



UNIVERSITAS INDONESIA

**ANALISA PENGGUNAAN *VOICEBANK* BERFORMAT *KVK*
DALAM APLIKASI *SPEECH SYNTHESIZER* DAN *SINGING*
SYNTHESIZER BAHASA INDONESIA**

SKRIPSI

DIMAS TRIBUDI WIRIAATMADJA

0606073865

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
DESEMBER 2010**



UNIVERSITAS INDONESIA

**ANALISA PENGGUNAAN *VOICEBANK* BERFORMAT *KVK*
DALAM APLIKASI *SPEECH SYNTHESIZER* DAN *SINGING*
SYNTHESIZER BAHASA INDONESIA**

SKRIPSI

Diajukan sebagai salah satu syarat memperoleh gelar sarjana

DIMAS TRIBUDI WIRIAATMADJA

0606073865


**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
DESEMBER 2010**

HALAMAN PERNYATAAN ORISINALITAS

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

Nama : Dimas Tribudi Wiriaatmadja

NPM : 0606073865

Tanda Tangan : 

Tanggal : 17 DESEMBER 2010

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Dimas Tribudi Wiriaatmadja
NPM : 0606073865
Program Studi : Teknik Elektro
Judul Skripsi : Analisa Penggunaan *Voicebank* Berformat KVK
dalam Aplikasi *Speech Synthesizer* dan *Singing
Synthesizer* Bahasa Indonesia

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk mata kuliah Skripsi pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Prof. Dr. Ir. Dadang Gunawan, M.Eng.



Penguji : Dr. Abdul Halim, M.Eng



Penguji : Filbert Hilman Juwono S.T., M.T



Ditetapkan di : Depok

Tanggal : 30 Desember 2010

UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kehadiran Allah SWT, karena atas segala rahmat dan penyertaan-Nya saya dapat menyelesaikan skripsi ini. Saya menyadari bahwa skripsi ini tidak akan terselesaikan tanpa bantuan dari berbagai pihak. Oleh karena itu, saya mengucapkan terima kasih kepada :

1. Bapak Prof. Ir. Dadang Gunawan, M.Eng. Ph.D, selaku pembimbing yang membantu memberikan arahan dan nasihat sehingga saya dapat menyelesaikan skripsi ini;
2. Para peneliti sebelum ini yang juga memberikan sumber bacaan yang banyak bagi saya;
3. Mr. Ameya selaku pembuat aplikasi UTAU.
4. Komunitas UTAU di seluruh dunia yang turut serta dalam membantu perkembangan UTAU sampai saat ini.
5. Adik saya Lulu dan Farid yang telah bersedia membantu saya dalam proses pembuatan *Voicebank* "LULU3".
6. Kedua orang tua saya yang selalu mensupport saya dalam menyelesaikan skripsi ini
7. Teman teman yang telah bersedia menjadi responden dalam simulasi yang saya lakukan
8. Dan seluruh Sivitas akademik Departemen Teknik Elektro yang tidak dapat saya sebutkan satu persatu.

Akhir kata, semoga Allah SWT berkenan membalas kebaikan semua pihak yang telah membantu. Semoga skripsi ini bermanfaat bagi perkembangan ilmu pengetahuan.

Depok, 17 Desember 2010

Dimas Tribudi Wiriaatmadja

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

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

Nama : Dimas Tribudi Wiriaatmadja
NPM : 0606073865
Program studi : Teknik Elektro
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis karya : Skripsi

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

**ANALISA PENGGUNAAN VOICEBANK BERFORMAT KVK
DALAM APLIKASI SPEECH SYNTHESIZER DAN SINGING
SYNTHESIZER BAHASA INDONESIA**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta sebagai pemegang Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 17 Desember 2010

Yang menyatakan



Dimas Tribudi Wiriaatmadja

ABSTRAK

Nama : Dimas Tribudi Wiriaatmadja
Program studi : Teknik Elektro
Judul : Analisa Penggunaan *Voicebank* Berformat KVK dalam Aplikasi *Speech Synthesizer* dan *Singing Synthesizer* Bahasa Indonesia

Bahasa Indonesia mempunyai banyak kata yang mengandung unsur KVK. Akan tetapi selama ini unsur KVK belum dipakai sebagai salah satu unsur vital dari *voicebank* yang diperuntukan untuk membuat ucapan dan nyanyian sintesis berbahasa Indonesia. Dalam skripsi ini dibuat sebuah *voicebank* yang dinamakan *voicebank* “LULU3” yang terdiri dari 1765 *file* suara yang berformat V, KV, VK, dan KVK. *Voicebank* ini direkam dengan bantuan seorang penyanyi wanita terlatih berkebangsaan Indonesia. Selanjutnya simulasi kemudian dilakukan dimana beberapa suara sintesis berbahasa Indonesia dibuat dengan menggunakan beberapa *synthesizer* yang menggunakan *voicebank* “LULU3”. *Speech synthesizer* yang digunakan dalam skripsi ini adalah *speech synthesizer* beralgoritma SOLA-Resampling, dan *speech synthesizer* berbasis UTAU. Sedangkan *singing synthesizer* yang digunakan dalam skripsi ini adalah *singing synthesizer* UTAU. Dari survey yang dilakukan terhadap sejumlah responden yang terdiri dari mahasiswa departemen teknik elektro universitas Indonesia, sebagian besar responden menyatakan bahwa suara sintesis yang dibuat dengan *voicebank* “LULU3” yang mempunyai unsur KVK terdengar lebih jelas daripada suara sintesis yang dibuat dengan *voicebank* “LULU3” yang tidak mempunyai unsur KVK.

Kata kunci : suara sintesis, *singing synthesizer* bahasa Indonesia, *voicebank* bahasa Indonesia, UTAU, *speech synthesizer* bahasa Indonesia, KVK

ABSTRACT

Name : Dimas Tribudi Wiriaatmadja
Study program: Electrical Engineering
Title : Analysis of KVK Formatted Voicebank Usage in Bahasa Indonesia Speech Synthesizer and Singing Synthesizer Application

Indonesian language has a lot of words that's containing the elements of KVK. However, until now the element of KVK is not used as a vital element in the voicebanks that's devoted to Indonesian speech synthesis and singing synthesis. In this paper, a voicebank named "LULU3" that's consisting of 1765 V ,KV, VK, and KVK pre-recorded speech was made. This voicebank was recorded with the help of a trained female singer with Indonesian nationality. Furthermore, the simulation was then performed in which several voices in Indonesian language is made synthetically using several synthesizers that's use voicebank "LULU3". The speech synthesizers used in this paper are speech synthesizer with SOLA-resampling algorithm, and UTAU based speech synthesizer. The singing synthesizer used in this paper is singing synthesizer UTAU. From a survey conducted on a number of respondents consisting of university students from electrical engineering department of University of Indonesia, the majority of respondents stated that the synthetic voice that's created with voicebank "LULU3" which has elements of KVK sounds more clearly than the synthetic voice created with voicebank "LULU3" which does not have the elements of KVK.

Keyword : synthetic voice, Indonesian *singing synthesizer*, Indonesian *voicebank*, UTAU, Indonesian *speech synthesizer*, KVK.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
LEMBAR PERSETUJUAN.....	iii
HALAMAN PENGESAHAN.....	iv
UCAPAN TERIMA KASIH.....	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	vi
ABSTRAK	vii
<i>ABTRACT</i>	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	ix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan Penulisan.....	5
1.3 Pembatasan Masalah.....	5
1.4 Metodologi Penulisan	6
1.5 Sistematika Penulisan	7
BAB 2 SUARA, SPEECH SYNTHESIS & TATA BAHASA INDONESIA.....	9
2.1 Suara.....	9
2.1.1 Frekuensi	9
2.1.2 Periode.....	9
2.1.3 Amplitudo.....	9
2.2 Suara Manusia	9
2.2 Suara Digital.....	11
2.2.1 Channel.....	11
2.2.2 Frekuensi Sampling.....	11
2.2.3 <i>Bitdepth</i>	11
2.2.3 kompresi	12
2.3 <i>Speech Synthesis</i>	12
2.3.1 Contanetative Synthesis.....	13
2.3.1.1 <i>Diphone Synthesis</i>	13
2.3.1.2 <i>Unit Selection Synthesis</i>	13
2.3.2 <i>Formant Synthesis</i>	13
2.4 Tata Bahasa Indonesia.....	14
2.4.1 Klasifikasi Bunyi.....	14
2.4.1.1 Bunyi Tunggal.....	14
2.5.2.2 Bunyi Ganda.....	14
2.5.1 Fonem dan Fonemisasi.....	14
BAB 3 PERANCANGAN <i>VOICEBANK</i> DAN <i>SPEECH SYNTHESIZER</i>	16
3.1 Pembuatan <i>Voicebank</i>	16
3.1.1 Perekaman.....	16
3.1.2 Pendeteksian Fonem.....	17
3.1.3 Pengolahan Data Rekaman.....	19
3.1.3.1 Pemisahan Suara.....	20

3.1.3.2 Pendeteksian frekuensi <i>Pitch</i>	20
3.1.3.3 Pencatatan Durasi dan Pitch Rata Rata.....	23
3.2 Sistem <i>Speech synthesizer</i>	25
3.2.1 <i>GUI Speech synthesizer</i>	25
3.2.1.1 <i>Modul Phoneme Synthesizer</i>	26
3.2.1.2. <i>Modul Phonemes Combiner</i>	27
3.2.1.3 <i>Modul Words Combiner</i>	28
3.2.2 <i>Algoritma Speech synthesizer</i>	29
3.2.2.1 <i>Algoritma Phoneme Synthesizer 1</i>	30
3.2.2.2 <i>Algoritma Phoneme Synthesizer2</i>	35
3.3 Sistem <i>Singing synthesizer</i>	42
 BAB 4 SIMULASI DAN ANALISA	 48
4.1 Simulasi Penggunaan Fonem KVK pada <i>Speech Synthesizer</i>	48
4.2 Simulasi Penggunaan Fonem KVK pada <i>Singing synthesizer</i>	52
 BAB 5 KESIMPULAN.....	 61
 DAFTAR REFERENSI	 62
 DAFTAR PUSTAKA	 64
 LAMPIRAN A: Source Code <i>Speech synthesizer 1 dan 2</i>	 65
 LAMPIRAN B: Komposisi <i>Voicebank “LULU3”</i>	 68
 LAMPIRAN C: Tabel Pendeteksian Fonem Bahasa Indonesia	 73
 LAMPIRAN D: Form Survey	 77

DAFTAR GAMBAR

Gambar 2.1 Jenis Suara Manusia Berdasarkan Frekuensi Pitchnya.....	10
Gambar 2.2 Suara Digital.....	11
Gambar 3.1 File Suara BasBisBus.wav Hasil Rekaman	20
Gambar 3.2 File suara Bas.wav yang Diambil dari File BasBisBus.wav	20
Gambar 3.3 Tampilan GUI pada UTAU	21
Gambar 3.4 Tampilan Project Properties pada UTAU	22
Gambar 3.5 Perintah Select Multi pada Proses Pendeteksian Pitch	22
Gambar 3.6 Frekuensi rata rata yang Sudah Dideteksi UTAU	23
Gambar 3.7 File Ba.wav Dilihat dengan Software Audacity.....	24
Gambar 3.8 Source Code Penyimpanan Durasi dan Pitch	24
Gambar 3.9 GUI Speech synthesizer 2	25
Gambar 3.10 Modul Phoneme Synthesizer	26
Gambar 3.11 Tampilan Modul Phonemes Combiner	27
Gambar 3.12 Ilustrasi Pensintesisan Suara “I” dan “bu”	28
Gambar 3.13 Ilustrasi Penggabungan Suara “I” dan “bu”	28
Gambar 3.14 Modul Words Combiner.....	29
Gambar 3.15 Ilustrasi Pensintesisan “Ibu Budi” dengan Words combiner	29
Gambar 3.16 Source Code untuk Menggabungkan 5 File Wav.....	30
Gambar 3.17 Flowchart Speech synthesizer 1	31
Gambar 3.18 Tampilan Speech Engine UTAU pada Command Prompt.....	32
Gambar 3.19 Perintah untuk Mengubah Pitch pada Speech Engine UTAU	32
Gambar 3.20 Flowchart Speech Synthesizer 2	35
Gambar 3.21 Algoritma SOLA	37
Gambar 3.22 Ilustrasi Peningkatan Framerate	38
Gambar 3.23 Proses Pitch Shifting	39
Gambar 3.24 Pembuatan Nyanyian Sintesis “Padamu Negri” pada UTAU.....	43
Gambar 3.25 Ilustrasi Parameter Offset dalam UTAU	44
Gambar 3.26 Ilustrasi Preutterance	44
Gambar 3.27 Ilustrasi Preutterance dan Overlap.....	45
Gambar 3.28 Penggunaan Preutterance dan Overlap	46
Gambar 3.29 GUI pada UTAU untuk Mengubah Parameter File Suara	46
Gambar 4.1 Form Isian Uji Speech Synthesizer	49
Gambar 4.2 Form Simulasi Singing Synthesizer	53
Gambar 4.3 GUI KVKhelper	58
Gambar 4.4 Ilustrasi Trimming pada Voiced Sound.....	59
Gambar 4.5 File Suara Voiced Sound Setelah Dilakukan Trimming	59
Gambar 4.6 File “jan_s.wav” Diolah Diperpendek dengan Menggunakan algoritma SOLA	60
Gambar 4.7 File “jan_s.wav” Diolah Diperpendek dengan Menggunakan Speech Engine UTAU	60
Gambar 4.8 File “jan_s.wav” Diolah Diperpendek dengan Cara Trimming Manual.....	60
Gambar 4.9 File Suara “jan.wav”	60

DAFTAR TABEL

Tabel 2.1 Daftar Nilai Frekuensi Pitch untuk Setiap Nada.....	10
Tabel 3.1 Daftar Fonem Vokal Bahasa Indonesia.....	17
Tabel 3.2 Cuplikan Daftar Fonem Berformat VK.....	18
Tabel 3.3 Cuplikan Daftar Fonem Berformat KV.....	19
Tabel 3.4 Cuplikan Daftar Fonem Berformat KVK.....	19
Tabel 3.5 Daftar Pitch dan Durasi dari Kalimat “Dimana Budi Tinggal” yang diucapkan oleh penyedia suara voicebank “LULU3”	33
Tabel 3.6 Perbandingan antara <i>Pitch</i> Suara Asli dengan <i>Pitch</i> Suara Sintesis Hasil Pensintesisan dengan <i>Phoneme Synthesizer</i> pada <i>Speech Synthesizer 1</i>	34
Tabel 3.7 Perbandingan antara Durasi Suara Asli dengan Durasi Suara Sintesis Hasil Pensintesisan dengan <i>Phoneme synthesizer</i> pada <i>Speech synthesizer1</i>	34
Tabel 3.8 Perbandingan antara <i>Pitch</i> Suara Asli dengan <i>Pitch</i> Suara Sintesis Hasil Pensintesisan dengan <i>Phoneme Synthesizer</i> pada <i>Speech Synthesizer 1</i>	40
Tabel 3.9 Perbandingan antara Durasi Suara Asli dengan Durasi Suara Sintesis Hasil Pensintesisan dengan <i>Phoneme synthesizer</i> pada <i>Speech synthesizer2</i>	40
Tabel 3.10 Perbandingan antara <i>Pitch</i> Suara Asli dengan <i>Pitch</i> Suara Sintesis Hasil Pensintesisan dengan <i>Speech Synthesizer 2</i>	41
Tabel 3.11 Perbandingan antara Durasi Suara Asli dengan Durasi Suara Sintesis Hasil Pensintesisan dengan <i>Speech synthesizer2</i>	41
Tabel 3.12 Perbandingan frekuensi <i>Pitch</i> Nada dan Frekuensi <i>Pitch</i> Suara yang Dihasilkan UTAU	47
Tabel 3.12 Perbandingan Durasi Nada Seharusnya dan Durasi File Suara yang Dihasilkan UTAU	47
Tabel 4.1 Perbandingan Pilihan Responden atas Kejelasan Suara antara yang Dibuat dengan Menggunakan Metode 1 dengan yang Dibuat Menggunakan Metode 2.....	49
Tabel 4.2 Perbandingan Pilihan Responden atas Kejelasan Suara antara yang Dibuat dengan Menggunakan Metode 3 dengan yang Dibuat Menggunakan Metode 4.....	50
Tabel 4.3 Perbandingan Pilihan Responden atas Kejelasan Suara antara yang Dibuat dengan Menggunakan Metode 2 dengan yang Dibuat menggunakan Metode 3	51
Tabel 4.4 Data Pendapat Subjektif Responden Atas Kualitas Nyanyian yang dibuat dengan Metode 3 dan Metode 1	54
Tabel 4.5 Data Pendapat Subjektif Responden Atas Kualitas Nyanyian yang dibuat dengan metode 3 dan metode 1	55
Tabel 4.6 Data Pendapat Subjektif Responden Atas Kualitas Nyanyian yang Dibuat dengan Metode 2 dan 1	56
Tabel 4.7 Data Pendapat Subjektif Responden Atas Kualitas Nyanyian yang Dibuat dengan Metode 3 dibandingkan dengan metode 2 atau 1	56

DAFTAR ISTILAH

Audacity : Software yang digunakan dalam skripsi ini untuk pengolahan rekaman, terutama dalam proses trimming dan pencatatan durasi

Difon: Satuan bunyi yang terdiri dari dua fon

Fon: Satuan bunyi terkecil

Fonem: Satuan bunyi

Frekuensi Pitch: Ukuran tinggi rendahnya nada.

Frekuensi Sample: Banyaknya nilai sample yang dimainkan dalam satu detik dalam sebuah file suara

KV: Format fonem yang terdiri dari konsonan kemudian diikuti oleh vokal

KVK: Format fonem yang terdiri dari konsonan kemudian diikuti oleh vokal dan diakhiri oleh konsonan

SOLA: Algoritma untuk melakukan time scaling, SOLA sendiri merupakan singkatan dari *synchronous overlap and add*

Resampling: Algoritma untuk melakukan pitch shifting

Trifon: Satuan bunyi yang terdiri dari tiga fon

UTAU: Software aplikasi singing synthesizer yang dikembangkan Mr. Ameya sejak tahun 2008.

V: Format fonem yang terdiri dari vokal saja

VK: Format fonem yang terdiri dari vokal kemudian diikuti oleh konsonan.

Voicebank: Kumpulan *prerecorded* speech yang digunakan untuk membuat suara sintesis

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Speech synthesizer adalah sebuah sistem dimana sistem tersebut dapat menghasilkan keluaran yang berupa suara manusia. *Speech synthesizer* memiliki banyak aplikasi, namun aplikasi yang populer adalah aplikasi *singing synthesizer* dan *text to speech*. *Singing synthesizer* adalah sebuah sistem yang dapat menghasilkan suara manusia dengan nada yang dapat diatur sehingga menyerupai suara manusia yang sedang bernyanyi. Sedangkan *Text to speech* adalah aplikasi dimana *input*-nya berupa teks dan *output*-nya berupa ucapan. *Speech synthesizer* dan *singing synthesizer* mempunyai fungsi yang bermacam macam diantaranya adalah

a. *Artificial singer*

Dengan *artificial singer*, Seorang pencipta lagu dapat mensimulasikan lagu yang mereka buat tanpa harus menggunakan jasa seorang penyanyi profesional.

b. Pengabadi Suara

Dengan *singing synthesizer*, suara seorang penyanyi tetap dapat didengar walaupun penyanyi tersebut sudah meninggal

c. *artificial seiyuu/ voice actor*

Perkembangan multimedia yang begitu pesat mensyaratkan para pengembang aplikasi dan game untuk bisa menciptakan sesuatu yang interaktif. Disini peran dari suara manusia sangat dibutuhkan. Dengan *artificial voice actor*, seorang pengembang aplikasi dapat menghasilkan suara manusia dengan intonasi dan nada apapun tanpa harus ke studio rekaman dan menyewa jasa seorang *voice actor*

d. Sebagai *speech engine* dari aplikasi *text to speech*

Speech synthesizer adalah salah satu bagian penting dari sistem *text to speech*. bagian inilah yang mengubah fonem fonem kedalam suara.

