cracle	cracle	cracle	cracle
6.	cracle 16	cracle	bronchial	cracle	bronchial	cracle	bronchial
7.	cracle 17	pleural	pleural	pleural	pleural	pleural	pleural
8.	cracle 18	pleural	cracle	pleural	cracle	pleural	cracle
9.	cracle 19	cracle	cracle	cracle	cracle	cracle	cracle
10.	cracle 20	cracle	cracle	cracle	cracle	cracle	cracle

Tabel 4.4 Hasil Uji Coba Suara Pleural

No.	Codebook	32		64		128	
	Repetisi	5	10	5	10	5	10
	Nama File						
1.	pleural11	pleural	pleural	pleural	pleural	pleural	pleural
2.	pleural 12	pleural	pleural	pleural	pleural	pleural	pleural
3.	pleural 13	pleural	pleural	pleural	pleural	pleural	pleural
4.	pleural 14	pleural	pleural	pleural	pleural	pleural	pleural
5.	pleural 15	pleural	pleural	pleural	pleural	pleural	pleural
6.	pleural 16	pleural	pleural	pleural	pleural	pleural	pleural
7.	pleural 17	pleural	pleural	pleural	pleural	pleural	pleural
8.	pleural 18	pleural	pleural	pleural	pleural	pleural	pleural
9.	pleural 19	cracle	pleural	cracle	pleural	cracle	pleural
10.	pleural 20	cracle	pleural	cracle	pleural	cracle	pleural

4.3 Persentase Akurasi

Berikut ini adalah persentase akurasi program terhadap berdasarkan hasil uji coba yang telah diperoleh.

Tabel 4.5 Persentase akurasi semua penyakit

Parameter Penyakit	Codebook 32		Codebook 64		Codebook 128	
	5	10	5	10	5	10
Bronchial	60%	100%	60%	100%	60%	100%
Cracle	70%	80%	70%	80%	70%	80%
Pleural	80%	100%	80%	100%	80%	100%

Tabel 4.6 Persentase akurasi semua parameter

Repetisi	Codebook	32	64	128
5		70%	70%	70%
10		93,33%	93,33%	93,33%

4.4 Analisis Hasil Percobaan

Pada penelitian ini, sampel suara yang ingin diidentifikasi dibandingkan dengan sampel suara yang ada pada *database*. Hanya ada dua kondisi yang mungkin keluar sebagai hasil identifikasi, yaitu :

1. Sistem dapat mengidentifikasi penyakit paru-paru

Kondisi ini terjadi jika hasil pengenalan sesuai dengan masukan. *Label* atau jenis penyakit dengan probabilitas tertinggi merupakan jenis gelombang yang sesuai dengan masukan.

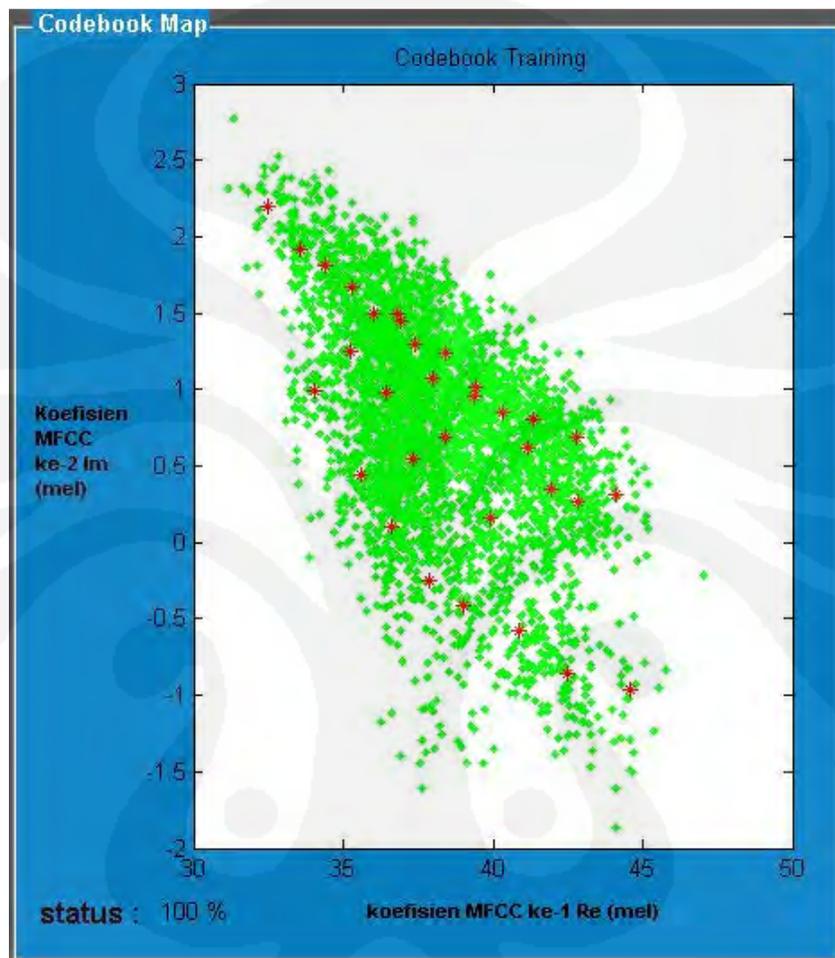
2. Sistem salah mengidentifikasi penyakit paru-paru