Sejarah mencatat penelitian tentang *speech synthesizer* ini sudah dimulai sejak abad ke 13 di Jerman, pada waktu itu filsuf jerman albertus magnus dan ilmuwan inggris berhasil membuat kepala besi yang dapat berbicara, namun sayang tidak ada dokumen yang memadai tentang divais ini. Penelitian lain tentang *speech synthesis* yang tercatat dalam sejarah adalah penelitian pada tahun 1759 yang dilakukan Christian kratzenstein [9]. Kratzenstein berhasil menciptakan rongga resonansi yang divibrasikan sehingga dapat menghasilkan 5 bunyi vokal yaitu a, o, e, i, u. Di tahun 1791 Wolfgang von Kempelen dalam tulisannya "*acoustic-mechanical speech machine*" berhasil membuat sebuah sistem yang dapat menghasilkan suara dari vokal dan konsonan. Pada tahun 1873 Charles Wheatstone membuat sebuah alat bernama "*speaking machine*" yang merupakan rekonstruksi dari desain buatan Kempelen. Pada tahun 1930. Laboratorium Bell mengembangkan "Vocoder", sebuah alat yang kemudian disempurnakan oleh Homer Dudley menjadi Voder. Voder ini di tampilkan pada tahun 1939 di New York world's fair. [9]

Saat ini perkembangan teknologi *speech synthesizer* sudah sangat berkembang pesat. Perkembangan pesat ini ditenggarai karena teknologi komputer yang sudah sangat maju. Sehingga sinyal suara dapat diolah secara digital. Penelitian modern tentang *speech synthesizer* yang terpublikasi cukup baik salah satunya adalah MBROLA. Mbrola adalah sebuah *project text to speech* yang menggunakan teknik *diphone synthesis*. Mbrola dipelopori oleh *Faculté Polytechnique de Mons, Belgia*. *Project* lain tentang *text to speech* yang terpublikasi dengan baik adalah E-speak, salah satu *project text to speech* yang *open source* yang menggunakan teknik *formant synthesis*. Teknologi e-speak ini dipakai oleh situs *google translate* sebagai sistem *text to speech* sementara mereka untuk menyuarakan kata kata yang ditulis di google translate pada beberapa bahasa. Penelitian yang bersifat *open source* lain adalah euler *text to speech* yang dimulai di Prancis. Untuk *singing synthesizer project* yang bersifat *open community* dan cukup populer adalah UTAU. UTAU dimulai oleh Ameya di Jepang. *Project* ini tidak *open source*, namun dapat dikostumisasi. sehingga setiap orang dapat menambah *database* suaranya masing masing. Untuk penelitian

lainnya yang bersifat tertutup sudah cukup banyak dilakukan oleh pihak-pihak komersil, seperti Microsoft, AT&T, Yamaha, dan lain-lain

Singing synthesizer sedikit berbeda dengan *speech synthesizer*. Dalam *singing synthesizer* nada yang dikeluarkan harus mempunyai frekuensi yang sesuai dengan frekuensi *pitch* notasi music. Selain itu *singing synthesizer* harus bisa dengan jelas menghasilkan ucapan dari nada e2-C6 yang merupakan nada yang dapat dihasilkan oleh manusia. Penelitian *singing synthesizer* yang terpublikasi dengan baik diantaranya adalah

- a. “*A singing voice synthesis system based on sinusoidal modeling*” yang dilakukan oleh Macon, M.W. Jensen-Link, L. Oliverio, J. Clements, M.A. George, E.B dari Georgia Institute of Technology pada tahun 1997
- b. “*Singing voice synthesis/analysis*” oleh Youngmoo Edmund Kim dari Massachusetts Institute of Technology pada tahun 2003.
- c. “*Spectral Processing of singing voice*” oleh Alex Loscos, 2007 Universitas Pompeu Fabra (disponsori oleh Yamaha)
- d. “*Singing Driven Interfaces for Sound Synthesizers*” oleh Jordi Jane, 2008 Universitas Pompeu Fabra (disponsori oleh Yamaha)

Di Indonesia, penelitian tentang *Speech synthesizer* khususnya untuk aplikasi *text to speech* yang pernah dilakukan diantaranya :

- a. Pada tahun 2000 AA Arman dari teknik elektro ITB bekerja sama dengan *Faculté Polytechnique de Mons* di Belgia untuk membuat *database diphone id1* dan kemudian mengaplikasikannya sebagai program *text to speech* dengan menggunakan *speech engine Mbrola*
- b. Pada tahun 2002 Octoni Symbolon dari Universitas Gunadarma melakukan penelitian yang berjudul “*Text-To-Speech Indonesia Dengan Menggunakan Mbrola Synthesizer*”, menyajikan sebuah program aplikasi Text Editor dan mengintegrasikan teknologi Text-To-Speech

Synthesis Mbrola yang sudah memiliki *database* id1 ke dalam aplikasi tersebut.

- c. Pada 17 November 2006 Iwan Iwut Tritoasmoro dari ITT Telkom mempublikasikan, “*Text-To-Speech* Bahasa Indonesia Menggunakan *Concatenation Synthesizer* Berbasis Fonem”, dalam seminar nasional di bali
- d. Pada tahun 2007, Fanny Agung Nugroho dari ITTelkom melakukan penelitian yang berjudul “Perancangan *Text to speech* Bahasa Indonesia(Model Prosodi Dataset Pada Mbrola)”.
- e. Pada tahun 2008, Saono Wicaksono dari ITB melakukan penelitian yang berjudul Aplikasi “*Text To Speech synthesizer* Bahasa Indonesia Menggunakan Euler Tts System”. Penelitian ini bertujuan mengadaptasi *Euler TTS System* untuk bisa mengucapkan ucapan dalam bahasa Indonesia. Dalam penelitian ini modul preprocessor dan modul phonetizer diubah disesuaikan dengan ejaan bahasa Indonesia.
- f. Pada tahun 2009 Ratri Priyandewi dari ITTelkom melakukan penelitian “Pemodelan Prosodi Metode Dataset Untuk Text-To-Speech Bahasa Indonesia Dengan Synthesizer Mbrola”
- g. Pada tahun 2009 Bayu G Wundari dari teknik elektro UI melakukan penelitian untuk skripsi dengan judul “Sistem *Text to Speech* dengan Metode *Unit Selection Synthesis* untuk Bahasa Indonesia” sistem *text to speech* ini ditulis didalam bahasa delphi

Untuk penelitian *singing synthesizer*, sejauh yang saya tahu di Indonesia belum pernah ada yang melakukannya. Sedangkan untuk pembuatan *voicebank* UTAU sendiri, walaupun sudah cukup banyak dari seluruh dunia yang melakukannya bahkan di Indonesia sudah ada yang pernah membuatnya, *voicebank* tersebut dibuat lebih untuk membuat lagu lagu bahasa jepang bukan bahasa Indonesia. *Voicebank*

voicebank buatan orang Indonesia seperti EMU-3 dan Millya hanya terdiri dari difon yang dalam hal ini kurang bisa mensupport lagu lagu dalam Indonesia karena tidak akan bisa dengan jelas mengucapkan kata yang mengandung unsur kvk seperti “ber” pada kata berhasil, ataupun “bang” dalam kata bangsa.

Di Indonesia *voicebank* untuk *speech synthesis* yang terpublikasi dengan baik dan tersedia untuk umum diantaranya adalah:

- a. *Database diphone* ID1 buatan AA Arman dari ITB
- b. *Database formant* pada *speech engine* ESpeak
- c. *Voicebank* emu-3 untuk aplikasi UTAU
- d. *Voicebank* Millya act 6 untuk aplikasi UTAU

Dari *voicebank-voicebank* diatas, yang berbasiskan pada *pre-recorded speech* adalah *database diphone* ID-1, *voicebank* emu-3, dan *voicebank* Millya act 6. Dari *voicebank-voicebank* tersebut tidak ada yang menggunakan KVK sebagai salah satu unit fonem dalam *voicebank*-nya.

1.2 Tujuan Penulisan

Tujuan penulisan Skripsi ini adalah untuk menganalisa penggunaan unsur KVK dalam *voicebank* yang digunakan dalam aplikasi *speech synthesizer* beralgoritma SOLA-Resampling, *speech synthesizer* berbasis UTAU, dan *singing synthesizer* UTAU untuk membuat ucapan dan nyanyian sintesis dalam bahasa Indonesia.

1.3 Pembatasan Masalah

Masalah dalam skripsi ini akan dibatasi sampai pembuatan *voicebank* yang dinamakan “LULU3”, pengujian *voicebank* tersebut pada aplikasi *speech synthesizer* dan *singing synthesizer*, serta analisa atas suara yang dihasilkan sistem *speech synthesizer* dan *singing synthesizer* yang menggunakan *voicebank* “LULU3”. *Voicebank* “LULU3” dirancang terdiri dari 1765 file suara yang berformat V, KV,

VK, dan KVK. Daftar lengkap komposisi voicebank “LULU3” ada pada lampiran B. *Speech synthesizer* yang digunakan untuk pengujian adalah *speech synthesizer* dengan algoritma SOLA-Resampling dan *speech synthesizer* berbasis UTAU. *Singing synthesizer* yang digunakan untuk pengujian adalah *singing synthesizer* UTAU. Ucapan dan lagu yang dipilih dalam skripsi ini tidak mengandung kata serapan. Analisa pada skripsi ini adalah analisa hasil *survey* terhadap sejumlah *responden* atas perbandingan kejelasan antara suara sintesis yang dibuat dari voicebank berunsur KVK dibandingkan dengan suara sintesis yang dibuat dari voicebank yang tidak berunsur KVK. *Responden* dipilih dari mahasiswa departemen teknik elektro Universitas Indonesia angkatan 2006,2007, dan 2008.

1.4 Metode Penelitian

Adapun metode penulisan yang penulis lakukan dalam menyusun skripsi ini adalah:

1. Dengan melakukan studi literatur dan kepustakaan dari situs situs internet, buku, e-book, jurnal, dan artikel-artikel terkait yang berkaitan dengan system *speech synthesizer*, *singing synthesizer*, dan tata bahasa Indonesia,
2. Teknik observasi, yaitu dengan meninjau dan mempelajari sistem sistem *speech synthesizer* dan *singing synthesizer* yang sudah ada
3. Diskusi dengan dosen pembimbing tugas akhir dan teman teman
4. Pembuatan Voicebank yang dinamakan “LULU3” yang terdiri dari 1765 fonem berformat V, VK, KV, dan KVK.
5. Pembuatan Aplikasi *Speech synthesizer* beralgoritma SOLA-Resampling dengan menggunakan matlab
6. Pembuatan Aplikasi *Speech synthesizer* berbasis UTAU dengan menggunakan matlab
7. Pembuatan Aplikasi KVKhelper untuk mengolah file suara berformat KVK yang akan digunakan dalam *singing synthesizer* UTAU

8. Pengujian akurasi pitch dan durasi dari aplikasi *singing synthesizer* UTAU, *speech synthesizer* berbasis UTAU, *speech synthesizer* beralgoritma SOLA-Resampling, dan KVKhelper

9. *Survey* terhadap sejumlah *responden* yang terdiri dari mahasiswa teknik elektro Universitas Indonesia angkatan 2006-2008 untuk membandingkan antara suara sintesis yang dibuat dari voicebank “LULU” yang mempunyai unsur KVK dengan suara sintesis yang dibuat dari voicebank “LULU” yang tidak mempunyai unsur KVK.

10. Pengolahan dan Analisa atas hasil *survey* pada poin nomor 9.

1.5 Sistematika Penulisan

Skripsi ini terdiri dari lima bab, dimana masing masing bab akan menjelaskan sebagai berikut

a. Bab I Pendahuluan

Pada bab ini akan dijelaskan latar belakang masalah, tujuan penulisan, pembatasan masalah metodologi penulisan, dan sistematika penulisan.

b. Bab II Suara, *Speech synthesizer*, dan Tata Bahasa Indonesia

pada bab ini akan dibahas mengenai teori teori penunjang yang menjelaskan tentang suara, karakteristik suara manusia, *speech synthesizer*, aturan fonem pada bahasa Indonesia.

c. Bab III Perancangan *Voicebank* dan *Speech synthesizer*

Pada bab ini akan dijelaskan tentang proses pembuatan *voicebank* “LULU3” yang terdiri dari difon dan KVK, pembuatan sistem *speech synthesizer*, dan penjelasan tentang software *singing synthesizer* UTAU

d. Bab IV Simulasi dan Analisa

Pada bab ini akan dijelaskan mengenai simulasi yang dilakukan terhadap *voicebank* “LULU3”, serta pengolahan dan analisa hasil *survey* penilaian *responden* terhadap ucapan dan nyanyian sintesis yang terdiri dari unsur KVK, KV, V, dan VK dibanding dengan ucapan dan nyanyian sintesis yang terdiri dari unsur KV, V dan VK.

e. Bab V Kesimpulan

Pada bab ini akan dijelaskan mengenai kesimpulan yang dapat diambil dari skripsi ini



BAB 2

SUARA, *SPEECH SYNTHESIS*, DAN TATA BAHASA INDONESIA

2.1 Suara

Dalam fisika, suara adalah gelombang berjalan yang dapat ditransmisikan melalui benda padat, cair, plasma, ataupun gas. Suara yang dapat didengar manusia mempunyai frekuensi dari 20 Hz – 20000 Hz. Suara yang dihasilkan manusia mempunyai frekuensi dari 80 Hz sampai 1100 Hz. Gelombang suara mempunyai beberapa parameter seperti frekuensi, panjang gelombang, periode, amplitudo, intensitas, kecepatan, dan arah

2.1.1 Frekuensi

Dalam kaitannya dengan suara, frekuensi adalah jumlah getaran per detik. Satuan frekuensi adalah Hertz, yang diambil dari nama fisikawan Jerman, Heinrich Hertz. Frekuensi dalam skripsi ini mengacu pada dua hal yaitu frekuensi *sampling* dan frekuensi *pitch*. Frekuensi *sampling* adalah banyaknya nilai yang *disampling* dari sebuah sinyal dalam satu detik. Misalkan ada sebuah *file wave* (.wav) mempunyai durasi 1 detik dan frekuensi *sampling* 44100 Hz. Berarti dalam *file* tersebut tersimpan 44100 nilai. Sedangkan frekuensi *pitch* berkaitan dengan tinggi rendahnya sebuah nada. Semakin besar frekuensi *pitch* maka semakin tinggi nada yang dihasilkan.

2.1.2 Periode

Periode adalah kebalikan dari frekuensi, diartikan sebagai waktu yang diperlukan untuk melakukan satu getaran.

2.1.3 Amplitudo

Amplitudo adalah nilai tertinggi dalam sebuah getaran, dalam sinyal suara semakin tinggi amplitudonya maka semakin keras suaranya.

2.2 Suara Manusia

Suara manusia adalah suara yang dibuat manusia dengan menggunakan pita suaranya. Berdasarkan jenis kelamin, suara manusia dibedakan menjadi suara laki-laki dan suara perempuan. Suara manusia memiliki frekuensi 80 Hz – 1100 Hz. Dalam dunia musik suara laki laki dibedakan menjadi bass, *bariton*, *tenor*, dan *countertenor*. Bass memiliki rentang nada dari E2-E4. *Bariton* memiliki rentang nada dari G2-F4. *Tenor* memiliki rentang nada dari C3 – A4. Untuk suara wanita, dalam musik dibagi menjadi tiga yaitu *contralto* yang mempunyai rentang nada F3-G5, *mezzosopran* yang mempunyai rentang nada dari A3-A5, dan sopran yang mempunyai rentang nada dari C4-C6. Daftar lengkap dari rentang suara manusia dapat dilihat pada gambar 2.1, pada gambar ini dijelaskan tentang jenis jenis suara berdasarkan frekuensi *pitchnya*



Gambar 2.1 Jenis Suara Manusia Berdasarkan Frekuensi *Pitchnya* [11]

Dalam dunia musik setiap nada memiliki frekuensi *pitch*. Daftar frekuensi *pitch* dari sebuah nada dapat dilihat pada tabel 2.1, pada tabel ini ditunjukkan frekuensi dari setiap nada mulai dari C1 – B7. Pada tabel 2.1 bisa dilihat bahwa antara oktaf, nilai frekuensi *pitch*-nya akan meningkat dua kali. Contohnya adalah pada C, disini C1 bernilai 32.7 Hz, maka nilai C2 adalah dua kali C1 yaitu 65.4 Hz, dan nilai C3 adalah 2 kali nilai C2 yaitu 130.8 Hz.

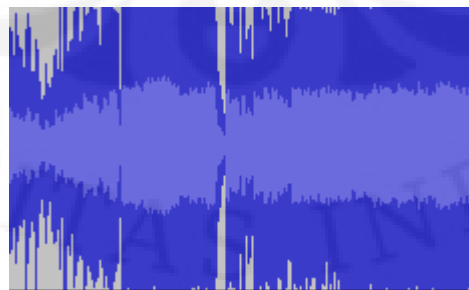
Tabel 2.1 Daftar Nilai Frekuensi *Pitch* untuk Setiap Nada [12]

Note	Hz	Note	Hz	Note	Hz	Note	Hz	Note	Hz	Note	Hz	Note	Hz
C1	32.7	C2	65.4	C3	130.8	C4	261.6	C5	523.3	C6	1046.5	C7	2093.0
C#1	34.6	C#2	69.3	C#3	138.6	C#4	277.2	C#5	554.4	C#6	1108.7	C#7	2217.5
D1	36.7	D2	73.4	D3	146.8	D4	293.7	D5	587.3	D6	1174.7	D7	2349.3
D#1	38.9	D#2	77.8	D#3	155.6	D#4	311.1	D#5	622.3	D#6	1244.5	D#7	2489.0
E1	41.2	E2	82.4	E3	164.8	E4	329.6	E5	659.3	E6	1318.5	E7	2637.0
F1	43.7	F2	87.3	F3	174.6	F4	349.2	F5	698.5	F6	1396.9	F7	2793.8
F#1	46.2	F#2	92.5	F#3	185.0	F#4	370.0	F#5	740.0	F#6	1480.0	F#7	2960.0
G1	49.0	G2	98.0	G3	196.0	G4	392.0	G5	784.0	G6	1568.0	G7	3136.0
G#1	51.9	G#2	103.8	G#3	207.7	G#4	415.3	G#5	830.6	G#6	1661.2	G#7	3322.4
A1	55.0	A2	110.0	A3	220.0	A4	440.0	A5	880.0	A6	1760.0	A7	3520.0
A#1	58.3	A#2	116.5	A#3	233.1	A#4	466.2	A#5	932.3	A#6	1864.7	A#7	3729.3
B1	61.7	B2	123.5	B3	246.9	B4	493.9	B5	987.8	B6	1975.5	B7	3951.1

Meskipun dalam teorinya manusia hanya bisa mengeluarkan suara dari 80-1100 Hz, dalam kenyataannya beberapa penyanyi profesional dapat mengeluarkan suara yang lebih tinggi dan lebih rendah. Rekor suara tertinggi dipegang oleh Adam Lopez dengan frekuensi 4138 Hz (C8).

2.3 Suara digital

Suara digital adalah suara yang disimpan dalam bentuk data digital. Dalam mengubah ke dalam bentuk digital dilakukan proses *sampling* dengan *pulse code modulation*. Dalam suara digital, ada beberapa hal yang perlu diperhatikan seperti *channel*, *samplerate*, kompresi, dan *bit depth*. Gambar 2.2 adalah *visualisasi* dari potongan suara digital pada *software* Audacity.



Gambar 2.2 Suara Digital

2.3.1 Channel

Channel adalah sebuah *stream* dari satu sinyal *audio*. Dalam suatu musik digital, sangat memungkinkan untuk mempunyai lebih dari satu *channel*. Musik *stereo* misalnya mempunyai dua buah *channel*. *Channel* pertama akan dikeluarkan lewat *speaker* pertama, *channel* kedua akan dikeluarkan lewat *speaker* kedua. Informasi yang terdapat pada *channel* bisa sama bisa berbeda antara satu dengan lainnya. Untuk musik *mono*, musik tersebut hanya mempunyai satu *channel*, apabila dalam suatu komputer memiliki dua *speaker*, kedua *speaker* akan mengeluarkan suara yang sama

2.3.2 frekuensi *sampling*

Frekuensi *sampling* atau yang biasa disebut *sample rate* adalah banyaknya *sample* yang diambil dalam satu detik. *Sample* ini sesungguhnya adalah nilai amplitudo pada saat *sample* diambil. *Sample* pada *file* musik biasanya adalah 44.100 Hz, artinya saat pendigitalan suara, komputer mengambil 44.100 *sample* dalam satu detik. Dalam pendigitalan suara, semakin besar frekuensi *sampling*nya semakin presisi suaranya.

2.3.3 *Bit depth*

Bit depth adalah banyaknya *bit* yang dipakai dalam menyimpan satu *sampling*. Dalam musik umumnya *bit* yang dipakai adalah 8 dan 16 *bit*. Semakin banyak *bit* yang digunakan semakin bagus kualitas suaranya

2.3.4 Kompresi

Kompresi sering digunakan dalam musik digital, biasanya untuk mengecilkan ukuran *file* namun tidak merubah kualitasnya. Kompresi ada yang bersifat *lossy* ada yang tidak. Bedanya adalah *lossy* berarti kualitas suara tersebut mengalami penurunan kualitas tapi masih layak untuk didengar, perbandingan suara dengan ukuran *file* nya sangat baik. *Non lossy* berarti suara tidak mengalami penurunan kualitas. Untuk mendengarkan musik yang ingin didengar dalam bentuk kompresi, musik tersebut harus didengar melalui *decoder*. Satu hal yang penting disini, karena *decoder* membutuhkan suatu proses juga,

biasanya *file* yang digunakan untuk keperluan pemrograman tidak boleh dikompresi, artinya *file* tersebut harus merupakan *file* tanpa kompresi seperti format “.wav” misalnya

2.4 *Speech Synthesis*

Speech synthesizer dalam bahasa Indonesia diartikan sebagai sebuah sistem yang dapat mengeluarkan suara manusia *synthesis*. Suara yang dihasilkan dari sebuah *speech synthesizer* ini nantinya dapat diaplikasikan di *hardware* atau *software*. *Singing synthesizer* adalah sebuah sistem yang dapat menghasilkan suara manusia dengan nada yang dapat diatur sesuai partitur music sehingga suara yang dihasilkan menyerupai suara manusia yang sedang bernyanyi.

Penelitian tentang *speech synthesis* dan *singing synthesis* sangat erat berhubungan. Namun, ada perbedaan antara nyanyian dan ucapan. Dalam nyanyian, “*voiced sound*” lebih banyak terdengar dibandingkan dengan didalam ucapan [2]. Dalam bahasa Inggris perbandingan antara *voiced sound* pada nyanyian dan ucapan adalah 90% berbanding 60% [2]. “*Voiced sound*” memiliki sifat yang *periodic* sehingga dapat lebih mudah untuk di analisis [2]. Perbedaan lainnya adalah dalam nyanyian, frekuensi *pitch* yang digunakan lebih lebar variasinya dibanding dalam mengeluarkan ucapan [2]. Frekuensi *pitch* dalam nyanyian biasanya diatur terlebih dahulu sedangkan dalam ucapan, frekuensi yang dikeluarkan biasanya tercipta secara spontan [2]. Pada ucapan, frekuensi *pitch* dan durasi juga tergantung pada emosi dari pembicara. Dalam publikasi yang berjudul “*Generation of Emotions by a Morphing Technique in English, French and Spanish*”, Philippe Boula de Mareüil, Philippe Célérier, dan Jacques Toen menjelaskan tentang variasi volume, *pitch* dan durasi dalam sebuah ucapan berdasarkan emosi manusia [5]. Untuk bahasa Inggris, ketika marah, durasi *pause* akan memendek dan suku kata yang mengalami penekanan akan memanjang dan mendapat penambahan energi [5]. Untuk emosi senang, dalam pengucapan kalimat bahasa Inggris *pitch* rata-rata dan energi meningkat [5].

Untuk membuat sebuah sistem *speech synthesizer*, ada beberapa metode, namun yang paling berkembang ada dua yaitu *concatenative synthesis*, dan *formant synthesis*.

Concatenative synthesis sendiri dibagi menjadi dua yaitu *unit selection synthesis*, dan *diphone synthesis*.

2.4.1 *Concatenative synthesis*

Concatenative synthesis adalah teknik untuk mensintesis suara dengan menggabungkan *sample sample* pendek rekaman suara. Lama dari setiap unit suara tidak secara tegas di definisikan dan bisa bervariasi sesuai dengan implementasinya, dalam *speech synthesis*, *concatenative synthesis* menggabungkan berbagai kata untuk menjadikannya sebuah ucapan. [9] *Concatenative synthesis* ada dua macam yaitu *diphone synthesis* dan *Unit selection synthesis*. [9]

a. *Diphone synthesis*

Diphone synthesis menggunakan database minimal yang mengandung semua *diphone (sound to sound transition)* dalam sebuah bahasa. Jumlah *diphone* tergantung dari fonemisasi sebuah bahasa. Spanyol mempunyai 800 *diphone* dan Jerman mempunyai 2500 *diphone*. [9]

b. *Unit Selection Synthesis*

Unit selection synthesis menggunakan database rekaman suara. Bedanya dengan *diphone synthesis* adalah pada *unit selection synthesis* rekaman tidak terbatas hanya berupa difon saja tapi juga bisa dimasukan database fonem, difon, sukukata, morfem, kata, frase dan kalimat. Kelebihan dari *Unit selection synthesis* ini adalah cara ini memiliki kualitas suara yang cenderung lebih bagus tapi database yang dibuat cenderung lebih besar daripada teknologi lainnya.

2.4.2 *Formant synthesis*

Formant synthesis tidak menggunakan *sample* rekaman suara manusia, melainkan menggunakan *additive synthesis* dan model akustik. Parameter-parameter yang digunakan dalam *formant synthesis* biasanya menggunakan frekuensi dasar, suara, dan level *noise*

yang divariasikan setiap waktu untuk menghasilkan suatu suara manusia. *Formant synthesis* memiliki kecenderungan mempunyai suara yang kurang natural dibandingkan dengan *concatenative synthesis*. Akan tetapi *formant synthesis* mempunyai kelebihan seperti tingkat kejelasan yang lebih tinggi daripada *concatenative synthesis*. [9]

2.3 Tata Bahasa Indonesia

Bahasa Indonesia adalah bahasa resmi yang dipakai oleh masyarakat Indonesia. Bahasa Indonesia mempunyai beberapa aturan dan tata bahasa. Kedua hal ini sangat berkaitan dalam pembuatan sistem *speech synthesizer* dan *singing synthesizer*.

2.3.1 Klasifikasi bunyi

Berdasarkan Perwujudannya dalam sukukata, bunyi dibagi menjadi dua macam, yaitu bunyi tunggal dan bunyi rangkap. Bunyi tunggal ini hanya mempunyai satu huruf saja, bunyi tunggal dibagi menjadi dua yaitu vokal (A,I,U,E,E2,O), dan Konsonan (B,C,D,F,G,H,J,K,L,M,N,P,Q,R,S,T,V,W,X,Y,Z). Bunyi rangkap adalah bunyi yang terbentuk dari gabungan huruf, bunyi rangkap dibagi menjadi dua yaitu diftong (vokal rangkap), dan klaster (gugusan konsonan): pr, pt, kr, dan lain lain

2.3.2 Fonem dan Fonemisasi

Fonem adalah satuan bunyi bahasa yang bersifat fungsional. Sedangkan fonemisasi adalah usaha untuk menemukan bunyi-bunyi yang berfungsi dalam rangka pembedaan makna tersebut. fonem terdiri dari berbagai macam jenis diantaranya kvk, vk, kv, dan v. *Diphone* adalah satuan bunyi yang sesungguhnya merupakan transisi antara dua huruf (vv, vk, kv, kk). Dalam kaitannya dengan sistem *speech synthesis*, *diphone* merupakan bagian dengan frekuensi *pitch* yang stabil sehingga dalam *speech synthesis* ada sebuah metode yang dinamakan *diphone synthesis* yang merupakan sebuah metode *speech synthesis* yang hanya menggunakan database *diphone* saja. *Triphone* merupakan kumpulan huruf yang terdiri dari tiga buah huruf yang bisa dilafalkan dengan jelas. contohnya adalah “bra”, “pra”, dan “tri”.

BAB 3

PERANCANGAN VOICEBANK DAN SPEECH SYNTHESIZER

3.1 Pembuatan Voicebank

Pembuatan *voicebank* merupakan hal terpenting dalam aplikasi *speech synthesizer* dan *singing synthesizer*, *voicebank* yang bermutu baik akan menghasilkan suara sintesis yang baik pula. Disini *voicebank* yang dibuat adalah *voicebank* yang dinamakan *voicebank* “LULU3” yang terdiri dari 1765 file suara berformat V (Vokal), VK (Vokal Konsonan), KV (Konsonan Vokal), dan KVK (Konsonant Vokal Konsonan). Nama “LULU3” tidak mengacu kepada hal yang teknis, hanya sekedar nama untuk sebuah *voicebank*. Kata “LULU” diambil dari nama penyedia *voicebank* “LULU3”. Angka 3 diambil dari nama pembuat skripsi ini yaitu “Tribudi”. Pembuatan *voicebank* “LULU3” dilakukan didalam ruangan rendah noise, dengan menggunakan *microphone* sennheisser e835, sebuah keyboard PSR-2000 sebagai pre-amplifier, dan sebuah *laptop* acer 4730 z untuk perekaman. Suara yang direkam untuk *voicebank* ini adalah suara seorang penyanyi terlatih berjenis kelamin perempuan berusia 14 tahun. Ada tiga tahapan dalam proses pembuatan *voicebank* ini yaitu:

- Perekaman
- Pendeteksian fonem
- Pengolahan Suara

3.1.1 Perekaman

Pada tahapan perekaman, seorang penyanyi terlatih akan mengucapkan berbagai suara yang berformat VK, KVK, KV, dan V. Cara pengucapan dalam perekaman ini dilakukan beragam tergantung tingkat kesulitan dalam mengucapkannya. Ada yang diucapkan sekaligus banyak. misal: a, i, u, e, o, e2. ada pula yang dilakukan dengan mengucapkan sebuah kata seperti mengucapkan kata bara untuk merekam suara ra. Nada pada fonem fonem yang diucapkan berkisar pada nada A#3. Proses perekaman dilakukan dengan menggunakan *microphone*

sennheisser e835 yang dihubungkan terlebih dahulu ke sebuah *preamplifier* (Yamaha psr 2000) sebelum dihubungkan ke *laptop*. Setelah dilakukan pengamatan, file suara yang direkam dengan format KV harus ada dua jenis yaitu yang panjang dan yang pendek. Hal ini karena algoritma SOLA mempunyai batas faktor *time scaling* sehingga dibutuhkan file KV yang durasinya singkat untuk keperluan *speech synthesizer*. Sedangkan file KV yang panjang dibutuhkan untuk *singing synthesis* karena suara berformat KV sering digunakan untuk nada yang memiliki durasi lama.

3.1.2 Pendeteksian fonem

Pendeteksian fonem adalah hal yang penting, hal ini bertujuan untuk mengurangi jumlah *voicebank* yang harus disediakan sehingga dapat mengurangi ukuran *file*. Disini pendeteksian dilakukan dengan mencarinya di kamus bahasa Indonesia yang diterbitkan dinas pendidikan nasional pada tahun 2008. Apabila ditemukan kata dalam bahasa Indonesia yang mengandung sebuah fonem yang belum ada pada daftar fonem, maka fonem tersebut akan dimasukkan kedalam tabel daftar fonem. Berikut ini adalah daftar fonem yang berhasil dideteksi oleh penulis.

a. V(Vokal)

Voicebank yang masuk dalam kategori ini adalah *voicebank* yang terdiri dari Vokal saja. Tabel 3.1 merupakan kumpulan *voicebank* yang terdiri dari vokal yang berhasil dideteksi penulis.

Tabel 3.1 Daftar Fonem Vokal Bahasa Indonesia

Fonem	Kata	Fonem	kata
a	apel	o	obat
i	ibu	e2	elang
u	udang	au	auto
e	enak		

Pada Tabel 3.1, Kolom Fonem adalah daftar fonem dalam bahasa Indonesia, Kolom kata adalah daftar kata dalam bahasa Indonesia dimana fonem tersebut terkandung,

b. VK (Vokal Konsonan)

Voicebank yang masuk dalam kategori ini adalah *voicebank* yang memiliki *format* VK, yaitu terdiri dari Vokal didepan kemudian diikuti oleh konsonan, disini vokal dan konsonan bisa lebih dari satu, artinya fonem seperti “ang” masuk dalam katagori ini. Tabel 3.2 adalah cuplikan daftar fonem dengan *format* VK. Daftar lengkap hasil deteksi fonem yang ber*format* KV dapat dilihat pada lampiran C.

Tabel 3.2 Cuplikan Daftar Fonem yang Ber*format* VK

Fonem	Kata	Fonem	Kata
ab	abjad	am	ambil
ad	admiral	an	andai
af	afdol	ang	angkuh
ah	ahli	ar	arti
ak	aksara	as	asma
al	alga	at	atmosfer

c. KV (Konsonan Vokal)

Voicebank yang masuk dalam kategori ini adalah *voicebank* yang memiliki *format* KV, yaitu terdiri dari Konsonan didepan kemudian diikuti oleh Vokal, disini vokal dan konsonan bisa lebih dari satu, artinya fonem seperti “bra” masuk dalam katagori ini. Tabel 3.3 adalah cuplikan daftar fonem dengan *format* KV. Daftar lengkap hasil deteksi fonem yang ber*format* KV dapat dilihat pada lampiran C.

Tabel 3.3 Cuplikan Daftar Fonem Berformat KV

Fonem	Kata	Fonem	Kata
ba	hamba	ca	cari
bi	binatang	ci	peci
bu	burung	cu	cumi
be	besan	da	dari
bo	bodoh	co	colok
be2	belakang	ce2	cerita

d. KVK (Konsonan Vokal)

Voicebank yang masuk dalam kategori ini adalah *voicebank* yang memiliki *format* KVK, yaitu terdiri dari Konsonan didepan kemudian diikuti oleh vokal dan setelah itu diikuti oleh konsonan. Disini vokal dan konsonan bisa lebih dari satu, artinya fonem seperti “bang” masuk dalam katagori ini. Tabel 3.4 adalah cuplikan daftar fonem dengan *format* KVK. Daftar lengkap hasil deteksi fonem yang *berformat* KVK dapat dilihat pada lampiran.

Tabel 3.4 Cuplikan Daftar Fonem Berformat KVK

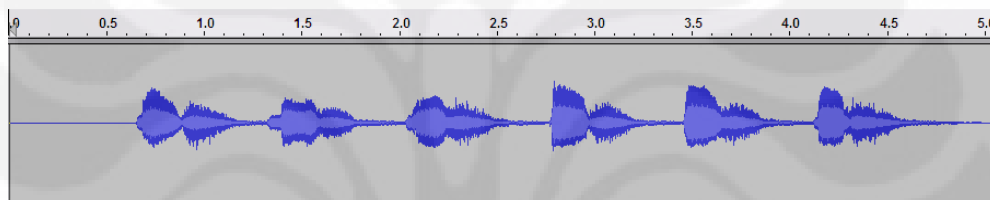
Fonem	Kata	Fonem	Kata
bah	bahtera	bar	lebar
bad	abad	bas	tebas
bab	bab	bat	hebat
bak	tebak	baw	kerbau
bal	sambal	bay	lambai
bam	lebam	bih	tasbih

3.1.3 Pengolahan Data Rekaman

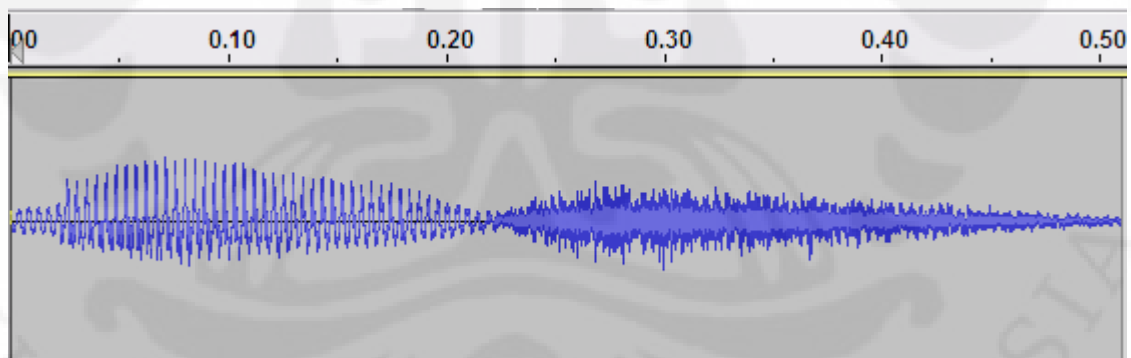
File file Suara yang sudah direkam, kemudian diolah. Ada tiga tahapan pengolahan yang harus dilakukan untuk bisa menggunakan *voicebank* tersebut ke dalam *speech synthesizer* dan *singing synthesizer*. Tahapan tahapan tersebut adalah pemisahan suara, *pitch detection*, dan pencatatan durasi dan *pitch*.

3.1.3.1 Pemisahan Suara

Pada tahapan ini, data hasil rekaman dipisahkan (*trimming*) sehingga hanya terdiri dari satu unit saja dengan menggunakan audacity, Gambar 3.1 adalah gambar *file* mentah hasil rekaman yang dilihat pada audacity. Sedangkan gambar 3.2 adalah gambar *file* setelah di *trimming*.



Gambar 3.1: *File* Suara BasBisBus.wav Hasil Rekaman



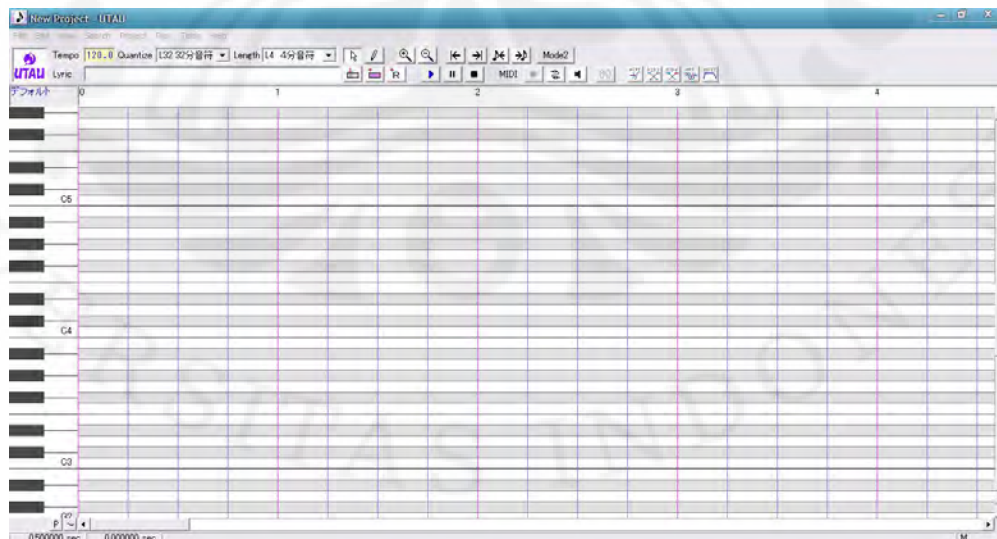
Gambar 3.2: *File* Suara Bas.wav yang Diambil dari *File* Suara BasBisBus.wav

Setelah seluruh *file* suara berhasil dipisahkan sehingga hanya terdengar satu fonem saja, *file file* tersebut digabung kedalam sebuah *folder*. Disini saya memasukkannya kedalam *folder* “LULU3”.

3.1.3.2 Pendeteksian Frekuensi *Pitch*

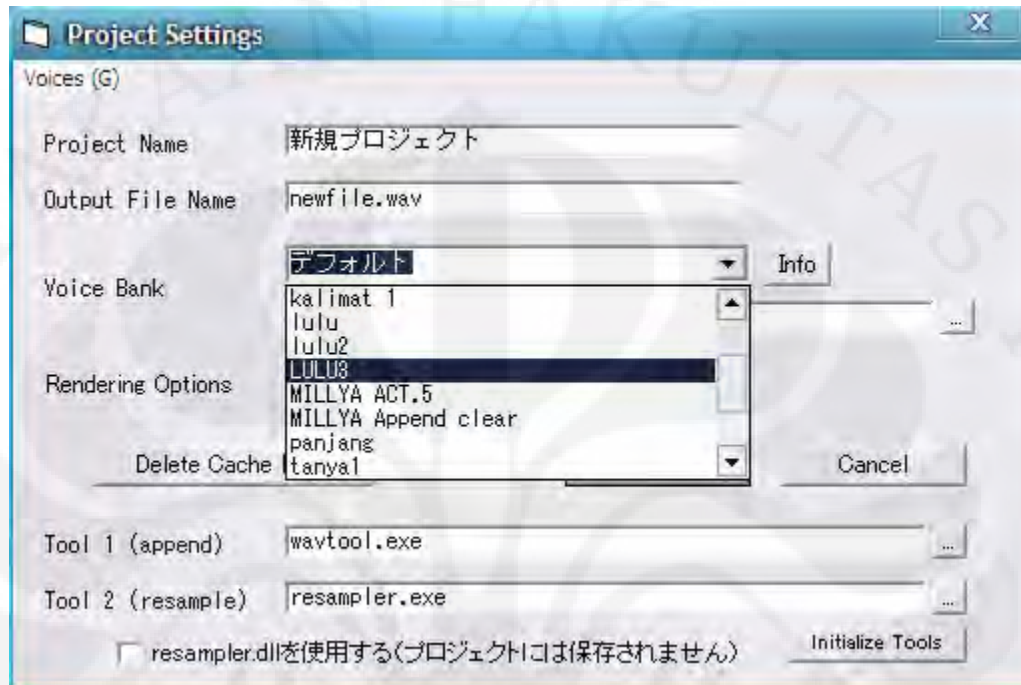
Pada tahapan ini dilakukan pendeteksian frekuensi *pitch* dengan menggunakan UTAU. *Database pitch* ini dibutuhkan untuk menentukan nilai faktor perubahan frekuensi dalam proses *pitch shifting*. Untuk melakukan pendeteksian *pitch* dengan UTAU, maka yang pertama kali harus dilakukan adalah menginstal UTAU. *Software* UTAU bisa didapatkan dengan gratis dari situs milik Mr. Ameya di <http://utau2008.xrea.jp/>. Versi UTAU yang digunakan pada skripsi ini adalah versi 0.2.76 yang dirilis pada 25 Mei 2010. Setelah menginstal UTAU, langkah selanjutnya adalah menginstal *patch* bahasa inggris yang tersedia di http://utau.wikia.com/wiki/UTAU_wiki:UTAU_GUI_Translation. *Patch* bahasa inggris ini sangat dibutuhkan karena *software* UTAU ini ditulis dalam bahasa jepang sehingga sangat sulit dimengerti bagi orang-orang yang tidak mengerti bahasa jepang. Langkah selanjutnya adalah memasukan *folder voicebank* LULU3 yang telah dibuat kedalam *directory voicebank* UTAU yaitu di “C:\program files\UTAU\voice”.

Setelah langkah-langkah di atas telah dilakukan maka, pendeteksian *pitch* dengan UTAU sudah bisa dilakukan. Pertama-tama buka program UTAU dari *windows*, *shortcut* program biasanya terletak di *desktop* atau *startmenu*. Apabila tidak berhasil ditemukan maka, buka *directory* “C:\program files\UTAU\ “ dan jalankan UTAU.exe dari sana. Setelah itu akan muncul tampilan *GUI* seperti pada gambar 3.4.

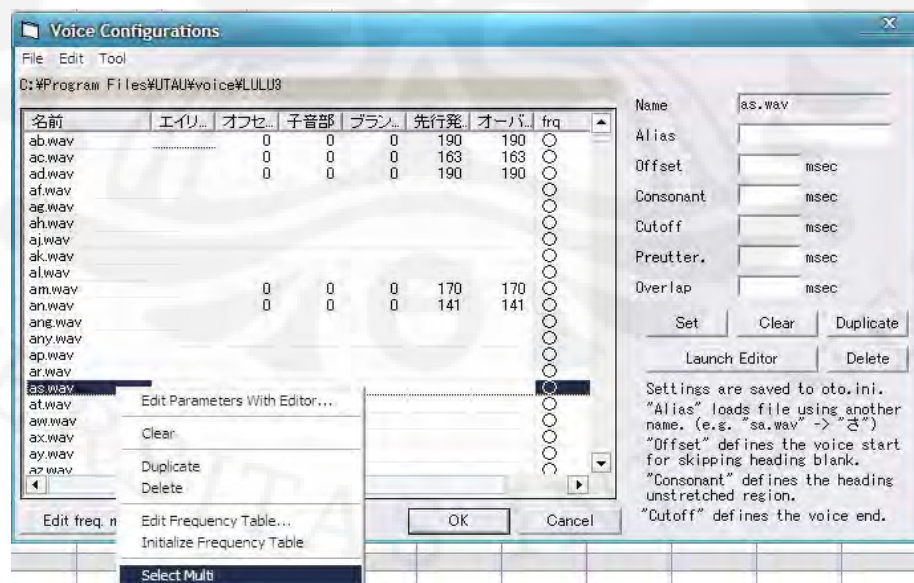


Gambar 3.3 Tampilan *GUI* pada UTAU

Langkah selanjutnya adalah klik menu *project* kemudian klik *project properties*, dan pilih *voicebank* “LULU3” yang telah dibuat. Tekan OK jika sudah. Gambar 3.4 adalah tampilan *GUI* pada *project properties* dalam UTAU.