Kondisi ini terjadi jika hasil pengenalan berbeda dengan masukan. *Label* atau jenis penyakit dengan probabilitas tertinggi bukan merupakan jenis gelombang yang sesuai dengan masukan. Dalam hal ini, sistem akan menjadikan *label* dengan nilai *log of probability* yang tertinggi sebagai hasil pengenalan.

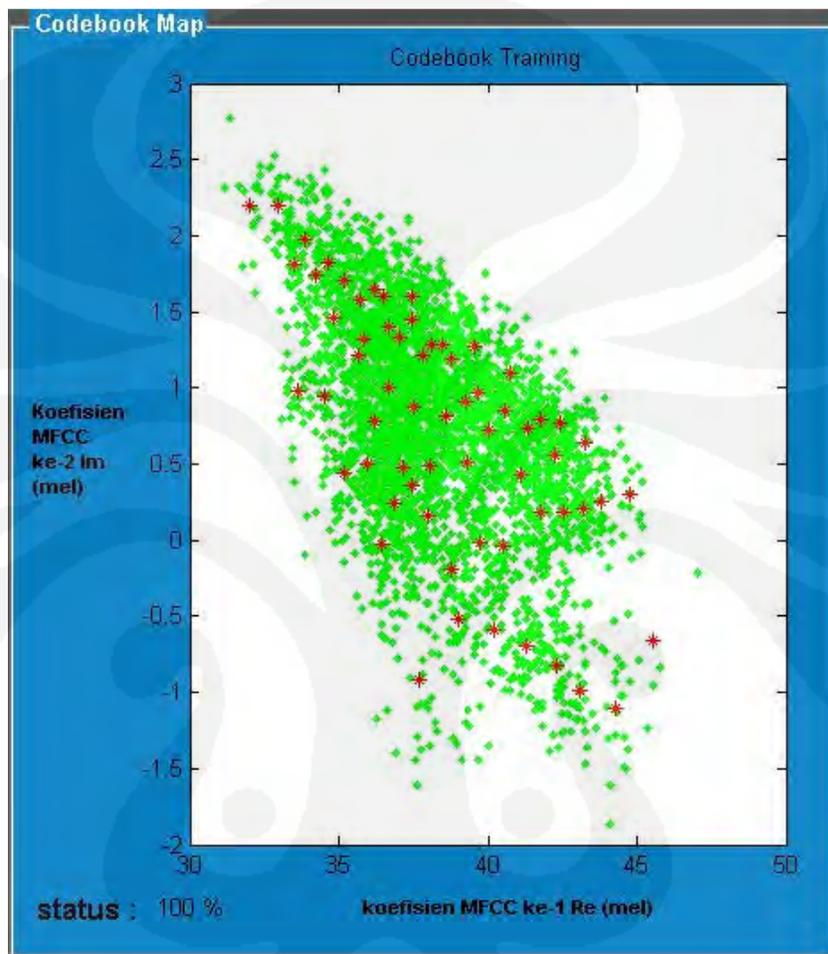
Keberhasilan sistem dalam mengidentifikasi sampel uji tergantung dari beberapa faktor, diantaranya adalah ukuran *codebook* dan jumlah *database*.