Gambar 3.4 Tampilan *Project Properties* pada UTAU

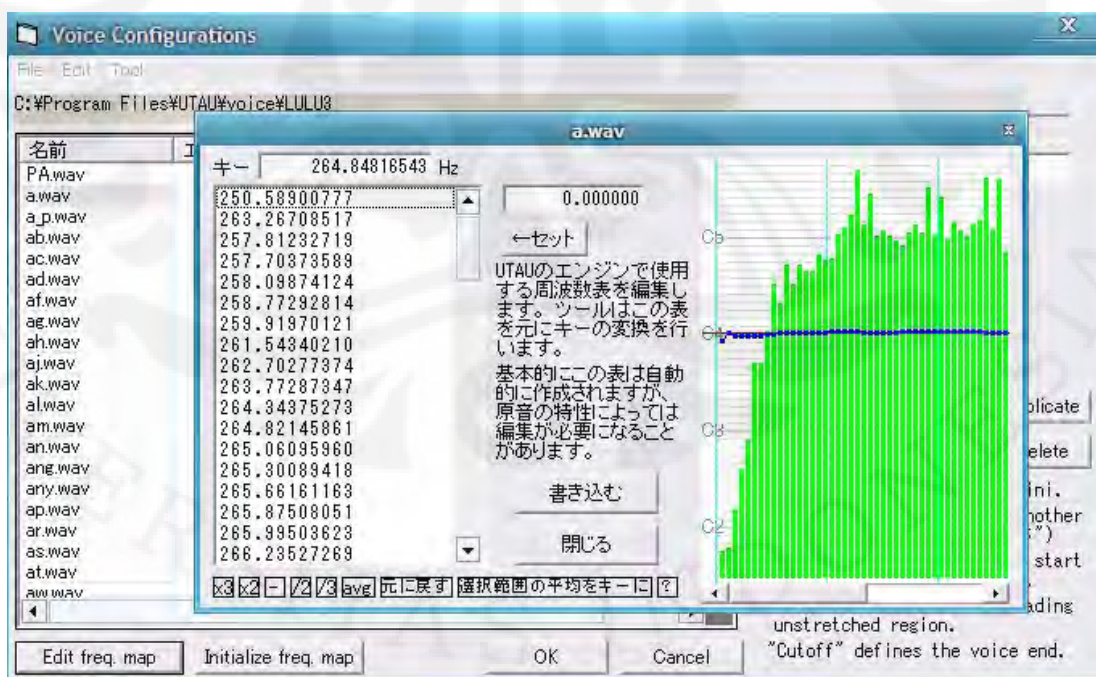


Gambar 3.5 Perintah *Select Multi* pada Proses Pendeteksian *Pitch*

Selanjutnya klik menu *Tools* kemudian *voicebank* setting. Maka akan muncul tampilan seperti gambar 3.5 . Selanjutnya klik kanan pada salah satu baris, lalu pilih “*select multi*” seperti pada gambar 3.5. Setelah itu pilih seluruh baris dan selanjutnya klik tombol “*set initial frequency*” untuk memulai proses pendeteksian *pitch*. Proses pendeteksian *pitch* akan membuka sebuah *command prompt windows*. Proses ini akan berlangsung sangat lama, Pendeteksian satu buah *file* suara akan membutuhkan waktu sekitar 10 – 30 detik. *File* berekstensi *frq* akan muncul setelah pendeteksian *pitch* dari suatu *file* selesai dilakukan

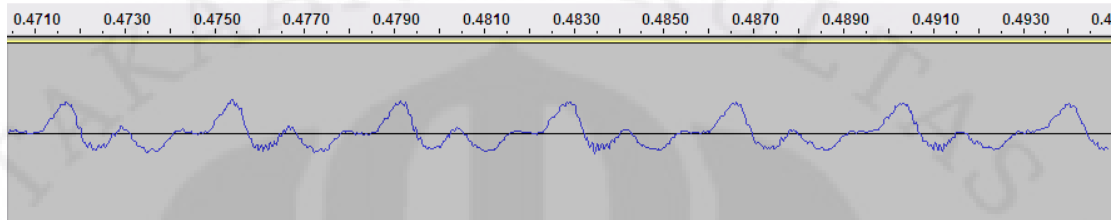
3.1.3.3 Pencatatan Durasi dan *Pitch* rata rata

Pencatatan durasi dan *pitch* rata rata berguna untuk mendapatkan nilai perbandingan *pitch* dan nilai perbandingan durasi yang akan dipakai pada aplikasi *speech synthesizer* yang digunakan dalam skripsi ini. Untuk mencatat *pitch* rata rata maka caranya adalah dengan melihatnya satu persatu pada *voicebank* setting yang ada pada UTAU. Klik tombol *tools* pada UTAU dan kemudian klik “*voicebank settings*”. Setelah itu klik salah satu nama *file* suara kemudian klik edit frequency map untuk menampilkan tampilan seperti pada gambar 3.6



Gambar 3.6 Frekuensi Rata Rata yang Sudah Dideteksi oleh UTAU

Untuk pencatatan durasi, digunakan *software* audacity. Disini *file file* suara dibuka dengan audacity kemudian di lihat durasi *filenya*. Gambar 3.7 adalah tampilan dari *file* suara “ba” yang dilihat dengan audacity. Disini bisa dilihat bahwa durasi *file* “ba” adalah 0.495 detik.



Gambar 3.7 File Ba.wav Dilihat dengan Software Audacity

```
function [durasiawal, pitchawal] = voicebank(voicepath)
switch voicepath
case 'a'
    durasiawal = 0.49
    pitchawal = 264.84
case {'ab'}
    durasiawal = 0.399
    pitchawal = 275.83
case 'ac'
    durasiawal = 0.49
    pitchawal = 273.22
case {'ad'}
    durasiawal = 0.399
    pitchawal = 270.99
case 'af'
    durasiawal = 0.49
    pitchawal = 261.669
case {'ag'}
    durasiawal = 0.399
    pitchawal = 268.397
case 'ah'
    durasiawal = 0.49
    pitchawal = 276.78
case {'aj'}
    durasiawal = 0.399
    pitchawal = 256.101
```

Gambar 3.8 Source Code Penyimpanan Durasi dan Pitch

Disini tidak semua *file* dicatat durasi dan *pitch* rata ratanya, hanya sebagian saja terutama yang akan digunakan dalam pengujian dengan *speech synthesizer*. Data

durasi dan *pitch* dicatat dalam fungsi *voicebank* dalam *file* matlab. Gambar 3.8 adalah sebagian dari kumpulan data durasi dan *pitch* awal yang ditulis dalam fungsi *voicebank*.

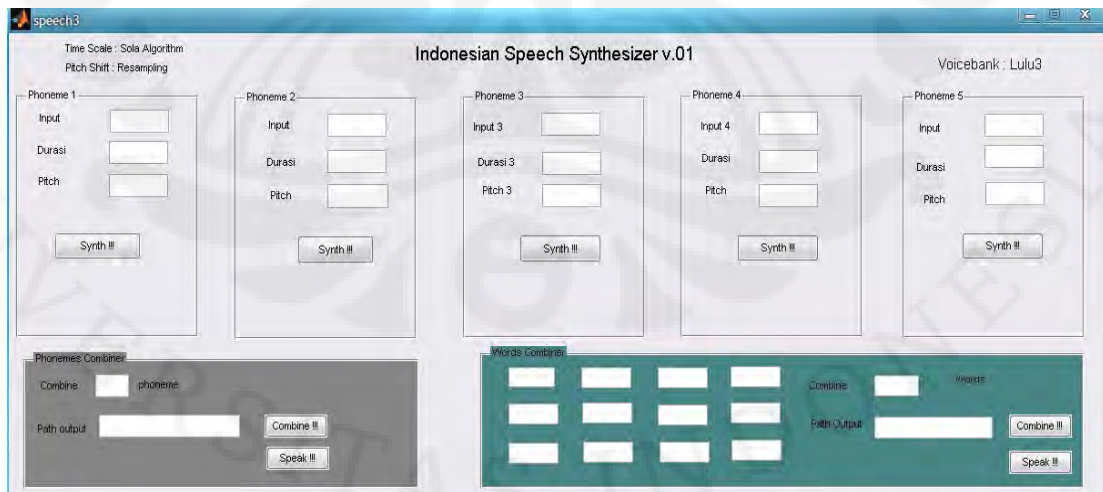
3.2 Sistem *Speech synthesizer*

Speech synthesizer adalah sebuah sistem yang dapat menghasilkan suara manusia. *Speech synthesizer* yang ada dalam skripsi ini dibuat dengan program matlab dan mempunyai kemampuan untuk dapat mengubah *pitch* dan durasi. Didalam skripsi ini dibuat dua buah *speech synthesizer* dengan spesifikasi sebagai berikut.

- *Speech synthesizer* 1 (menggunakan *speech engine* UTAU untuk melakukan *pitch shifting* dan *time scaling*)
- *Speech synthesizer* 2 (menggunakan *resampling* untuk melakukan *pitch shifting* dan algoritma SOLA untuk melakukan *time scaling*)

3.2.1 GUI *Speech synthesizer*

GUI *Speech synthesizer* yang digunakan disini dibuat dengan menggunakan matlab. Tampilan GUI bisa dilihat pada gambar 3.9

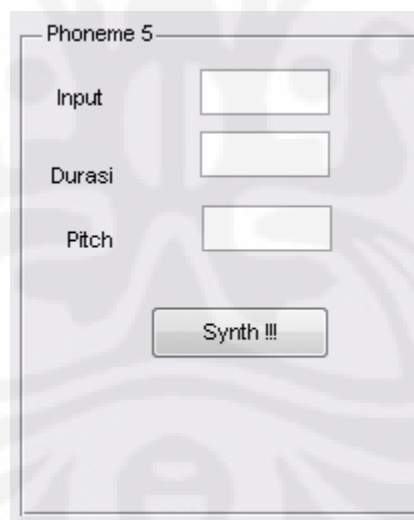


Gambar 3.9 GUI *Speech Synthesizer* 2

GUI Speech synthesizer 1 dan 2 memiliki tampilan yang sama, yang membedakan adalah *speech engine* yang digunakan untuk mengolah suaranya. *GUI speech synthesizer* ini mempunyai tiga modul, yaitu *phoneme synthesizer*, *Phonemes combiner*, dan *words combiner*. Modul *phoneme synthesizer* berfungsi untuk mengubah *pitch* dan durasi dari sebuah suara. Modul *Phonemes combiner* berguna untuk menggabungkan *file file* fonem yang sudah disintesis. Modul *Words combiner* berguna untuk menggabungkan kata yang sudah digabung oleh modul *Phonemes combiner*

3.2.1.1 Modul *Phoneme synthesizer*

Modul *phoneme synthesizer* ini berguna untuk mensintesis *file file wav* yang ada pada *voicebank* LULU3 dengan *pitch* dan durasi yang diatur oleh pengguna. Ada tiga *form* isian disini yaitu *input*, *pitch*, dan durasi. Gambar 3.10 adalah tampilan dari modul *phoneme synthesizer*.



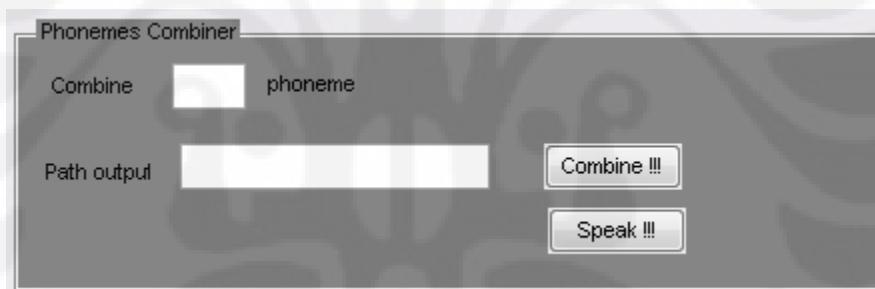
Gambar 3.10 Modul *Phoneme Synthesizer*

Pada masukan input, pengguna memasukkan nama dari *file* tanpa ekstensi apapun. Misalkan pengguna ingin mensintesis *file* suara “ba.wav” dari *database* LULU3. Maka yang dimasukkan kedalam input adalah “ba” tanpa tanda petik. Pada masukan durasi, pengguna memasukkan durasi yang diinginkan. Misalkan pengguna

ingin membuat suara “ba” dengan durasi 0.3 detik. Maka yang dimasukan ke dalam masukan durasi adalah angka “0.3” tanpa tanda petik. Pada masukan *pitch*, pengguna menggunakan *pitch* yang diinginkan. Misalkan pengguna ingin membuat suara “ba” dengan *pitch* 270 Hz. Maka yang dimasukan ke dalam masukan *pitch* adalah 270. Tombol “Synth !!!” adalah tombol untuk menjalankan modul membuat *file* yang diinginkan oleh user. *File GUI* “ps1.wav” akan dibuat di *directory* yang sama dengan direktori *GUI*.

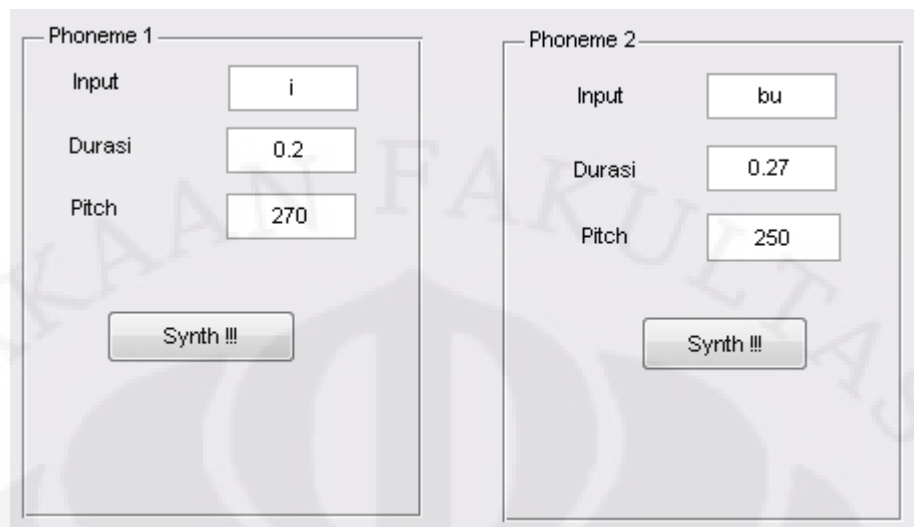
3.2.1.2 Modul *Phonemes Combiner*

Modul *phonemes combiner* berguna untuk menggabungkan suara suara yang sudah disintesisakan untuk membentuk satu kata. *Phonemes combiner* mampu menggabungkan sampai 5 suara sintesis yang telah disintesisakan dari 5 *phoneme synthesizer*. Gambar 3.11 adalah tampilan dari modul *phonemes combiner*.

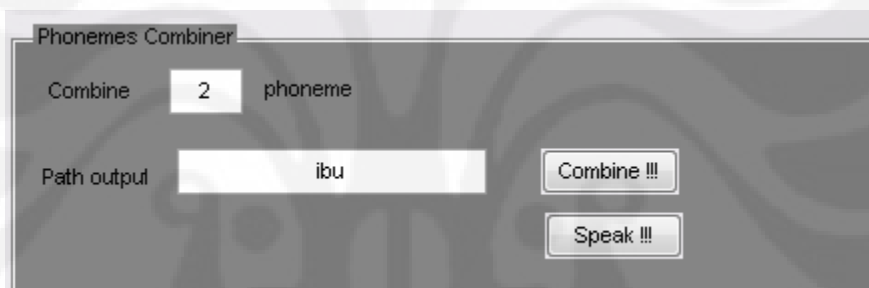


Gambar 3.11 Tampilan Modul *Phonemes Combiner*

Pada modul ini terdapat dua masukan yaitu masukan pertama yang menanyakan berapa jumlah fonem yang akan dimasukan Serta *path output* yang merupakan path tempat *file output* akan dibuat. Misalkan pengguna ingin membuat suara sintesis “ibu”. Maka pertama tama pengguna harus mensintesisakan suara “i” dan “bu” terlebih dahulu dengan *phoneme synthesizer*. Gambar 3.12 adalah ilustrasi pensintesisan suara “i” dan “bu”. Setelah disintesisakan dengan mengklik tombol “synth !!!” pada masing masing modul, langkah selanjutnya adalah menggabungkannya dengan modul *phonemes combiner*. Gambar 3.13 adalah ilustrasi penggabungan “I” dan “bu” dengan menggunakan modul *phonemes combiner*.



Gambar 3.12 Ilustrasi Pensintesisan Suara “I” dan “Bu”

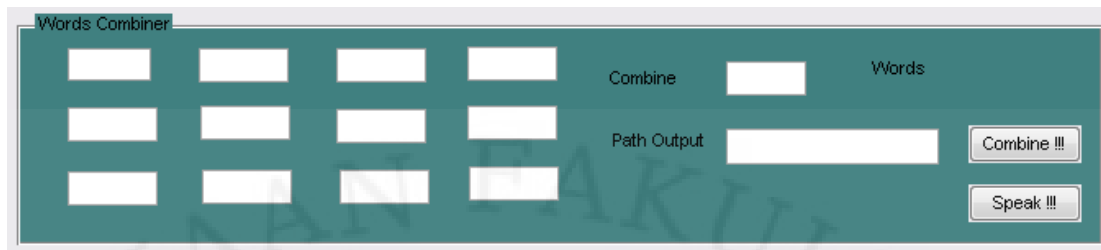


Gambar 3.13 Ilustrasi Penggabungan Suara “I” dan “bu”

Pada Gambar 3.13, inputan pertama dimasukan nilai 2 karena phoneme yang digabung adalah phonem dari modul *phoneme synthesizer* 1 dan 2. Inputan path *output* sebetulnya bebas diisi apapun. Namun untuk memudahkan sebaiknya diisi dengan nama yang sama dengan suara yang terkandung didalamnya. Ketika tombol “Combine !!!” di klik maka *file* “ibu.wav” akan muncul di direktori yang sama. Ketika tombol “Speak !!!” di klik maka *file* “ibu.wav” akan dimainkan.

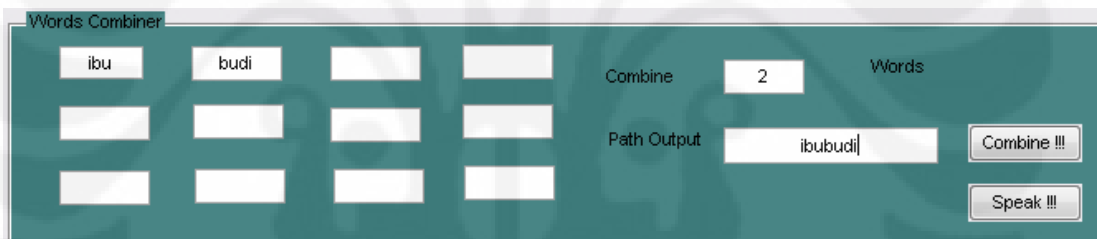
3.2.1.3 Modul *Words combiner*

Modul *words combiner* ini berfungsi untuk menggabungkan *file file* yang sudah dibuat dari modul *phonemes combiner*. Gambar 3.14 adalah tampilan dari modul *words combiner*.



Gambar 3.14 Modul *Words Combiner*

Modul *words combiner* ini mampu menggabungkan sampai 12 kata. Cara kerjanya mirip dengan modul *phonemes combiner* bedanya adalah *phonemes combiner* menggabungkan *GUI file* yang dibuat dari *phoneme synthesizer* sedangkan *words combiner* menggabungkan *file file* yang dihasilkan dari *phonemes combiner* sudah diberi nama unik masing masing. Misalkan pengguna ingin mensintesis ucapan "Ibu Budi". Maka modul ini diisi dengan tampilan seperti pada gambar 3.15



Gambar 3.15 Ilustrasi Pensintesisan "Ibu Budi" dengan *Words Combiner*

3.2.2 Algoritma *Speech synthesizer*

Dalam Skripsi ini, dibuat dua buah *speech synthesizer*. Untuk modul *words combiner* dan modul *phonemes combiner* kedua *speech synthesizer* menggunakan algoritma yang sama yaitu membaca nilai sinyal berdasarkan waktu dan kemudian meruntkannya dalam matriks dengan perintah *cat* pada *matlab*. Gambar 3.16 adalah *Source code* untuk menggabungkan *file file* wav dari program *speech synthesizer* dalam skripsi ini.

```

[y1,fs]=wavread('ps1.wav');
[y2,fs]=wavread('ps2.wav');
[y3,fs]=wavread('ps3.wav');
[y4,fs]=wavread('ps4.wav');
[y5,fs]=wavread('ps5.wav');

G=cat(1,y1,y2,y3,y4,y5);
wavwrite(G,44100,outputpath)

```

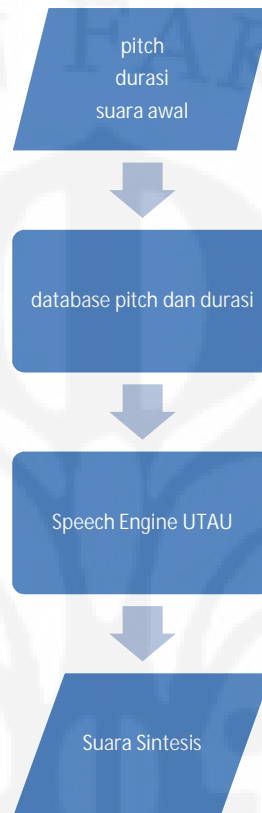
Gambar 3.16 *Source Code* untuk Menggabungkan 5 *File* *Wav*

Perbedaan antara *speech synthesizer* 1 dan 2 terdapat pada modul *phoneme synthesizer*. Modul *phoneme synthesizer* pada *speech synthesizer* 1 menggunakan *speech engine* UTAU untuk merubah *pitch* dan durasinya. Sedangkan modul *phoneme synthesizer* pada *speech synthesizer* 2 menggunakan Algoritma SOLA dan *resampling* untuk merubah durasi dan *pitch* nya.

3.2.2.1 Algoritma *Phoneme synthesizer* 1

Proses yang terjadi pada *speech synthesizer* 1 akan dijelaskan sebagai berikut. Pertama-tama pengguna akan memasukan nilai frekuensi *pitch*, durasi, dan suara yang diinginkan. Nilai frekuensi *pitch* yang dimasukan berkisar antara 200 – 300 hz. Nilai durasi yang diinginkan dimasukan berkisar dari 0.1 - 0.4 detik. Pembatasan ini dimaksudkan untuk menjaga kualitas suara agar tidak terlalu rusak, karena proses *time scaling* dan *pitch shifting* adalah proses yang merusak kualitas *file*. Untuk melihat suara suara apa saja yang dapat disintesisakan bisa dilihat *folder* “voice/lulu3”. *Input* akan dimasukan sesuai dengan nama *file* yang ada disana tanpa ekstensi apapun (misal: a, ba, bang, in). Setelah data data tersebut dimasukan, sistem akan mencocokkan kedalam *database pitch* dan durasi. Kemudian sistem akan mengeluarkan faktor *pitch percent* dan *velocity*. *Pitch percent* adalah perbandingan antara frekuensi input dan frekuensi dari *file* suara yang ada pada *voicebank* dikali dengan seratus. *Velocity* adalah perbandingan durasi *file* suara yang diinginkan dengan

durasi dari *file* suara yang ada dalam *voicebank*. *Speech synthesizer* 1 mempunyai *flowchart* seperti gambar 3.17.