4.4.1 Analisis Pengaruh Variasi Ukuran *Codebook*

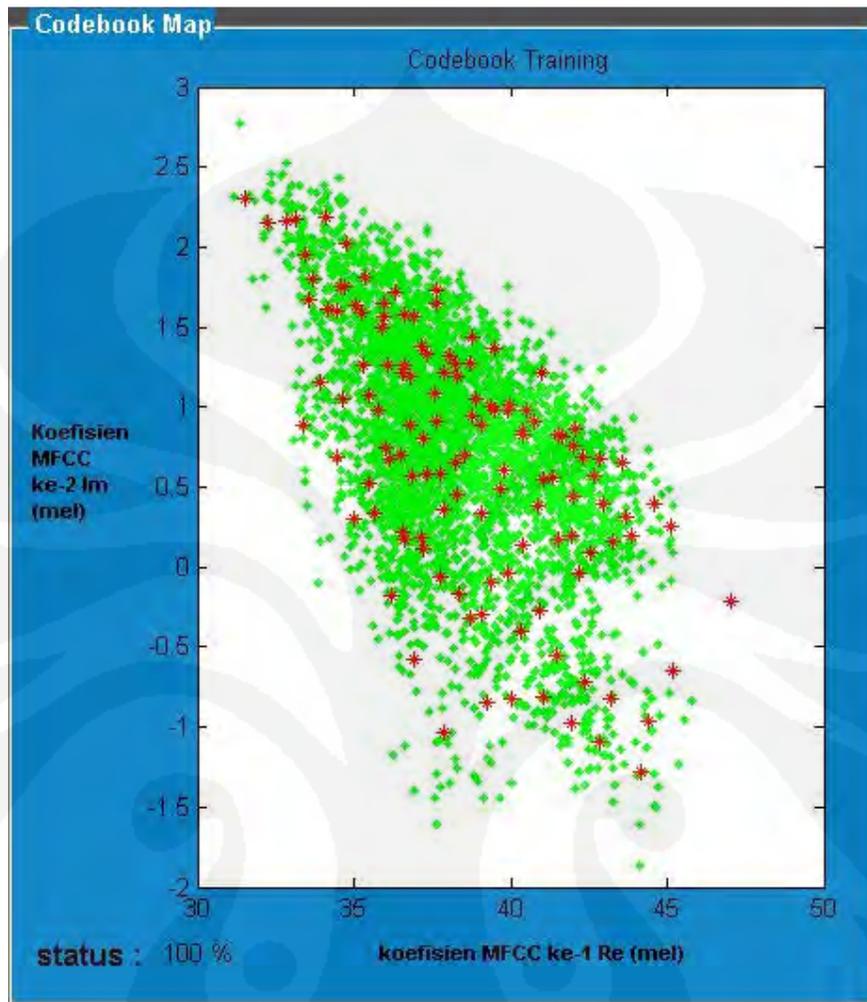
Hasil uji coba dilihat pada Tabel 4.2 sampai Tabel 4.6. Semakin besar ukuran *codebook*, maka jumlah *cluster* yang terbentuk semakin banyak, sehingga terbentuk *centroid* yang semakin banyak pula. Semakin banyak jumlah *centroid*, maka panjang *codeword* akan semakin besar dan tentunya *codeword* ini akan semakin mewakili sinyal masukan aslinya. Selain itu, ukuran *cluster* akan semakin kecil yang dapat menyebabkan sistem mengkuantisasi vektor data lebih teliti karena jarak antar-vektor data serta antara vektor data dengan *centroid* semakin kecil. Untuk lebih jelasnya, dapat dilihat pada gambar 4.1 – 4.3 berikut ini.



Gambar 4.1 Codebook 32 bit



Gambar 4.2 Codebook 64 bit



Gambar 4.3 Codebook 128 bit

Berdasarkan uraian di atas, dapat disimpulkan bahwa semakin besar ukuran *codebook*, maka tingkat akurasi program semakin meningkat. Namun, pada penelitian ini, persentase akurasi tidak mengalami kenaikan atau penurunan atau stabil dengan ukuran *codebook* yang semakin meningkat. Hal tersebut dapat dilihat pada Tabel 4.2 sampai Tabel 4.6. Hal ini disebabkan karena sinyal suara memiliki rentang data yang cukup besar, sehingga cukup sulit untuk membandingkan data-data yang secara kasat mata terlihat mirip. Selain itu, dari gambar di atas juga terlihat bahwa terlihat beberapa *centroid* yang tidak melingkupi daerah persebaran vektor data, yang menyebabkan beberapa *centroid* tersebut tidak mewakili sinyal masukan.

Universitas Indonesia

Selain itu, semakin besar ukuran *codebook*, semakin lama pula waktu pemrosesannya karena semakin banyak jumlah *centroid* yang akan dicari. Oleh karena itu pada penelitian ini hanya digunakan ukuran *codebook* 32, 64 dan 128. Menurut beberapa penelitian sinyal suara ^[2], untuk *codebook* dengan ukuran lebih besar dari 128, hasil yang didapatkan tidak terlalu berbeda dengan hasil yang didapatkan pada uji coba dengan ukuran *codebook* 128. Namun waktu pemrosesan menjadi jauh lebih lama sehingga menjadi tidak efisien. Oleh karena itulah hanya digunakan *codebook* berukuran 32, 64, dan 128 dalam pengujian. Karena untuk mendapatkan hasil uji coba yang efektif, perlu diperhatikan pula waktu pemrosesan yang semakin cepat akan menjadi lebih efektif, dengan persentase akurasi yang tinggi.

Ketika sampel baru diuji dan dilakukan pencarian letak *centroid-centroidnya*, maka pencarian tersebut mengacu kepada letak *centroid* yang terdapat pada database *codebook*. Pada jenis penyakit yang sama dengan sampel yang berbeda pun dapat terjadi kesalahan dalam pendeteksian penyakitnya. Hal ini dikarenakan distorsi pada masing-masing sampel tidak selalu sesuai dengan distorsi yang terdapat pada *database* untuk penyakit yang sama. Sehingga terdapat kemungkinan letak sampel baru ini lebih mendekati letak *centroid* jenis penyakit lainnya sehingga kombinasi urutan *centroid* yang dihasilkan agak berbeda dengan penyakit yang sebenarnya.

4.4.2 Analisis Pengaruh Variasi Jumlah Database

Parameter berikutnya yang dianalisa adalah jumlah *database*. Semakin banyak sampel yang dimasukkan ke dalam *database* untuk masing-masing penyakit, maka semakin banyak variasi yang dimiliki untuk masing-masing penyakit sehingga proses *recognition* akan lebih mudah karena dengan banyaknya *database* untuk masing-masing penyakit, masalah kemiripan atau perbedaan satu penyakit terhadap penyakit lainnya akan berkurang. Karena jumlah *database* yang lebih banyak maka secara tidak langsung akan menambah karakteristik label yang tersimpan pada *database*. Oleh karena itu, dengan banyaknya sampel yang dimasukkan atau sampel yang dijadikan *database*, maka karakteristik yang tadinya belum ada pada saat proses pembentukan *database*

dilakukan dengan jumlah database yang lebih kecil, akan terpenuhi dengan jumlah *database* yang lebih besar^{[1][16]}.

Dengan banyaknya variasi sampel untuk masing-masing penyakit pada database, maka letak *centroid* untuk tiap-tiap penyakit juga akan semakin bervariasi, sehingga penyakit-penyakit yang mirip sekalipun akan memiliki letak *centroid* yang berbeda koordinatnya, tidak terlalu berdekatan apalagi berhimpit. Dengan demikian proses pengenalan akan semakin mudah dan tingkat keberhasilan yang dicapai akan semakin tinggi.