Gambar 3.17 *Flowchart Speech Synthesizer 1*

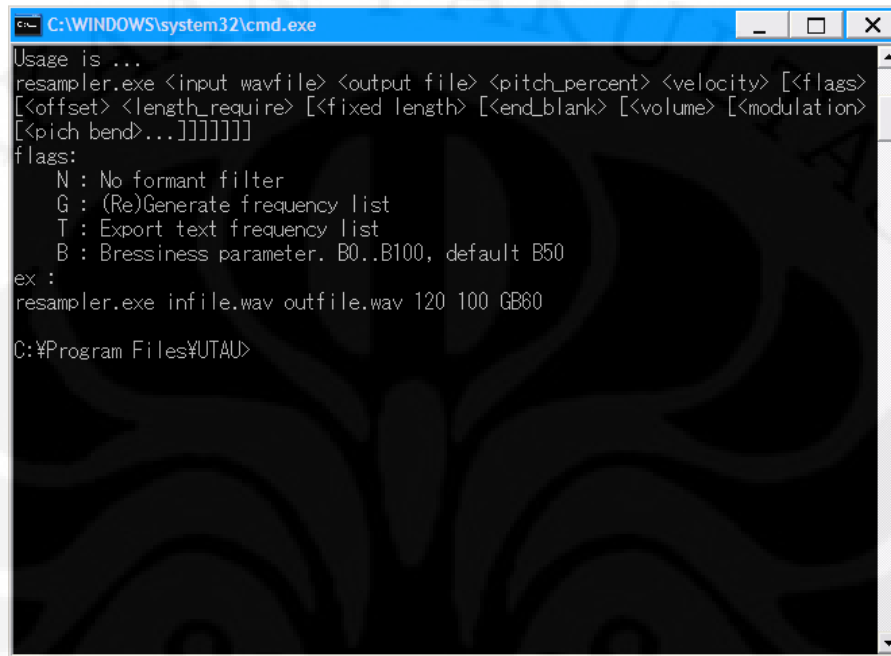
$$h = 100. \quad (3.1)$$

$$= 100. \quad (3.2)$$

Setelah nilai *pitch percent* dan *velocity* didapatkan maka selanjutnya *file* suara akan disintesis dengan menggunakan *Speech Engine UTAU* berdasarkan nilai *Pitch percent* dan *velocity* tersebut.

UTAU memiliki *speech engine* yang bernama *resampler.exe*. *Speech engine* ini dibuat oleh Mr.Ameya, yang juga merupakan pembuat *software singing synthesizer UTAU*. *Speech engine* ini berfungsi untuk merubah *pitch*, *velocity*,

melakukan pendeteksian *pitch*, melakukan *formant filter*, dan sebagainya. *Speech engine* ini bisa dijalankan melalui *command prompt windows*. Gambar 3.18 adalah tampilan *speech engine* UTAU dalam *command prompt*.



```

C:\WINDOWS\system32\cmd.exe
Usage is ...
resampler.exe <input wavfile> <output file> <pitch_percent> <velocity> [<flags>
[<offset> <length_require> [<fixed length> [<end_blank> [<volume> [<modulation>
[<pich bend>...]]]]]]]]
flags:
N : No formant filter
G : (Re)Generate frequency list
T : Export text frequency list
B : Bressiness parameter. B0..B100, default B50
ex :
resampler.exe infile.wav outfile.wav 120 100 GB60
C:\Program Files\UTAU>

```

Gambar 3.18 Tampilan *Speech engine* UTAU dalam *Command Prompt*

Untuk menjalankan *Speech engine* ini maka sebuah *command* harus ditulis dengan *format* seperti yang terlihat pada gambar 3.19. *Speech engine* UTAU dalam skripsi ini digunakan hanya untuk merubah *pitch* dan durasi saja. *Command* yang digunakan untuk merubah *pitch* bisa dilihat pada gambar 3.19

```
C:\Program Files\UTAU>resampler.exe voice/lulu3/a.wav a300hz 120 100
```

Gambar 3.19 Perintah untuk Mengubah *Pitch* pada *Speech Engine* UTAU

Gambar 3.19 adalah *command* untuk memerintahkan *speech engine* utau untuk mensintesisakan suatu file suara. Kode “voice/lulu/a.wav” adalah path file suara yang akan disintesisakan. Kode “a300hz” adalah nama *file output* yang akan dibuat. Kode “120” adalah nilai *pitch* percent, disini berarti *pitch* akan lebih tinggi 1.2 kali dari

pitch awal. Kode “100” adalah nilai *velocity*. Angka 100 menyatakan durasi tidak akan berubah.

Speech synthesizer dengan *speech Engine* UTAU kurang akurat dalam *pitch shifting* dan *time scaling*. Sebuah percobaan saya lakukan dimana penyedia *voicebank* “lulu” diminta mengucapkan kalimat “dimana budi tinggal”. Setelah itu kalimat tersebut saya trimming sampai berbentuk fonem fonem dan kemudian dideteksi *pitch* dan durasinya. Tabel 3.5 adalah daftar *pitch* dan durasi dari kalimat “dimana budi tinggal” yang diucapkan oleh penyedia suara *voicebank* “lulu”.

Tabel 3.5 Daftar *Pitch* dan Durasi Dari Kalimat “Dimana Budi Tinggal” yang Diucapkan oleh Penyedia Suara *Voicebank* “LULU3”.

Fonem	Durasi (s)	<i>Pitch</i> (Hz)
di	0.184	224.92
ma	0.136	316.36
na	0.15	341.29
bu	0.13	225.68
di	0.087	203.85
ting	0.237	185.59
gal	0.282	189.04

Pensintesan suara kemudian dilakukan dengan menggunakan *voicebank* “lulu3”. Disini saya berusaha membuat file suara dengan *pitch* dan durasi yang diatur sama dengan kalimat “dimana budi tinggal” yang diucapkan oleh penyedia *voicebank* “lulu3”. Tabel 3.6 adalah data perbandingan antara *pitch* suara asli dengan *pitch* suara sintesis hasil *phoneme synthesizer* pada *speech synthesizer* 1. Disini bisa dilihat bahwa terdapat error atas akurasi *pitch* yang cukup besar yaitu sampai dengan 8.7%. Dari tabel 3.6 dapat dilihat bahwa *Speech synthesizer* yang saya rancang dengan menggunakan *speech engine* UTAU memiliki akurasi *pitch* yang buruk.

Tabel 3.6 Perbandingan antara *Pitch* Suara Asli dengan *Pitch* Suara Sintesis Hasil Pensintesisan dengan *Phoneme Synthesizer* pada *Speech Synthesizer 1*

fonem	frekuensi <i>pitch</i> suara asli (Hz)	frekuensi <i>pitch</i> suara sintesis (Hz)	% error suara sintesis terhadap suara asli
di	224.92	205.32	8.714209%
ma	316.36	296.678	6.221393%
na	341.29	326.39	4.365789%
bu	225.68	225.47	0.093052%
di	203.85	186.33	8.594555%
ting	185.59	189.22	1.955924%
gal	189.04	174.61	7.633305%
Error rata rata			5.37%

Tabel 3.7 adalah data perbandingan antara durasi suara asli dengan durasi suara sintesis hasil *speech synthesizer 1*. Disini bisa dilihat bahwa terdapat error atas akurasi durasi yang cukup besar yaitu sampai dengan 55.17%. Dari tabel 3.7 dapat dilihat bahwa *Speech synthesizer* yang saya rancang dengan menggunakan speech engine UTAU memiliki akurasi durasi yang buruk.

Tabel 3.7 Perbandingan antara Durasi Suara Asli dengan Durasi Suara Sintesis Hasil Pensintesisan dengan *Phoneme synthesizer* pada *Speech synthesizer 1*

Fonem	Durasi Suara Asli (s)	Durasi Suara Sintesis (s)	% Error Durasi Suara Sintesis Terhadap Suara Asli
di	0.184	0.208	13.04347826%
ma	0.136	0.13	4.411764706%
na	0.15	0.159	6%
bu	0.13	0.127	2.307692308%
di	0.087	0.039	55.17241379%
ting	0.237	0.246	3.797468354%
gal	0.282	0.364	29.07801418%
Error rata rata			16.25869023%

Persentase error frekuensi pitch pada tabel 3.6, 3.8, dan 3.10 dihitung dengan rumus:

$$\% \text{ error} = \frac{|f_{\text{target}} - f_{\text{hasil}}|}{f_{\text{target}}} \times 100 \quad (3.3)$$

Sedangkan persentase error durasi pada tabel 3.7, 3.9, dan 3.11 dihitung dengan rumus:

$$\% \text{ error} = \frac{|T_{\text{target}} - T_{\text{hasil}}|}{T_{\text{target}}} \times 100 \quad (3.4)$$

3.2.2.2 Algoritma *Phoneme synthesizer 2*

Untuk *Speech synthesizer 2*, Prosesnya sama dengan *speech synthesizer 1* Pertama-tama pengguna akan memasukan nilai frekuensi *pitch*, durasi, dan suara yang diinginkan. Nilai frekuensi *pitch* yang dimasukan berkisar antara 200 – 300 hz. Nilai durasi yang diinginkan dimasukan berkisar dari 0.1 - 0.4 detik. Gambar 3.20 adalah *flowchart* dari sistem *phoneme synthesizer* pada *speech synthesizer 2*.



Gambar 3.20 *Flowchart Speech synthesizer 2*

Untuk daftar file suara yang tersedia dapat dilihat pada *folder* “voice/lulu3/”. *Inputan* akan dimasukan sesuai dengan nama *file* yang ada disana (misal: a, ba, bang, in). Setelah data data tersebut dimasukan sistem akan mencocokkan kedalam *database pitch* dan durasi. Kemudian sistem akan mengeluarkan faktor *beta* dan *alpha*. Beta adalah perbandingan antara frekuensi input dan frekuensi dari *file* suara yang ada pada *voicebank*. Faktor Alpha adalah perbandingan durasi *file* suara yang diinginkan dengan durasi dari *file* suara yang ada dalam *voicebank*. Faktor alpha dan beta dicari dengan menggunakan persamaan 3.5 dan 3.6.

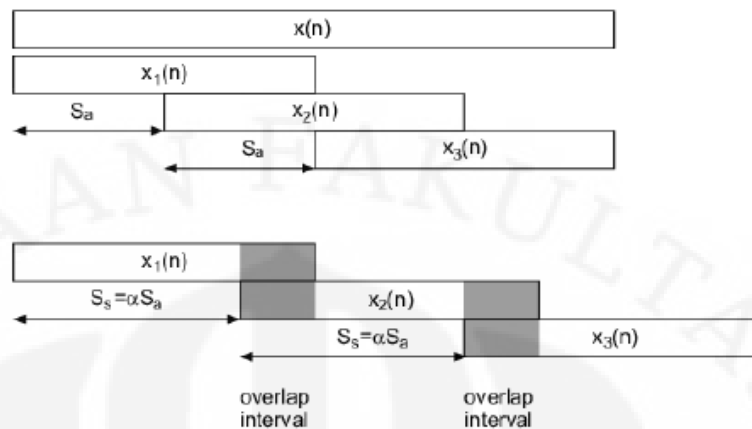
$$h = \frac{f_{input}}{f_{file}} \quad (3.5)$$

$$\alpha = \frac{d_{input}}{d_{file}} \quad (3.6)$$

Setelah nilai beta dan alpha didapatkan maka selanjutnya *file* suara akan disintesiskan dengan menggunakan algoritma SOLA dan *resampling* berdasarkan nilai alpha dan beta tersebut.

Synchronized overlap and add atau biasa disingkat SOLA, adalah sebuah algoritma untuk merubah durasi sebuah *file* suara tanpa merubah *pitch*nya, algoritmanya bekerja sebagai berikut:

1. Sinyal input dibagi menjadi beberapa *segment* dengan panjang yang sama yang di *overlap* satu sama lain
2. Segmen *segment* yang ter-*overlap* digeser sesuai dengan *time scaling factor* yang diinginkan (α)
3. Area *overlap* pada setiap interval dicari untuk mendapatkan *discrete time lag*-nya pada kemiripan maksimum. Pada saat didapat kemiripan maksimum, *segment segment* yang *overlap* diberikan efek *fade in* dan *fade out* untuk menghindari perubahan suara yang drastis. *segment-segment* tersebut kemudian digabungkan



Gambar 3.21: Algoritma SOLA [4]

Gambar 3.21 adalah algoritma ilustrasi dari algoritma SOLA. Pada gambar bisa dilihat bahwa panjang SS tergantung pada faktor alfa (α).

Resampling adalah salah satu algoritma untuk melakukan *pitch shifting* pada file suara. *Pitch shifting* adalah algoritma yang berfungsi mengubah *pitch* suatu unit menjadi lebih tinggi atau menjadi lebih rendah. Cara yang paling mudah untuk merubah frekuensi *pitch* adalah dengan melakukan *resampling* seperti yang dilakukan pada *speech synthesizer 2*. *Resampling* adalah mengubah frekuensi *sample (framerate)* dari *file* suara tersebut, Cara ini memang bisa dengan mudah mengubah *pitch* tapi apabila kita merubah frekuensi *sampling* dari sebuah *file* suara, maka *file* suara tersebut akan berubah durasinya. Oleh karena itulah selain merubah frame ratenya sebesar n kali, kita juga harus memanjangkan suara tersebut sebesar n kali agar *file* suara tersebut tetap memiliki durasi yang sama. Karena frekuensi *pitch* pada *database* suara diketahui, untuk merubah *pitch* ke suatu nilai, pertama tama kita harus mengubah *framerate*-nya terlebih dahulu. Persamaan 3.6 adalah rumus untuk menghitung framerate dalam proses *pitch shifting*

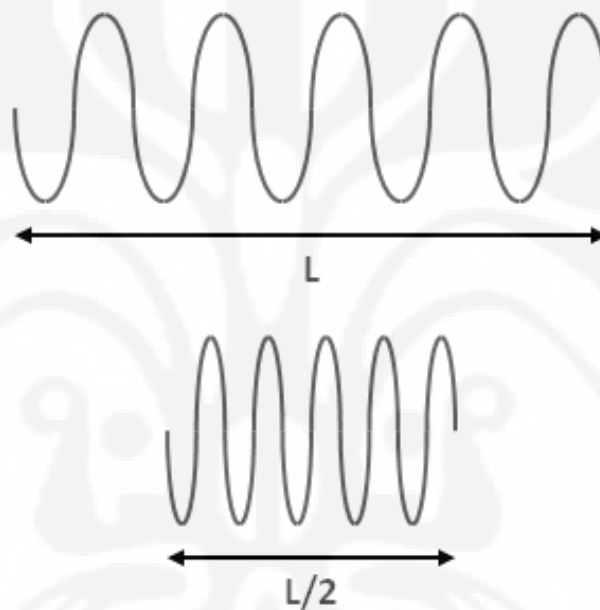
$$f_{s_{baru}} = \frac{fp_{baru}}{fp_{awal}} \times f_{s_{awal}} \quad (3.6)$$

keterangan:

f_s = frekuensi sample (*framerate*)

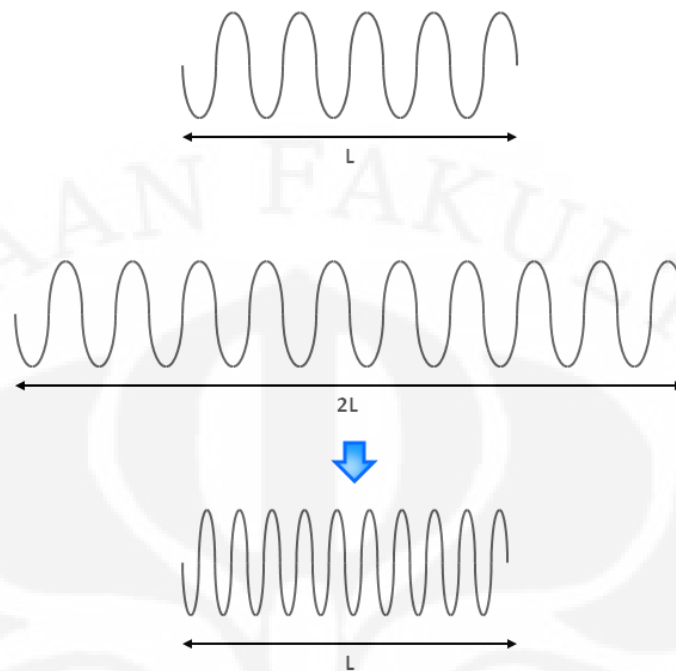
f_p = frekuensi *pitch*

Gambar 3.22 adalah ilustrasi dari algoritma *pitch shifting*. Misalkan jika kita merubah frekuensi *sample* sebesar 2 kali, maka frekuensi *pitch* akan meningkat sebesar 2 kali, dan durasi suaranya akan menjadi setengahnya.



Gambar 3.22 Ilustrasi Peningkatan *Framerate* [14]

Gambar 3.23 adalah ilustrasi dari algoritma *pitch shifting*. Pada gambar 3.23 sebuah file suara akan diubah frekuensi *pitch* nya sebesar dua kali lipat. Maka untuk melakukannya file suara tersebut harus diperpanjang sebesar dua kali lipat dulu dengan algoritma *time scaling*. Kemudian, barulah *resampling* dilakukan. Disini akan didapat file suara dengan panjang yang sama dengan file awal namun frekuensinya berubah dua kali lipat



Gambar 3.23: Proses *Pitch Shifting* [14]

Speech synthesizer dengan algoritma SOLA-Resampling yang ada dalam skripsi ini kurang akurat dalam *pitch shifting* dan *time scaling*. Dengan percobaan yang sama dengan yang dilakukan untuk menguji akurasi *phoneme synthesizer 1*. Disini saya mencoba mensintesis suara “dimana budi tinggal” dengan *speech synthesizer 2* dengan menggunakan nilai *pitch* dan durasi yang ada pada tabel 3.5.

Tabel 3.8 adalah tabel perbandingan antara *pitch* suara asli dengan *pitch* suara sintesis hasil *phoneme synthesizer* pada *speech synthesizer 2*. Disini bisa dilihat bahwa terdapat error atas akurasi *pitch* yang cukup besar yaitu sampai dengan 10.74%. Tabel 3.9 adalah data perbandingan antara durasi suara asli dengan durasi suara sintesis hasil *phoneme synthesizer* pada *speech synthesizer 2*. Disini bisa dilihat bahwa terdapat error atas akurasi durasi yang cukup besar yaitu sampai dengan 59.77%. Dari tabel 3.8 dan tabel 3.9 dapat dilihat bahwa *Speech synthesizer* yang saya rancang dengan menggunakan algoritma SOLA-Resampling memiliki akurasi durasi dan *pitch* yang buruk.

Tabel 3.8 Perbandingan antara *Pitch* Suara Asli dengan *Pitch* Suara Sintesis Hasil Pensintesisan Dengan *Phoneme Synthesizer* Pada *Speech Synthesizer 2*

fonem	frekuensi <i>pitch</i> suara asli (Hz)	frekuensi <i>pitch</i> suara sintesis (Hz)	% error suara sintesis terhadap suara asli
di	224.92	223.779	0.507291481%
ma	316.36	331.422	4.761031736%
na	341.29	353.87	3.686014826%
bu	225.68	207.97	7.847394541%
di	203.85	199.86	1.95732156%
ting	185.59	172.047	7.297268172%
gal	189.04	168.73	10.74375793%
Error rata rata			5.257154322%

Tabel 3.9 Perbandingan antara Durasi Suara Asli dengan Durasi Suara Sintesis Hasil Pensintesisan dengan *Phoneme synthesizer* pada *Speech synthesizer 2*

Fonem	Durasi Suara Asli (s)	Durasi Suara Sintesis (s)	% Error Durasi Suara Sintesis Terhadap Suara Asli
di	0.184	0.228	23.91304348%
ma	0.136	0.16	17.64705882%
na	0.15	0.184	22.66666667%
bu	0.13	0.182	40%
di	0.087	0.139	59.77011494%
ting	0.237	0.295	24.47257384%
gal	0.282	0.4	41.84397163%
Error rata rata			32.90191848%

Tabel 3.8 dan tabel 3.9 disini menunjukkan nilai *pitch* dan durasi yang dikeluarkan oleh *phoneme synthesizer 2*. Sedangkan untuk hasil akhir dari suara yang dihasilkan oleh *speech synthesizer 2* ada pada tabel 3.10 dan tabel 3.11. Disini ada sebuah *bugs* yang disebabkan oleh algoritma resampling pada saat penggabungan. *Bugs* tersebut adalah pada *speech synthesizer* ini tidak ada sistem yang mengatur *framerate* per fonem. Jadi ketika fonem fonem tersebut digabung menjadi sebuah file

suara “dimana budi tinggal”. File suara tersebut hanya mempunyai satu framerate, tidak beragam seperti yang terjadi pada keluaran *phoneme synthesizer* pada *speech synthesizer 2*. Sehingga pitch dan durasi akan berubah lagi setiap phonemanya. Tabel 3.10 adalah tabel perbandingan antara *pitch* suara asli dengan *pitch* suara sintesis hasil *speech synthesizer 2*. Tabel 3.11 adalah tabel perbandingan antara durasi suara asli dan durasi suara sintesis hasil *speech synthesizer 2*.