Berdasarkan penelitian di atas secara keseluruhan, semakin besar jumlah *database*, maka semakin tinggi tingkat akurasi yang dicapai. Persentase akurasi rata-rata meningkat 10% - 40%. Hal ini dapat dilihat pada Tabel 4.1 sampai Tabel 4.3. Pada penelitian ini, jumlah sampel yang dijadikan *database* divariasikan sebanyak 5 (lima) dan 10 (sepuluh) buah sampel. Penggunaan jumlah *database* yang sedikit ini disebabkan karena keterbatasan sampel yang ada.

Kesalahan dalam pendeteksian terjadi karena adanya kemiripan karakteristik antara sampel penyakit satu dengan sampel penyakit lainnya. Karena suara abnormal pada proses pernafasan menghasilkan suara yang memiliki kemiripan karakteristik satu sama lain walaupun untuk jenis penyakit yang berbeda. Karena program hanya memproses perwakilan sinyal dari masing-masing sampel, maka kesalahan pendeteksian tak dapat dihindari.

4.4.3 Analisis Pengaruh Hubungan Ukuran *Codebook* dan Jumlah *Database*

Dalam merancang suatu sistem, dibutuhkan parameter-parameter yang dapat menunjang sistem menjadi optimal. Untuk mencapai kondisi tersebut, maka diperlukan kombinasi parameter-parameter yang telah dilakukan pengujian sebelumnya. Setelah dilakukan uji coba terhadap sampel-sampel pada variasi ukuran *codebook* dan jumlah *database*, maka dapat dilihat pada Tabel 4.5 dan Tabel 4.6 variasi parameter yang mana yang memberikan hasil yang optimal.

Seperti yang telah dijelaskan sebelumnya, semakin besar ukuran *codebook* yang digunakan, maka kinerja sistem menjadi lebih optimal. Namun waktu pemrosesan data menjadi lebih lama. Jika dihubungkan dengan parameter

Universitas Indonesia

tersebut, maka semakin banyak jumlah *database*, semakin banyak pula vektor data yang mengalami proses vektor kuantisasi. Hal ini juga sangat berpengaruh pada waktu pemrosesan yang menjadi semakin bertambah. Namun, hal ini tidak menjadi suatu kendala karena jumlah *database* yang semakin banyak akan berpengaruh positif terhadap sistem.

Berdasarkan penjelasan di atas, maka, untuk menciptakan kombinasi parameter yang optimal bagi sistem, yaitu dengan menghasilkan tingkat akurasi yang paling tinggi dan waktu pemrosesan yang lebih cepat dalam suatu percobaan. Berdasarkan Tabel 4.5, nilai rata-rata tingkat akurasi maksimum adalah sebesar 93,33%. Namun, karena terdapat nilai akurasi yang sama untuk tiga parameter, maka ukuran *codebook* yang optimal pada penelitian ini adalah 32, karena waktu yang dibutuhkan selama pemrosesan data lebih cepat dibandingkan dengan ukuran *codebook* 64 dan 128. Jumlah *database* yang optimal adalah 10 (sepuluh) buah, karena semakin banyak jumlah *database*, semakin banyak variasi karakteristik dari sistem sehingga dapat melakukan identifikasi lebih akurat terhadap sampel baru yang diujikan.