Tabel 3.10 Perbandingan antara *Pitch* Suara Asli dengan Suara Sintesis Hasil Pensintesisan dengan *Speech Synthesizer 2*

fonem	frekuensi <i>pitch</i> suara asli (Hz)	frekuensi <i>pitch</i> suara sintesis (Hz)	% error suara sintesis terhadap suara asli
di	224.92	245.435	9.121020807%
ma	316.36	279.995	11.49481603%
na	341.29	270.4	20.77119166%
bu	225.68	248.705	10.20249911%
di	203.85	239.133	17.30831494%
ting	185.59	188.89	1.778113045%
gal	189.04	174.7	7.585696149%
Error rata rata			11.18023596%

Tabel 3.11 Perbandingan antara Durasi Suara Asli dengan Durasi Suara Sintesis Hasil Pensintesisan dengan *Speech synthesizer 2*

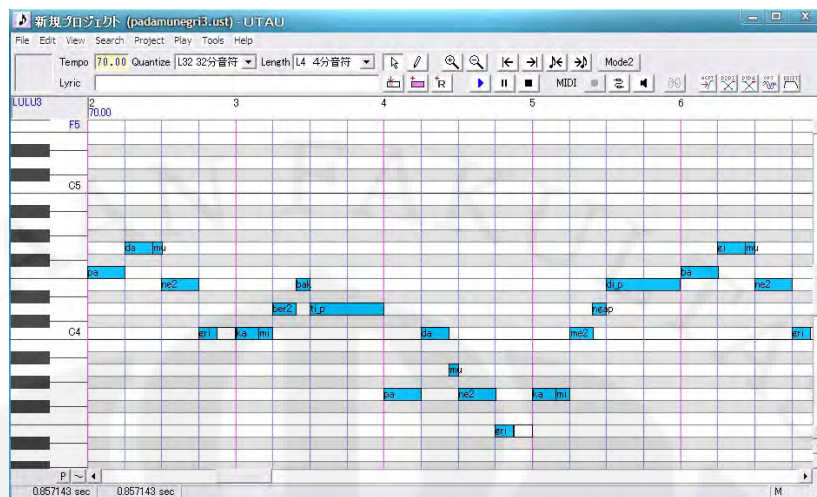
Fonem	Durasi Suara Asli (s)	Durasi Suara Sintesis (s)	% Error Durasi Suara Sintesis Terhadap Suara Asli
di	0.184	0.206	11.95652174%
ma	0.136	0.201	47.79411765%
na	0.15	0.247	64.66666667%
bu	0.13	0.1534	18%
di	0.087	0.1125	29.31034483%
ting	0.237	0.25	5.485232068%
gal	0.282	0.365	29.43262411%
Error rata rata			29.52078672%

Pada tabel 3.10 bisa dilihat bahwa hasil pitch suara sintesis berbeda dengan hasil pitch suara sintesis pada tabel 3.8. Pada tabel 3.10 juga terdapat error atas akurasi *pitch* yang cukup besar yaitu sampai dengan 20.77%. Sedangkan tabel 3.11 adalah data perbandingan antara durasi suara asli dengan durasi suara sintesis hasil *speech synthesizer 2*. Pada tabel 3.11 bisa dilihat bahwa hasil durasi suara sintesis pada tabel 3.11 berbeda dengan hasil durasi suara sintesis pada tabel 3.9. Pada tabel 3.11 bisa dilihat bahwa terdapat error atas akurasi durasi yang cukup besar yaitu sampai dengan 64.67%. Dari tabel 3.10 dan 3.11 dapat dilihat bahwa *Speech synthesizer 2* yang saya rancang dengan menggunakan algoritma SOLA-Resampling memiliki akurasi durasi dan pitch yang buruk.

3.3 Sistem Singing synthesizer

Singing synthesizer adalah sebuah sistem yang dapat menghasilkan suara yang mempunyai nada layaknya suara manusia yang sedang bernyanyi. *Singing synthesizer* yang digunakan dalam skripsi ini menggunakan *software* UTAU, sebuah *software* dari jepang yang berlisensi shareware yang dibuat oleh Mr.Ameya pada tahun 2008. Sampai saat ini Mr.Ameya terus mengembangkan *software* ini. Rilis terakhir adalah versi 2.7.6 yang dirilis pada tanggal 25 Mei 2010. *Software* ini cukup populer di kalangan penggemar anime dan manga. Sampai saat ini sudah cukup banyak orang-orang dari seluruh dunia yang sudah membuat *voicebank* untuk dipergunakan dalam UTAU ini termasuk orang Indonesia. Sayangnya *voicebank* yang dibuat kurang dapat menyanyikan lagu dalam bahasa Indonesia. *Voicebank* tersebut dibuat lebih untuk menyanyikan lagu-lagu dalam bahasa Jepang.

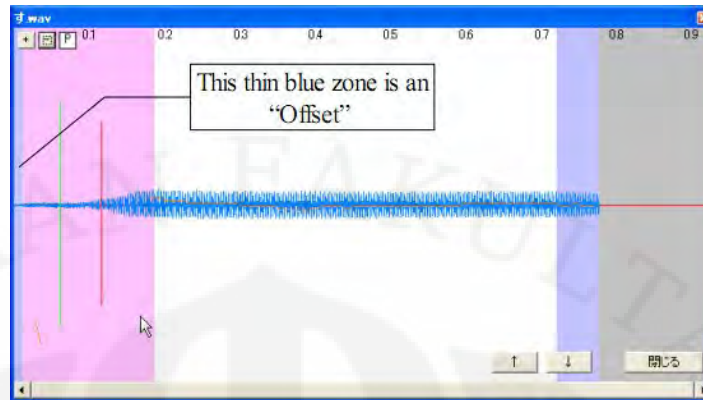
Gambar 3.24 adalah tampilan lagu “padamu negri” yang dibuat dengan menggunakan *software* UTAU. Pada bagian atas terdapat tempo untuk mengatur cepat lambatnya lagu. pada bagian bawah terdapat *score* yang merupakan tempat untuk menaruh lirik dan notasi lagu.



Gambar 3.24 Pembuatan Nyanyian Sintesis “Padamu Negri” pada UTAU

Format voicebank yang digunakan dalam *software* UTAU ini menggunakan kumpulan *file wav*. Untuk memasukan *voicebank* yang sudah dibuat kedalam sistem UTAU, pertama-tama kumpulan *file file* unit (V, VK, KV, dan KVK) hasil rekaman harus dimasukan dalam satu *folder* yang sudah diberi nama sesuai dengan nama *voicebank*nya. Selanjutnya *folder* tersebut harus dipindahkan ke direktori “voice” pada UTAU (“C:\program files\UTAU\voice”). Selanjutnya *file* “oto.ini” harus di buat. *file* ini akan mengindeks seluruh *file* suara yang sudah ada beserta data *alias*, *offset*, *cutoff*, *preutterance*, *overlap*, dan *consonant*. Penjelasan mengenai konstanta konstanta ini akan dijelaskan sebagai berikut.

- *alias* adalah nama lain yang dari suatu *filesuara*. biasanya ditulis dalam aksara hiragana apabila *filesuara* tersebut aslinya mempunyai nama dalam huruf latin. atau bisa saja dalam kasus sebaliknya *file* tersebut mempunyai nama *file* dalam hiragana sedangkan aliasnya ditulis dalam huruf latin.
- *offset* adalah batas awal (dalam mili detik) dari sebuah *file* suara. Dalam *voicebank* yang dibuat dalam skripsi ini, seluruh *file* suara akan diset *null* nilai *offset* nya. Gambar 3.25 adalah ilustrasi parameter *offset* pada UTAU

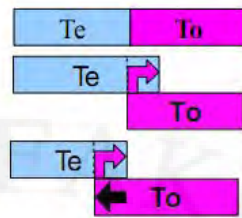


Gambar 3.25 : Ilustrasi Parameter *Offset* dalam UTAU [7]

- cutoff adalah batas akhir (dalam mili detik) dari sebuah *file* suara. Dalam *voicebank* yang dibuat dalam skripsi ini, seluruh *file* suara akan diset *null* nilai cutoff nya
- consonant adalah perbatasan (dalam mili detik) dari konsonan dan vokal dalam sebuah *file* suara. Dalam UTAU, suara sebelum nilai konsonan tidak akan mengalami pemanjangan.
- *preutterance* adalah batas *overlapping* awal (dalam mili detik) dari sebuah *file* suara. Ilustrasi dari *preutterance* dijelaskan dalam gambar 3.28 Pada gambar 3.28 gambar pertama adalah gambar dalam score musik pada UTAU dimana ada suara "te" dan "to". Apabila tidak diberikan nilai *preutterance*. Maka suara akan berbunyi normal. Namun apabila diberikan nilai *preutterance* (misal 100), maka suara "to" tersebut akan mulai 100 milidetik lebih awal dan durasi suara to akan berkurang 100 milidetik.



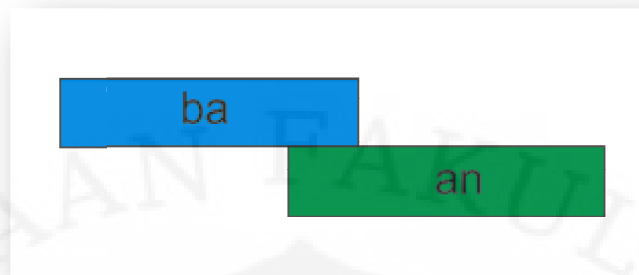
Gambar 3.26 Ilustrasi *Preutterance* [7]



Gambar 3.27 Ilustrasi *Preutterance* dan *Overlap* pada UTAU [7]

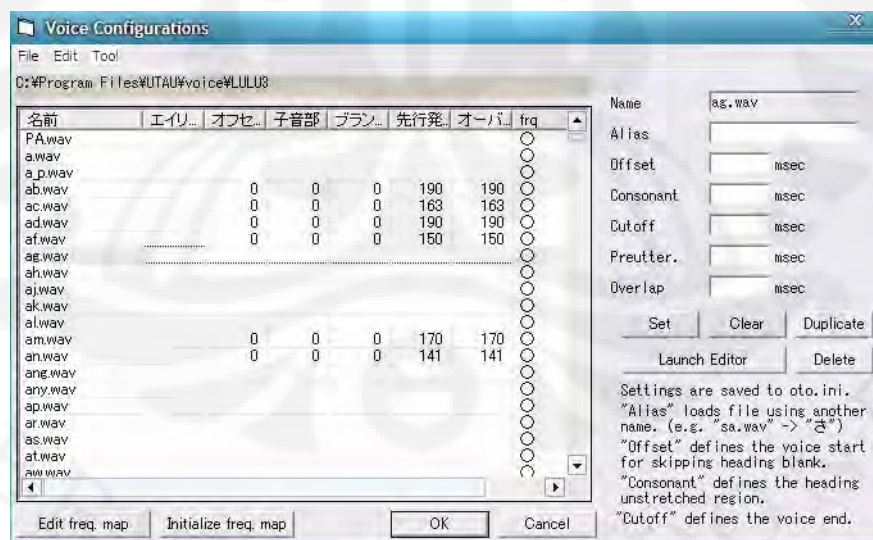
- *overlap* adalah batas *overlapping* akhir (dalam mili detik) dari sebuah *file* suara yang ada sebelum *file* suara ini. Konsep nilai *overlap* ini akan dijelaskan pada gambar 3.29. Pada gambar 3.29, gambar pertama adalah gambar normal tanpa ada nilai *overlap* maupun *preutterance*. Pada gambar kedua, nilai *overlap* pada “to” diset (misal 50), dengan pengesetan ini suara sebelum “to” yang dalam hal ini adalah “te” akan dipanjangkan sampai batas *overlap* dari “to” yaitu sebesar 50 milidetik. Namun tidak seperti *preutterance* yang memendekan suara *file* lainnya. Suara “to” tidak mengalami pemendekan melainkan tetap dimainkan juga. Artinya suara te dan to akan berbunyi secara bersamaan selama 50 milidetik. Pada gambar ketiga, *preutterance* dan *overlap* diset (misal *preutterance* = 100 *overlap* = 50). Dalam kasus ini maka suara “te” akan diperpanjang 50 ms, dan suara to akan dimulai lebih awal sebesar 100 ms. *Preutterance* dan *overlap* digunakan karena dalam bahasa jepang ada beberapa suara yang harus dimulai terlebih dahulu sebelum benar benar masuk ketukan. [7]

Pada *voicebank* yang dibuat dalam skripsi ini, *preutterance* dan *overlap* dibuat untuk *voicebank* yang berformat VK. Ilustrasi penggunaan fitur *preutterance* dan *overlap* untuk bahasa Indonesia bisa dilihat pada gambar 3.28. Pada gambar 3.28 bisa dilihat bahwa “ba” dan “an” digabung untuk menghasilkan bunyi ban.



Gambar 3.28 Penggunaan *Preutterance* dan *Overlap*

Gambar 3.29 adalah tampilan *GUI* untuk merubah nilai nilai pada alias, consonant, *offset*, *cutoff*, *preutterance*, dan *overlap* dapat diubah secara manual pada file “oto.ini” dengan menggunakan text editor atau dengan menggunakan *GUI* yang disediakan oleh UTAU. Gambar 3.29 adalah tampilan *GUI* yang disediakan UTAU untuk mengubah konstanta konstanta pada sebuah file suara “oto.ini” akan dibuat secara otomatis apabila user menekan tombol ok yang terlihat di *GUI* pada gambar 3.29



Gambar 3.29 *GUI* pada UTAU untuk Mengubah Parameter *File* Suara

Algoritma *pitch* shifting pada Software UTAU sangat baik. Tabel 3.12 adalah pengetesan yang saya lakukan dengan mengecek frekuensi *pitch* dari setiap fonem

dari lagu padamu negeri hasil sintesis dengan menggunakan *singing synthesizer* UTAU.

Tabel 3.12: Perbandingan Frekuensi *Pitch* Nada dan Frekuensi *Pitch* Suara yang Dihasilkan UTAU

fonem	pa	da	mu	ne	gri
frekuensi yang dihasilkan (Hz)	195.26	261.34	220.06	196.44	164.44
frekuensi nada seharusnya (Hz)	G3 (196)	C4(261.6)	A3(220)	G3(196)	E3 (164.8)
% error frekuensi utau terhadap frekuensi nada	0.3775%	0.099%	0.029%	0.224%	0.22%

Persentase error dihitung dengan rumus:

$$\% \text{ error} = \frac{| \text{Nilai Not} - \text{Durasi File yang Dihasilkan UTAU} |}{\text{Durasi seharusnya}} \quad (3.7)$$

Untuk *Time scaling* nya, UTAU juga memiliki akurasi *time scale* yang baik. Tabel 3.13 adalah hasil pengetesan yang saya lakukan dengan mengecek durasi dari setiap fonem dari lagu padamu negeri hasil sintesis dengan menggunakan *singing synthesizer* UTAU.

Tabel 3.13: Perbandingan Durasi Nada Seharusnya dan Durasi File Suara yang Dihasilkan UTAU

Fonem	Nilai Not	Durasi seharusnya	Durasi File yang Dihasilkan UTAU	%Error Perubahan Durasi oleh UTAU
pa	1	0.857142857	0.8566	0.063333333%
da	0.75	0.642857143	0.64115	0.265555556%
mu	0.25	0.214285714	0.21155	1.276666667%
ne	1	0.857142857	0.8548	0.273333333%
gri	0.5	0.428571429	0.427	0.366666667%
rata rata error				0.449111111%

BAB 4

SIMULASI DAN ANALISA

4.1 Simulasi Penggunaan Fonem Berformat KVK Pada *Speech Synthesizer*

Pada bagian ini, ingin dilihat apakah dalam bahasa Indonesia suara sintesis yang dihasilkan oleh *voicebank* yang berformat KVK, KV, VK, dan V lebih baik dari suara sintesis yang dihasilkan dari *voicebank* yang hanya berformat V, VK, dan KV saja. Disini dibuat 16 file suara yang berisi 4 macam ucapan dari *voicebank* LULU3 dengan menggunakan dua buah *speech synthesizer*. Ucapan ucapan tersebut adalah “selamat dikejar”, “maling kambing”, “timun emas”, dan “taring macan”. Pitch dan durasi setiap fonem diatur beragam dimana pitch berkisar antara 200 – 300 dan durasi berkisar dari 0.1 detik sampai 0.4 detik. Ucapan ucapan tersebut dibuat dengan empat metode. yaitu:

- a. Metode 1: Menggunakan *speech synthesizer* 1 dan menggunakan *voicebank* “LULU3” lengkap.
- b. Metode 2: Menggunakan *speech synthesizer* 1 dan menggunakan *voicebank* “LULU3 “tanpa KVK.
- c. Metode 3: Menggunakan *speech synthesizer* 2 dan menggunakan *voicebank* “LULU3” lengkap.
- d. Metode 4: Menggunakan *speech synthesizer* 2 dan menggunakan *voicebank* LULU3 tanpa KVK.

Gambar 4.1 merupakan form yang akan diisi oleh responden. Disini responden dipilih dari beberapa mahasiswa jurusan teknik elektro dan teknik komputer Universitas Indonesia dari angkatan 2006 – 2008 yang berkebangsaan Indonesia. Pada form tersedia 4 baris ucapan dan empat kolom jenis metode. Ucapan akan diperdengarkan berurutan dari yang dibuat dengan metode 1 sampai yang dibuat dengan metode 4. Responden kemudian akan mengurutkan dari ucapan yang terdengar paling jelas sampai yang terdengar paling tidak jelas.

form uji speech synthesizer

Nama: _____ Usia : _____

Jenis Kelamin: _____

Tentukan urutan kalimat dari yang paling jelas terdengar sampai yang paling tidak jelas

kalimat	1	2	3	4
selamat dikejar				
maling kambing				
timun emas				
taring macan				

Gambar 4.1 Form isian Uji *Speech synthesizer*

Dari hasil survey, Penulis kemudian membandingkan urutan kejelasan antara kolom 1 dengan kolom 2, kolom 3 dengan kolom 4, dan kolom 2 dengan kolom 3. Tabel 4.1 adalah data penilaian subjectif responden terhadap kejelasan antara kalimat yang dibuat dengan metode 1 dengan kalimat yang dibuat dengan metode 2.

Tabel 4.1 Perbandingan pilihan responden atas kejelasan suara antara yang dibuat menggunakan metode 1 dengan yang dibuat menggunakan metode 2

Kalimat	Jumlah pemilih		persentase jumlah pemilih suatu metode terhadap jumlah seluruh responden	
	metode 1	metode 2	metode 1	metode 2
selamat dikejar	12	4	75%	25%
maling kambing	15	1	93.75%	6.25%
timun emas	11	5	68.75%	31.25%
taring macan	13	3	81.25	18.75%
total	51	13	79.6875	20.313%

Pada Tabel 4.1, bisa dilihat bahwa dari 16 responden sebagian besar responden menyatakan bahwa suara sintesis yang dibuat dengan metode 1 lebih baik dari suara yang dibuat dengan menggunakan metode 2. Untuk ucapan “selamat dikejar”, 75% responden menjawab metode 1 lebih baik, dan 25% responden menjawab metode 2 lebih baik. Untuk ucapan “maling kambing”, 93.75% responden menjawab metode 1 lebih baik dan 6.25% responden menjawab metode 2 lebih baik. Untuk ucapan “timun emas” 68.75% responden menjawab metode 1 lebih baik dan 31.25% responden menjawab metode 2 lebih baik. Untuk ucapan “taring macan”, 81.25% responden menjawab metode 1 lebih baik dan 18.75% responden menjawab metode 2 lebih baik.

Jika ditotal secara keseluruhan. Dari 64 kasus terdapat 79.69% kasus dimana pada *speech synthesizer* yang menggunakan *speech engine* UTAU, suara sintesis yang dibuat dengan menggunakan unsur KVK dinyatakan lebih jelas oleh responden dibandingkan suara sintesis yang dibuat tanpa menggunakan unsur KVK.

Tabel 4.2 Perbandingan pilihan responden atas kejelasan suara antara yang dibuat menggunakan metode 3 dengan yang dibuat menggunakan metode 4

Kalimat	Jumlah pemilih		persentase jumlah pemilih suatu metode terhadap jumlah seluruh responden	
	metode 3	metode 4	metode 3	metode 4
selamat dikejar	16	0	100	0%
maling kambing	15	1	93.75	6.25%
timun emas	14	2	87.5	12.5%
taring macan	16	0	100	0%
Total	61	3	95.3125	4.6875%

Tabel 4.2 adalah data penilaian responden dalam menilai kejelasan suara sintesis antara yang dibuat menggunakan metode 2 dengan yang dibuat menggunakan metode 3. Pada Tabel 4.2, bisa dilihat bahwa dari survey terhadap 16 responden sebagian besar responden menyatakan bahwa suara sintesis yang dibuat dengan metode 3 lebih baik dari suara yang dibuat dengan suara sintesis yang dibuat dengan menggunakan metode 4. Untuk ucapan “selamat dikejar”, 100% responden menjawab metode 3 lebih baik. Untuk ucapan “maling kambing”, 93.75% responden menjawab metode 3 lebih baik dan 6.25% responden menjawab metode 4 lebih baik. Untuk ucapan “timun emas” 87.5% responden menjawab metode 3 lebih baik dan 12.5% responden menjawab metode 4 lebih baik. Untuk ucapan “taring macan”, 100% responden menjawab metode 3 lebih baik dari metode 4.

Jika ditotal secara keseluruhan. Dari 64 kasus 95% kasus menyatakan bahwa pada penggunaan *speech synthesizer* ber algoritma SOLA dan *resampling*, suara sintesis yang dibuat dengan menggunakan KVK dinyatakan lebih jelas oleh responden dibandingkan dengan suara sintesis yang dibuat tanpa menggunakan KVK.

Tabel 4.3 Perbandingan pilihan responden atas kejelasan suara antara yang dibuat menggunakan metode 3 dengan yang dibuat menggunakan metode 2

Kalimat	Jumlah pemilih		persentase jumlah pemilih suatu metode terhadap jumlah seluruh responden	
	metode 3	metode 2	metode 3	metode 2
selamat dikejar	3	13	18.75%	81.25%
maling kambing	11	5	68.75%	31.25%
timun emas	15	1	93.75%	6.25%
taring macan	9	7	56.25%	43.75%
Total	38	26	59.375%	40.625%

Pada Tabel 4.3, bisa dilihat bahwa dari 16 responden, untuk ucapan “selamat dikejar”, 18.75% responden menjawab metode 3 lebih baik, dan 81.25% responden menjawab metode 2 lebih baik. Untuk ucapan “maling kambing”, 68.75% responden menjawab metode 3 lebih baik dan 31.25% responden menjawab metode 2 lebih baik. Untuk ucapan “timun emas” 93.75% responden menjawab metode 3 lebih baik dan 6.25% responden menjawab metode 2 lebih baik. Untuk ucapan “taring macan”, 56.25% responden menjawab metode 3 lebih baik dan 43.75% responden menjawab metode 2 lebih baik.