BAB V

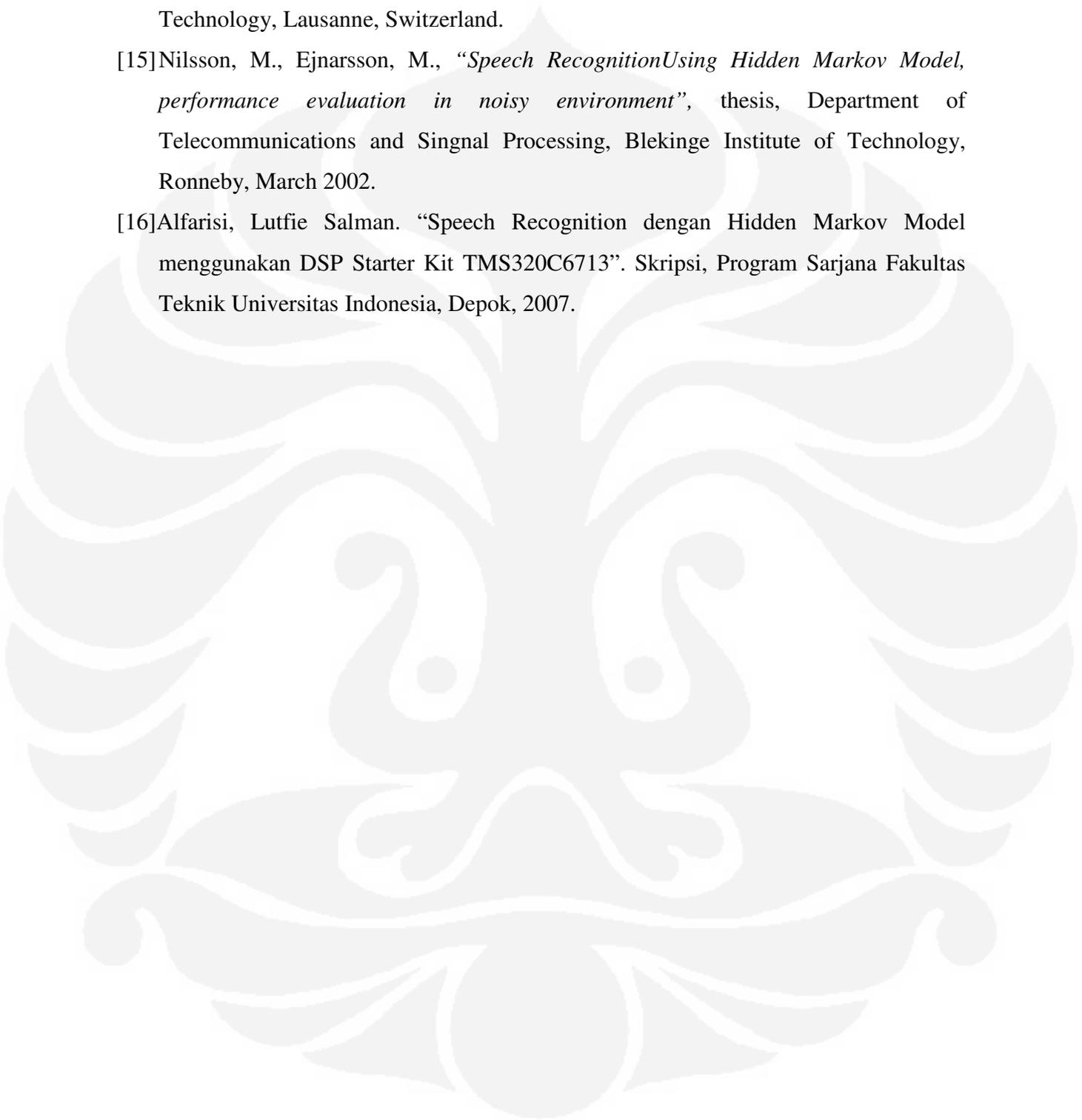
KESIMPULAN

Berdasarkan hasil uji coba dan analisis yang telah dilakukan, maka dapat diambil beberapa kesimpulan antara lain sebagai berikut.

1. Nilai akurasi rata-rata keseluruhan rangkaian uji coba yang dicapai adalah antara 70% sampai 93,33%.
2. Semakin besar ukuran *codebook*, maka tidak mempengaruhi tingkat akurasi yang dicapai sistem dalam mendeteksi penyakit paru-paru.
3. Semakin besar jumlah *database*, semakin besar pula tingkat keberhasilan sistem dalam mendeteksi penyakit paru-paru.
4. Pada penelitian ini, ukuran *codebook* yang optimal adalah 32 dan jumlah *database* yang optimal sebesar 10 (sepuluh) buah.
5. Semakin banyak jumlah *codebook* maka waktunya semakin lama