Jika ditotal secara keseluruhan. Dari 64 kasus terdapat 59.375% kasus dimana penggunaan *voicebank* “LULU3” dengan menggunakan KVK walaupun menggunakan *speech synthesizer* sederhana dinyatakan lebih baik daripada penggunaan *voicebank* LULU3 tanpa menggunakan KVK yang disintesiskan dari *speech engine* UTAU. Yang menarik disini adalah dengan penambahan KVK, sebuah suara sintesis yang dihasilkan dari *speech synthesizer* ber algoritma sederhana bisa menyaingi suara sintesis yang dihasilkan oleh sistem *speech synthesizer* yang ber algoritma lebih kompleks.

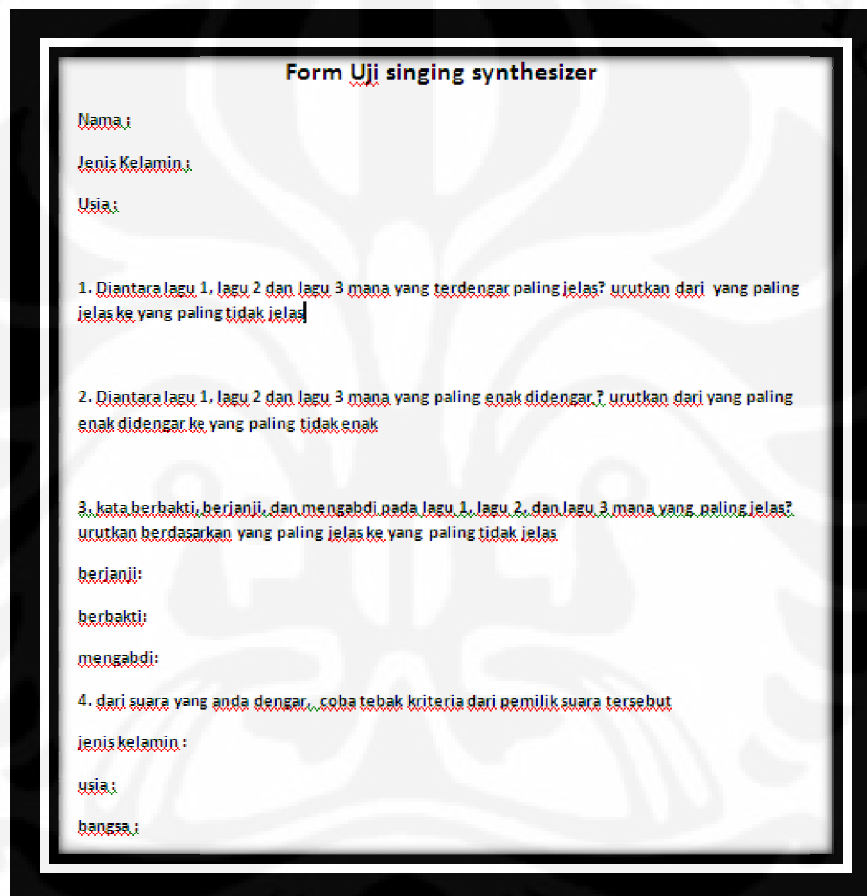
Dari simulasi pada *speech synthesizer* ini bisa disimpulkan bahwa dalam bahasa Indonesia, ucapan sintesis yang dibuat dengan *voicebank* berformat KVK membuat suara lebih jelas didengar.

4.2 Simulasi Penggunaan Fonem Berformat KVK pada *Singing synthesizer*

Pada bagian ini, penulis mencoba menggunakan *voicebank* “LULU3” yang terdiri dari fonem fonem berformat V, VK, KV, dan KVK untuk menyanyikan lagu berbahasa Indonesia dengan *singing synthesizer* UTAU. Suara yang dihasilkan nantinya akan dibandingkan dengan suara nyanyian yang dibuat dengan *voicebank* LULU3 yang tidak menggunakan KVK.

Untuk simulasi *singing synthesizer* ini, penulis akan menggunakan tiga metode untuk membuat sebuah file yang berisi suara nyanyian sintesis, yaitu:

- a. Metode 1: fonem fonem KVK dibuat dengan memanfaatkan fitur overlap dan pre-utterance pada UTAU yaitu dengan KV-VK
- b. Metode 2: fonem fonem KVK dibuat dengan cara menggabungkan format KV – KV
- c. Metode 3: fonem fonem KVK dibuat dengan menggunakan fonem KVK



Form Uji singing synthesizer

Nama: _____

Jenis Kelamin: _____

Usia: _____

1. Diantara lagu 1, lagu 2 dan lagu 3 mana yang terdengar paling jelas? urutkan dari yang paling jelas ke yang paling tidak jelas

2. Diantara lagu 1, lagu 2 dan lagu 3 mana yang paling enak didengar? urutkan dari yang paling enak didengar ke yang paling tidak enak

3. kata berbakti, berjanji, dan mengabdikan pada lagu 1, lagu 2, dan lagu 3 mana yang paling jelas? urutkan berdasarkan yang paling jelas ke yang paling tidak jelas

berjanji: _____

berbakti: _____

mengabdikan: _____

4. dari suara yang anda dengar, coba tebak kriteria dari pemilik suara tersebut

jenis kelamin: _____

usia: _____

bangsa: _____

Gambar 4.2 Form Simulasi *Singing synthesizer*

Lagu yang digunakan adalah lagu berbahasa Indonesia yang berjudul “Padamu Negri” yang diciptakan oleh Kusbini. Pada lagu ini terdapat tiga buah kata yang mengandung fonem KVK, yaitu berjanji, berbakti dan mengabdikan. Pada metode 1, kata kata tersebut akan dibuat dengan perunutan fonem be2 – er2 – ja-an-ji_p, be2-er2-ba-ak-ti_p, dan me2-nga-ab-di_p. Pada metode 2 kata kata tersebut akan dibuat

dengan perunutan fonem be2-ru-ja-nu-ji_p, be2-ru-ba-ku-ti_p, dan me2-nga-bu-di_p. Pada metode 3 kata kata tersebut akan dibuat dengan perunutan fonem ber-jan-ji_p, ber-bak-ti_p, dan me-ngap-di_p

Setelah file suara dibuat, maka selanjutnya adalah meminta pendapat responden untuk menilai ketiga file suara tersebut. Gambar 4.2 adalah form yang harus diisi setiap responden. Form tersebut mempunyai empat pertanyaan yang bisa dilihat juga pada Gambar 4.2 ataupun lampiran. Disini Lagu padamu negeri yang dinyanyikan oleh LULU3 akan diperdengarkan secara bergantian dari versi 1 sampai versi 3 sebanyak dua kali. Para responden kemudian akan menjawab pertanyaan pertanyaan yang ada pada form. Setelah Survey terhadap responden dilakukan. data pendapat responden dibandingkan. Tabel 4.4 adalah data pendapat subjectif responden atas kualitas suara yang dibuat dengan metode tiga dan metode dua.

Tabel 4.4 : Data Pendapat Subjectif Responden Atas Kualitas Nyanyian Yang Dibuat dengan Metode 3 dan Metode 2.

Perbandingan	Jumlah pemilih		Persentase jumlah pemilih suatu metode terhadap jumlah total responden	
	metode 3	metode 2	metode 3	metode 2
Kejelasan	10	5	66.67%	33.33%
enak didengar	10	5	66.67%	33.33%
kejelasan kata berjanji	8	7	53.33%	46.67%
kejelasan kata berbakti	12	3	80%	20%
kejelasan kata mengabdikan	13	2	86.67%	13.33%

Dari tabel 4.4 dapat dilihat bahwa sebagian besar menyatakan bahwa lagu yang dibuat dengan metode 3 lebih jelas dan lebih enak didengar dari metode 2. 66.67% responden menilai bahwa dalam sebuah lagu berbahasa Indonesia fonem KVK yang

dibuat dengan menggunakan KVK lebih jelas dan lebih enak didengar dari pada yang dibuat dengan menggunakan KV+KV

Tabel 4.5 adalah data pendapat subjectif responden atas kualitas suara yang dibuat dengan metode 3 dan metode 1. Dari tabel 4.5 dapat dilihat bahwa sebagian besar memilih lagu yang dibuat dengan metode 3 lebih jelas dan lebih enak didengar dari metode 1. 87.75% responden menilai bahwa dalam sebuah lagu berbahasa Indonesia, fonem KVK yang dibuat dengan menggunakan KVK lebih jelas didengar dari pada yang dibuat dengan menggunakan KV+VK. Fitur Preutterance dan overlap pada UTAU tidak bisa membuat transisi KV dan VK menjadi halus

Tabel 4.5 : Data Pendapat Subjectif Responden Atas Kualitas Suara Yang Dibuat dengan Metode 3 dan Metode 1.

Perbandingan	Jumlah pemilih		Persentase jumlah pemilih suatu metode terhadap jumlah total responden	
	metode 3	metode 1	metode 3	metode 1
Kejelasan	14	1	87.75%	12.25%
enak didengar	12	3	75%	25%
kejelasan kata berjanji	8	7	53.33%	46.67%
kejelasan kata berbakti	13	2	86.67%	13.33%
kejelasan kata mengabdikan	13	2	86.67%	13.33%

Tabel 4.6 adalah Data Pendapat Subjectif Responden atas Kualitas Suara yang Dibuat dengan Metode 2 dan Metode 1. Dari tabel 4.6 dapat dilihat bahwa 73.33% responden menilai bahwa dalam sebuah lagu berbahasa Indonesia, fonem KVK yang dibuat dengan menggunakan KV+KV lebih jelas didengar dari pada yang dibuat dengan menggunakan KV+VK.

Tabel 4.6 : Data Pendapat Subjectif Responden atas Kualitas Suara yang Dibuat dengan Metode 2 dan Metode 1.

Perbandingan	Jumlah pemilih		Persentase jumlah pemilih suatu metode terhadap jumlah total responden	
	metode 2	metode 1	metode 2	metode 1
Kejelasan	11	4	73.33%	26.67%
enak didengar	12	3	80%	20%
kejelasan kata berjanji	9	6	60%	40%
kejelasan kata berbakti	11	4	73.33%	26.67%
kejelasan kata mengabdikan	9	6	60%	40%

Tabel 4.7 : data pendapat subjectif responden atas kualitas suara yang dibuat dengan metode 3 dibandingkan dengan metode 2 atau metode 1.

Perbandingan	Jumlah pemilih		Persentase jumlah pemilih suatu metode terhadap jumlah total responden	
	metode 3	metode 2 atau 1	metode 3	metode 2 atau 1
Kejelasan	10	5	66.67%	33.33%
enak didengar	10	5	66.67%	33.33%
kejelasan kata berjanji	5	10	33.33%	66.67%
kejelasan kata berbakti	10	5	66.67%	33.33%
kejelasan kata mengabdikan	11	4	73.33%	26.67%

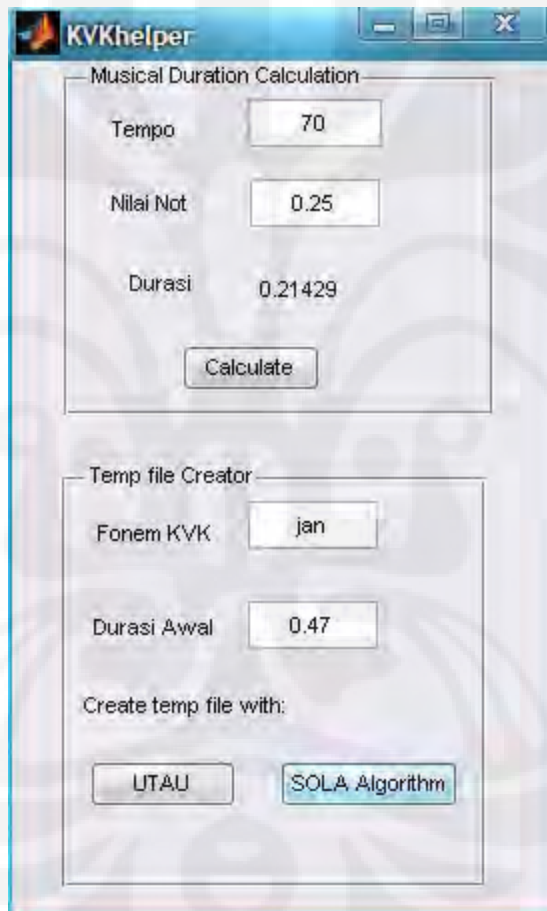
Tabel 4.7 adalah data pendapat subjectif responden atas kualitas suara yang dibuat dengan metode 3 dan metode 2 atau metode 1. Dari tabel 4.7 dapat dilihat 66.67% responden menilai bahwa dalam sebuah lagu berbahasa Indonesia, fonem KVK yang dibuat dengan menggunakan KVK lebih jelas dan lebih enak didengar dari pada yang dibuat dengan menggunakan KV+VK ataupun KV+KV.

Pada Tabel 4.7 terlihat bahwa pada kejelasan kata berjanji, jumlah pemilih metode 3 lebih sedikit dari pada pemilih metode 2 dan metode 1. Setelah diteliti, hal ini dikarenakan karena fonem “jan” hanya mengeluarkan suara “ja” bukan “jan”. Singing Synthesizer UTAU memang tidak dibuat untuk mengolah fonem berformat KVK. Untuk memendekan durasi, UTAU melakukan pemotongan sehingga kata “jan” hanya terdengar “ja”. Agar fonem KVK tetap bisa digunakan di UTAU dengan baik maka. Fonem fonem KVK yang tidak bisa terdengar dengan baik untuk durasi pendek harus di diperpendek terlebih dahulu sebelum diproses oleh UTAU.. Untuk tujuan ini dibuat sebuah program bernama KVKhelper yang ditulis dengan menggunakan matlab. Program ini bertujuan untuk membantu memperpendek durasi file suara pada voicebank “LULU3”. KVKhelper memiliki dua modul yaitu, *modul musical duration calculation* dan *modul temp file creator*. Modul *musical duration calculator* berfungsi menghitung durasi dari sebuah not musik. Persamaan 5.1 adalah persamaan yang digunakan untuk menghitung durasi dari sebuah not musik.

$$= \text{————} \times \quad (5.1)$$

Setelah durasi diketahui, maka selanjutnya adalah memperpendek durasi file suara. Ada tiga pilihan yaitu mengolah file suara secara manual, Menggunakan speech engine UTAU, atau menggunakan algoritma SOLA. Apabila ingin menggunakan speech engine UTAU atau algoritma SOLA, maka bisa menggunakan modul *temp file creator*. Disini ada dua masukan yaitu fonem kvk dan Durasi awal. Masukan fonem KVK adalah nama file suara yang akan diolah. Masukan durasi awal adalah lama durasi dari fonem yang akan diolah. Setelah kedua inputan diisi maka selanjutnya adalah menekan tombol. Ada dua tombol disini yaitu, UTAU dan SOLA

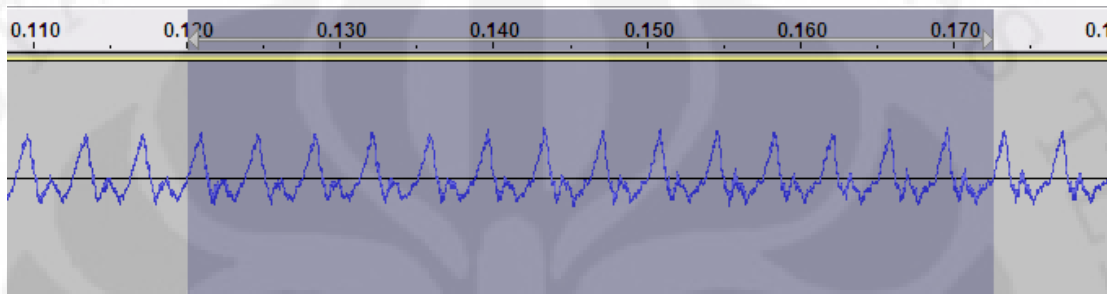
algorithm. Tombol UTAU akan memperpendek dengan speech engine UTAU sedangkan tombol SOLA algorithm akan memperpendek durasi dengan algoritma SOLA. Setelah tombol ditekan, sebuah file sementara dengan format nama “namafonem_s.wav” muncul di *directory* “voice/LULU3/. Setelah file sementara muncul, maka selanjutnya dilakukan pendeteksian *pitch* dengan menggunakan UTAU seperti yang sudah dijelaskan pada bagian 3.1.3.3. Gambar 4.3 adalah tampilan dari GUI dari program KVKhelper



Gambar 4.3 GUI KVKhelper

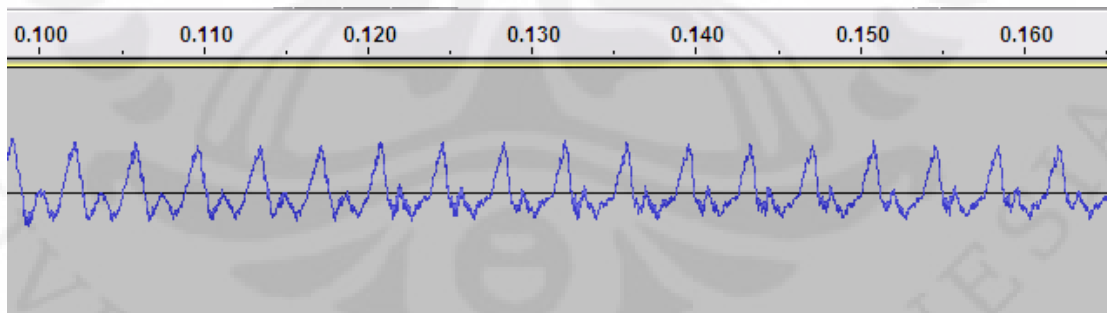
Algoritma yang dipakai untuk melakukan *time scaling* pada program KVKhelper ini sama dengan algoritma *time scaling* yang digunakan pada speech synthesizer 1 dan speech synthesizer 2. Seperti yang terlihat di tabel 3.7 dan 3.9 pada bab 3, hasil dari *time scaling* dengan menggunakan *speech engine* UTAU ataupun

SOLA memiliki akurasi durasi yang buruk sehingga disini dicoba digunakan Alternatif lain yaitu dengan mengedit file suara secara manual. Untuk mengedit file suara secara manual bisa digunakan *software* audacity. Pada *software* Audacity, *file* suara di-*trimming* sampai tepat durasinya. Untuk *voiced sound*, proses *trimming* harus dilakukan dengan hati hati jangan sampai merusak pola yang periodik. Gambar 3.4 adalah ilustrasi bagaimana melakukan *trimming* pada *voiced sound*.



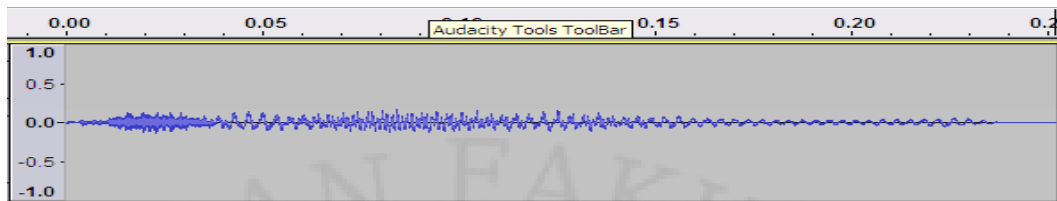
Gambar 4.4 Ilustrasi *Trimming* pada *Voiced Sound*

Pada gambar 3.4 bisa dilihat bahwa *voiced sound* merupakan sinyal periodik. Gambar yang diblok adalah sinyal yang akan dipotong. Disini bisa dilihat bahwa apabila dipotong maka sinyalnya akan tetap terlihat periodik. Gambar 3.5 adalah gambar setelah blok pada gambar 3.4 dihilangkan.

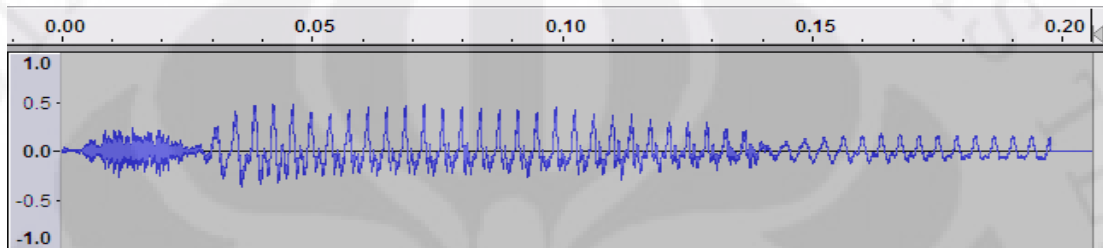


Gambar 4.5 File Suara *Voiced Sound* Setelah Dilakukan *Trimming*

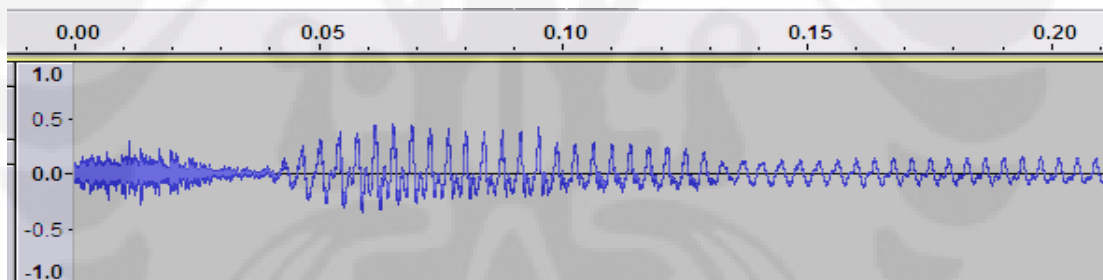
Gambar 4.6 sampai 4.9 adalah perbandingan file suara “jan_s.wav” yang diolah dengan tiga metode dan visualisasi file suara “jan.wav”.