DAFTAR ACUAN

- [1] Hartaman, Muhammad Rizky. “Rancang Bangun Sistem Pengenalan Penyakit Jantung dengan Metode Hidden Markov Model”. Skripsi, Program Sarjana Fakultas Teknik Universitas Indonesia, Depok, 2009.
- [2] Sibuea, Herdin; Panggabean, Marulam M & Gultom,S.P. (2005). *Ilmu Penyakit Dalam*.Jakarta. PT Asdi Mahasetya.
- [3] Batri, Nadim, “*Robust Spectral Parameter Coding in Speech recognition*”, thesis, Department of Electrical Engineering, McGill University, Montreal, Canada, 1998.
- [4] Kaelin, Mark, “Auscultation: Listening to Determine Dysfunction “. *Professionalization of Exercise Physiology online, An international electronic journal for exercise physiologists*. ISSN 1099-5862, Vol 4 No 8 August, 2001
- [5] MVS Pulmonary Auscultation, <http://sprojects.mmi.mcgill.ca/mvs/mvsteth.htm>
- [6] Pusat Penerbitan Departemen Ilmu Penyakit Dalam FKUI. (2006). *Buku Ajar Ilmu Penyakit Dalam*. Jakarta : Author
- [7] Asupan Keperawatan pada Klien Asma, <http://nursingbegin.com/tag/askep-asma/> (26 April 2009)
- [8] Asma Bronkhial, <http://doctorology.net/?p=144>
- [9] Emphysema, <http://www.totalkesehatananda.com/emphysema.html>
- [10] Wheeze, <http://en.wikipedia.org/wiki/wheeze> (15 Maret 2007)
- [11] Pleurisy, <http://www.totalkesehatananda.com/pleurisy.html>
- [12]Liu, Z., Yin, Q., Zhang, W., “*A Speaker Identification and Verification System*”, EEL6586 Final Project, 2002.
- [13]Luczkovich, Joseph J., Daniel, Hal J. III, dan Sprague, Mark W., "*Characterization of critical spawning habitats of weakfish, spotted seatrout and red drum in Pamlico Sound using hydrophone surveys*", Final Report and Annual Performance Report, Grant F-62-1, Grant F-62-2, Institute for Coastal and Marine Resources, Department of Biology, Department of Physics East Carolina University, Greenville, NC 27858, 1998.

- 
- [14] Do, M.N., “*An Automatic Speaker Recognition System*”, Digital Signal Processing Mini-Project, Audio Visual Communications Laboratory Swiss Federal Institute of Technology, Lausanne, Switzerland.
- [15] Nilsson, M., Ejnarsson, M., “*Speech Recognition Using Hidden Markov Model, performance evaluation in noisy environment*”, thesis, Department of Telecommunications and Signal Processing, Blekinge Institute of Technology, Ronneby, March 2002.
- [16] Alfari, Lutfie Salman. “Speech Recognition dengan Hidden Markov Model menggunakan DSP Starter Kit TMS320C6713”. Skripsi, Program Sarjana Fakultas Teknik Universitas Indonesia, Depok, 2007.

DAFTAR PUSTAKA

Diponegoro, Arman Djohan. “Analisis penentuan jenis kawanan ikan berdasarkan deteksi fasa pantulan gelombang akustik dan penerapan Hidden Markov Model”. 2006. Disertasi, Program Pascasarjana Fakultas Ilmu Kelautan dan Perikanan Institut Pertanian Bogor, Bogor, 2006.

Amin, Ahmad Mujadid. “Simulasi dan Analisis Program Speech To Text Menggunakan Metode Hidden Markov Model”. Skripsi, Program Sarjana Fakultas Teknik Universitas Indonesia, Depok, 2007.

Alfarisi, Lutfie Salman. “Speech Recognition dengan Hidden Markov Model menggunakan DSP Starter Kit TMS320C6713”. Skripsi, Program Sarjana Fakultas Teknik Universitas Indonesia, Depok, 2007.

Saaddatuddaroin. “Perancangan Perangkat Lunak Sendor Tsunami dengan Teknik Hidden Markov Model”. Skripsi, Program Sarjana Fakultas Teknik Universitas Indonesia, Depok, 2009.

Lestari, Afita Putri. “Rancang Bangun Pengenalan Penyakit Darah Menggunakan Hidden Markov Model”. Skripsi, Program Sarjana Fakultas Teknik Universitas Indonesia, Depok, 2008.