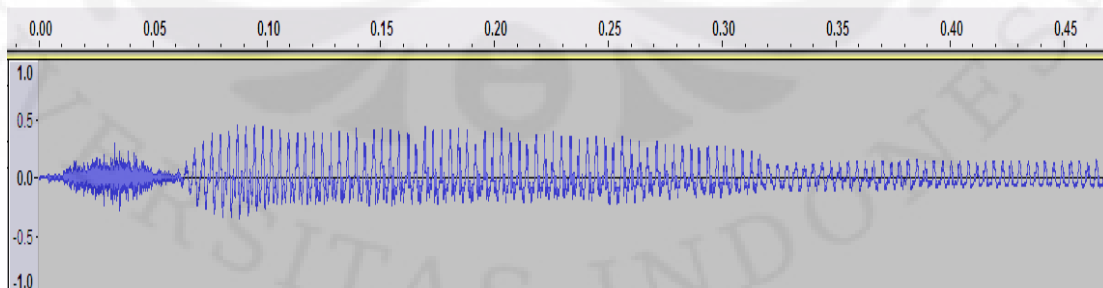
Gambar 4.6 File “jan_s.wav” Diolah Diperpendek Dengan Menggunakan Algoritma SOLA



Gambar 4.7 File “jan_s.wav” Diolah Diperpendek dengan Menggunakan Speech Engine UTAU



Gambar 4.8 File “jan_s.wav” Diolah Diperpendek dengan Cara *Trimming* Manual



Gambar 4.9 File Suara “jan.wav”

BAB 5

KESIMPULAN

- Dari survey terhadap 16 mahasiswa teknik elektro dan teknik komputer universitas Indonesia, sebagian besar menyatakan bahwa dalam speech synthesizer beralgoritma *SOLA-Resampling*, ucapan sintesis yang dibuat dengan tambahan unsur KVK terdengar lebih jelas dari pada ucapan sintesis yang dibuat tanpa menggunakan unsur KVK
- Dari survey terhadap 16 mahasiswa teknik elektro dan teknik komputer universitas Indonesia, sebagian besar menyatakan bahwa dalam speech synthesizer berbasis UTAU, ucapan sintesis yang dibuat dengan tambahan unsur KVK terdengar lebih jelas didengar dari pada ucapan sintesis yang dibuat tanpa menggunakan unsur KVK
- Dari survey terhadap 15 mahasiswa teknik elektro dan teknik komputer universitas Indonesia, sebagian besar menyatakan bahwa dalam singing synthesizer UTAU, nyanyian sintesis yang dibuat dengan tambahan unsur KVK lebih jelas didengar dari pada nyanyian sintesis yang dibuat tanpa menggunakan unsur KVK
- Penambahan unsur KVK pada *voicebank* membantu *speech synthesizer* menghasilkan suara yang lebih jelas dalam bahasa Indonesia.
- Penambahan unsur KVK pada *voicebank* membantu *singing synthesizer* UTAU menghasilkan suara yang lebih jelas dan lebih enak didengar dalam bahasa Indonesia.

DAFTAR REFERENSI

- [1] Thiery Dutoit (1999), “*A Short Introduction of Text to Speech Synthesis*”,
http://tcts.fpms.ac.be/synthesis/introtts_old.html. Juni 3, 2010
- [2] Youngmoo Edmund Kim, (2003) “*Singing voice synthesis/analysis*”. MIT
- [3] Arman, Arry Akhmad. “Konversi dari teks ke ucapan”. Diunduh dari:
<http://indotts.melsa.net.id/knowledge.html>. tanggal 3 Juni 2010
- [4] Stelkic Darko, Blarowski Mike. “*ELEC 484 Project - Time Segment processing*”,
Canada: University of Victoria
- [5] Philippe Boula de Mareüil, Philippe Célérier, Jacques Toen. “*Generation of Emotions by a Morphing Technique in English, French and Spanish*”. Elan TTS & LIMSI-CNRS
- [6] Arman, Arry Akhmad. “Proses Pembentukan dan Karakteristik Sinyal Ucapan”.
Diunduh dari <http://indotts.melsa.net.id/Karakteristik%20Sinyal%20Ucapan.pdf>
tanggal 3 Juni 2010
- [7] Kirk, “*How to create your own UTAU voicebank*”. Diunduh dari:
http://utau.wikia.com/wiki/File:How_To_Create_your_own_UTAU_voice_bank_re_v0.40.pdf tanggal 12 September 2010
- [8] “*Human Voice*”. Juni 3, 2010
http://www.wikipedia.org/wiki/human_voice
- [9] “*Speech Synthesis*”. Juni 3, 2010
http://www.wikipedia.org/wiki/speech_synthesis
- [10] “*Sound*”. Juni 3, 2010
<http://en.wikipedia.org/wiki/sound>
- [11] “*Sound Processing*”. Juni 3, 2010
http://en.wikipedia.org/wiki/sound_processing
- [12] “*Phonetic Transcription*”. Juni 3, 2010
http://en.wikipedia.org/wiki/phonetic_transcription

[13] “*Vocal Range*”. Juni 3, 2010

http://en.wikipedia.org/wiki/vocal_range

[14] Grondin, Francois. “*Guitar Pitch Shifter*”. Juni 3, 2010

<http://www.guitarpitchshifter.com>

[15] “*Useful Matlab function and Script for audio signal and system*”. Juni 3, 2010

<http://www.engr.uky.edu/~donohue/ee513/mfiles/MatlabEE513.htm>

[16] University of Winconsin. “*Digital Audio Properties*”. Juni 3, 2010

<https://academictech.doit.wisc.edu/audacity/audio-video-course/audio-editing-audacity/digital-audio-properties>

[17] Boomkamp, Joost. “*Theory of Digital Sound*”. Juni 3, 2010

<http://www.hammersound.net/audiobasics/audiobasics.html>

DAFTAR PUSTAKA

Pusat Bahasa Departement Pendidikan Nasional (2008). Kamus Besar Bahasa Indonesia. Jakarta: Tim penyusun

Davide Rochesso (2003), Introduction to Sound Processing, Universita di Verona

Alan W. Black, Lenzo, Kevin (2007), Building Synthetic Voices, Carnegie Mellon University. dilihat dari situs <http://festvox.org/bsv/> pada tanggal 11 Desember 2010.

Hasan, akhlan dan Sudaryat, Yayat (1996). "Fonologi Bahasa Indonesia". Departemen Pendidikan dan kebudayaan

LAMPIRAN A: SOURCE CODE SPEECH SYNTHESIZER

Source Code speech synthesizer

algoritma SOLA dan resampling [4]

```
inputpath = get(handles.inputedit,'String');
inputpath1 = ['voice/lulu3/', inputpath, '.wav']
%outputpath = get(handles.outputedit,'String');
outputpath = 'ps1.wav'
inputpitch1 = str2double(get(handles.pitchedit,'String'))

inputdurasi1 = str2double(get(handles.durasiedit,'String'))
[durasiawal, pitchawal] = voicebank(inputpath)

[y,Fs] = wavread(inputpath1);

Sa = 1000;
N = 2000;
M = ceil(length(y)/Sa);
beta = inputpitch1/pitchawal
alpha = beta*inputdurasi1/durasiawal
L = 5;
Ss = round(Sa*alpha);
if Ss >= N disp('alpha is not correct, Ss is >= N')
elseif Ss > N-L disp('alpha is not correct, Ss is > N-L')
end

y(M*Sa+N)=0;
Overlap = y(1:N);

% **** Main TimeScaleSOLA loop ****
for ni=1:M-1
    grain=y(ni*Sa+1:N+ni*Sa);
    XCORRsegment=xcorr(grain(1:L),Overlap(1,ni*Ss:ni*Ss+(L-1)));
    [xmax(1,ni),index(1,ni)]=max(XCORRsegment);
    fadeout=1:(-1/(length(Overlap)-(ni*Ss-(L-1)+index(1,ni)-1))):0;
    fadein=0:(1/(length(Overlap)-(ni*Ss-(L-1)+index(1,ni)-1))):1;
    Tail=Overlap(1,(ni*Ss-(L-1))+ ...
        index(1,ni)-1:length(Overlap)).*fadeout;
    Begin=grain(1:length(fadein)).*fadein;
    Add=Tail+Begin;
    Overlap=[Overlap(1,1:ni*Ss-L+index(1,ni)-1) ...
        Add grain(length(fadein)+1:N)];
end;
```

```
% **** end TimeScaleSOLA loop ****
% Output in WAV file
```

```
wavwrite(Overlap,beta*Fs,outputpath);
disp('proses timesretch telah selesai');
clear all;
```

algoritma memanggil speech engine UTAU

```
inputpath = ['voice/lulu3/',get(handles.inputedit,'String'),'.wav'];
outputpath = 'ps1.wav'
inputpitch1 = str2double(get(handles.pitchedit,'String'));
inputdurasi1= str2double(get(handles.durasiedit,'String'));
[durasiawal, pitchawal] = voicebank(inputpath)
beta = 100*inputpitch1/pitchawal
beta1 = num2str(beta)
teta=100*durasiawal/inputdurasi1
teta1=num2str(teta)
```

```
fname = ['',inputpath, '', outputpath, '',beta1, '',teta1]
system(['resampler.exe' fname]); %run executable with content of fname as inputs
```

```
disp('proses pitchshift telah selesai');
```

algoritma phonemes combiner

```
switch boxchoice
```

```
case 1
```

```
[y,Fs]=wavread('ps1.wav');
wavwrite(y,Fs,outputpath)
```

```
case 2
```

```
[y1,fs]=wavread('ps1.wav');
[y2,fs]=wavread('ps2.wav');
```

```
G=cat(1,y1,y2);
wavwrite(G,44100,outputpath)
```

```
case 3
```

```
[y1,fs]=wavread('ps1.wav');
[y2,fs]=wavread('ps2.wav');
```

```
[y3,fs]=wavread('ps3.wav');
```

Algoritma words Combiner

```
wordboxchoice= str2double(get(handles.wordchoice,'String'))
outputpathw = [get(handles.wordoutput,'String'),'wav'];
pathkata1= [get(handles.kata1,'String'),'wav'];
pathkata2= [get(handles.kata2,'String'),'wav'];
pathkata3= [get(handles.kata3,'String'),'wav'];
pathkata4= [get(handles.kata4,'String'),'wav'];
pathkata5= [get(handles.kata5,'String'),'wav'];
pathkata6= [get(handles.kata6,'String'),'wav'];
pathkata7= [get(handles.kata7,'String'),'wav'];
pathkata8= [get(handles.kata8,'String'),'wav'];
pathkata9= [get(handles.kata9,'String'),'wav'];
pathkata10= [get(handles.kata10,'String'),'wav'];
pathkata11= [get(handles.kata11,'String'),'wav'];
pathkata12= [get(handles.kata12,'String'),'wav'];
```

```
switch wordboxchoice
```

```
case 1
```

```
[y1,Fs]=wavread(pathkata1);
wavwrite(y1,Fs,outputpathw)
```

```
case 2
```

```
[y1,fs]=wavread(pathkata1);
[y2,fs]=wavread(pathkata2);
```

```
G=cat(1,y1,y2);
wavwrite(G,44100,outputpathw)
```

```
case 3
```

```
[y1,fs]=wavread(pathkata1);
[y2,fs]=wavread(pathkata2);
[y3,fs]=wavread(pathkata3);
```


LAMPIRAN B: KOMPOSISI VOICEBANK “LULU3”

Lampiran B adalah daftar *prerecorded speech* yang tersedia dalam *voicebank* “LULU3”. Pada *voicebank* “LULU3” terdapat 1765 fonem yang terdiri dari unsur V, KV, VK, dan KVK.

Fonem	Fonem	Fonem	Fonem	Fonem	Fonem	Fonem	Fonem	Fonem	Fonem
a	a_p	ab	ad	ac	af	ag	ah	aj	ak
al	am	an	ang	any	ap	ar	as	at	aw
ax	ay	az	ba	ba_p	bah	bak	bal	bam	ban
bang	bar	bap	bas	baw	bay	ba	be	be2	be_p
be2_p	beh2	beh	bek2	bek	bel2	bel	bem2	bem	ben2
ben	beng2	beng	bep2	bep	ber2	ber	bes2	bes	bet2
bet	bew2	bew	bey2	bey	bi	bi_p	bih	bik	bil
bim	bin	bing	bip	bir	bis	bit	biw	biy	blas
blat	blet2	blet	bles2	bles	blis	blit	blos	blot	blus
blut	bo	bo_p	bok	bol	bom	bon	bong	bop	bor
bos	bot	bow	boy	bra	bre2	bre	bri	bro	bru
bu	bu_p	buk	bul	bum	bun	bung	bup	bur	bus
but	buw	buy	bya	bye2	bye	byi	byo	byu	ca
ca_p	cah	cak	cal	cam	can	cang	cap	car	cas
cat	caw	cay	ce2	ce2_p	ce	ce_p	ceh2	ceh	cek2
cel	cel2	cem2	cem	cen2	cen	ceng2	ceng	cep2	cep
cer2	cer	ces2	ces	cet2	cet	cew2	cew	cey2	cey
cha	che2	che	chi	cho	chu	ci	ci_p	cih	cik
cil	cim	cin	cing	cip	cir	cis	cit	ciw	cla
cle2	cle	cli	clo	clu	co	co_p	coh	cok	col
com	con	cong	cop	cor	cos	cot	cow	coy	cra
cre2	cre	cri	cro	cru	cu	cu_p	cuh	cuk	cul
cum	cun	cung	cup	cur	cus	cut	cuw	cuy	da
da_p	dah	dak	dal	dam	dan	dap	dar	das	day
de2	de2_p	de	de_p	deh2	deh	dek2	dek	del2	del
dem2	dem	den2	den	dep2	dep	der2	der	des2	des
dey2	dey	di	di_p	dih	dik	dil	dim	din	dip
dir	dis	diy	do	do_p	doh	dok	dol	dom	don
dop	dor	dos	doy	dra	dre2	dri	dro	dru	du
du_p	duh	duk	dul	dum	dun	dup	dur	dus	duy
dwa	dwe2	dwe	dwi	dwo	dwu	dya	dye2	dye	dyi
dyo	dyu	e2	e2_p	e_p	e	eb2	eb	ec2	ec
ecd2	ed	ef2	ef	eg2	eg	eh2	eh	ej2	ej
ek2	el	em	em2	el2	en	en2	eng2	eng	eny2
eny	ep2	ep	er2	er	es2	es	et2	et	ew2

ew	ex2	ex	ey2	ey	ez2	ez	fa	fa_p	fah
fak	fal	fam	fan	fang	fap	far	fas	fat	faw
fay	fe	fe_2	fe_p	feh2	feh	fek	fel2	fem2	fem
fen2	fen	feng2	feng	fep	fep2	fer2	fer	fes2	fes
fet2	fet	few2	few	fey2	fey	fi	fi_p	fih	fik
fil	fim	fin	fing	fip	fir	fis	fit	fiw	fo
fo_p	foh	fok	fol	fom	fon	fong	fop	for	fos
fot	fow	foy	fra	fre2	fre	fri	fro	fru	fu
fu_p	fuh	fuk	ful	fum	fun	fung	fup	fur	fus
fut	fuw	fuy	fya	fye2	fye	fyi	fyo	fyu	ga
ga_p	gah	gak	gal	gam	gan	gang	gap	gar	gas
gat	gaw	gay	ge2	ge2_p	geh	geh2	gek	gek2	gel2
gel	gem	gem2	gen2	gen	geng2	geng	gep2	gep	ger2
ger	ges2	ges	get	get2	get	gew2	gew	gey2	gey
gha	ghe2	ghe	ghi	gho	ghu	gi	gi_p	gih	gik
gil	gim	gin	ging	gip	gir	gis	git	giw	giy
gla	gle2	gle	gli	glo	glu	go	go_p	goh	gok
gol	gom	gon	gong	gop	gor	gos	got	gow	goy
gra	gre2	gre	gri	gro	gru	gu	gu_p	guh	guk
gul	gum	gun	gung	gup	gur	gus	gut	guw	guy
gya	gye2	gye	gyi	gyo	gyu	ha	ha_p	hah	hak
hal	ham	han	hang	hap	har	has	hat	haw	hay
he2	he2_p	he	he_p	heh	heh2	hek2	hek	hel2	hel
hem2	hem	hen2	hen	heng2	heng	hep	hep2	her2	her
hes2	hes	het2	het	hew2	hew	hey2	hey	hi	hi_p
hih	hik	hil	him	hin	hing	hip	hir	his	hit
hiw	ho	ho_p	hoh	hok	hol	hom	hon	hong	hop
hor	hos	hot	how	hoy	hu	hu_p	huh	huk	hul
hum	hun	hung	hup	hur	hus	hut	huw	huy	i
i_p	ib	ic	id	if	ig	ih	ik	il	im
in	ing	ip	ir	is	it	iw	ix	iy	iz
ja	ja_p	jar	jas	jat	jaw	jay	je2	je2_p	je
je_p	jeh2	jeh	jek2	jek	jel	jel2	jem	jem2	jen
jen2	jeng	jeng2	jep	jep2	jer2	jer	jes	jes2	jet
jet2	jew	jew2	jey2	jey	ji	ji_p	jih	jik	jil
jim	jin	jing	jip	jir	jis	jit	jiw	jiy	jo
jo_p	joh	jok	jol	jom	jon	jong	jop	jor	jos
jot	jow	joy	ju	ju_p	juh	juk	jul	jum	jun
jung	jup	jur	jus	jut	juw	juy	jwa	jwe2	jwe
jwi	jwo	jwu	ka	ka_p	kah	kak	kal	kam	kan
kang	kap	kar	kas	kat	kaw	kay	ke2	ke2_p	keh2
keh	kek2	kek	kel2	kel	kem2	kem	ken2	ken	keng2

keng	kep2	kep	ker2	ker	kes2	kes	ket2	ket	kew
key2	key	kha	khe2	khe	khi	kho	khu	ki	ki_p
kih	kik	kil	kim	kin	king	kip	kir	kis	kit
kiw	kiy	kle	kli	klo	klu	ko	ko_p	koh	kok
kol	kom	kon	kong	kop	kor	kos	kot	kow	koy
kra	kre2	kre	kri	kro	kru	ku	ku_p	kuh	kuk
kul	kum	kun	kung	kup	kur	kus	kut	kuw	kuy
kwa	kwe2	kwe	kwi	kwo	kwu	la	la_p	lah	lak
lam	lan	lang	lap	lar	lur	las	lat	law	lay
le2_p	le	le_p	leh	lek2	lek	lem2	lem	len2	len
leng2	leng	lep2	ler2	lep	ler	les2	les	let2	let
lew2	lew	ley2	ley	li	li_p	lih	lik	lim	lin
ling	lip	lir	lis	lit	liw	liy	lo	lo_p	loh
lok	lom	lon	long	lop	lor	los	lot	low	loy
lu	lu_p	luh	luk	lum	lun	lung	lup	lus	lut
luw	luy	ma	ma_p	mah	mak	mal	mam	man	mang
map	mar	mas	mat	may	me2	me2_p	me	me_p	meh2
meh	mek2	mek	mel2	mel	mem2	mem	men2	men	meng2
meng	mep2	mep	mer2	mer	mes2	mes	met2	mey2	mey
mi	mi_p	mih	mik	mil	mim	min	ming	mip	mir
mis	mit	miy	mo	mo_p	moh	mok	mol	mom	mon
mong	mop	mor	mos	mot	moy	mu	mu_p	muh	muk
mul	mum	mun	mung	mup	mur	mus	mut	muy	na
na_p	nah	nak	nal	nam	nan	nang	nap	nar	nas
nat	naw	nay	ne2	ne2_p	ne	ne_p	neh2	neh	nek2
nek	nel2	nel	nem2	nem	nen2	nen	neng2	neng	nep2
nep	ner2	ner	nes2	nes	net2	net	new	ney2	ney
nga	ngan	ngap	nge2	nge	ngen2	ngen	ngep2	ngep	ngi
ngin	ngip	ngo	ngon	ngop	ngu	ngun	ngup	ni	ni_p
nih	nik	nil	nim	nin	ning	nip	nir	nis	nit
niw	no	no_p	noh	nok	nol	nom	non	nong	nop
nor	nos	not	now	noy	nu2	nu	nu_p	nuh	nuk
nul2	num	nun	nung	nup	nur	nus	nut	nuw	nuy
nya	nye2	nye	nyi	nyo	nyu	o	o_p	ob	oc
od	of	og	oh	oj	ok	ol	pap	om	on
ong	ony	op	or	os	ot	ow	ox	oy	oz
pa	pa_p	pah	pak	pal	pam	pan	pang	par	pas
pat	paw	pay	pe2	pe2_p	pe	pe_p	peh2	peh	pek2
pek	pel2	pel	pem2	pem	pen2	pen	peng2	peng	pep2
pep	per2	per	pes2	pes	pet	pew	pey2	pey	pi
pi_p	pih	pik	pil	pim	pin	ping	pip	pir	pis
pit	piw	yiy	pla	ple2	ple	pli	plo	plu	po

po_p	poh	pok	pol	pom	pon	pong	pop	por	pos
pot	pow	poy	pra	pre2	pre	pri	pro	pru	pu
pu_p	puh	puk	pul	pum	pun	pung	pup	pur	pus
put	puw	puy	ra	ra_p	ran	rak	ral	ram	ran
rang	rap	ras	rat	raw	ray	re	re2_p	re	re_p
reh2	reh	rek2	rek	rel	rel	rem2	rem	ren2	ren
reng2	reng	rep2	rep	res2	res	ret2	ret	rew2	rew
rey	rey	ri	ri_p	rih	rik	ril	rim	rin	rik
ril	rim	rin	ring	rip	ris	rit	riw	riy	ro
ro_p	roh	rok	rol	rom	ron	rong	rop	ros	rot
row	roy	ru	ru_p	ruh	ruk	rul	rum	run	rung
rup	rus	rut	ruw	ruy	sa	sa_p	sah	sak	sal
san	sang	sar	sas	sat	saw	say	se2	se2_p	se
se_p	seh2	she	sek	sek2	sel	sel2	sem2	sem	sen2
sen	ser2	ser	ses2	ses	set2	set	sew	sew2	sey2
sey	si	si_p	sih	sik	sil	silence	sim	sin	sing
sir	sis	sit	siw	siy	so	so_p	soh	sok	sol
som	son	song	sor	sos	sot	sow	soy	su	su_p
suh	suk	sul	sum	sun	sung	sur	sus	sut	suw
suy	sya	sya_p	ta	ta_p	tah	tak	tal	tam	tan
tang	tap	tar	tas	tat	taw	tay	te2	te2_p	te
te_p	teh2	the	tek2	tek	tel2	tel	tem2	tem	ten2
ten	teng2	teng	tep2	tep	ter2	ter	tes2	tes	tet2
tet	tew2	tey2	tey	ti	ti_p	tih	tik	til	tim
tin	ting	tip	tir	tis	tit	tiw	tiy	to	to_p
toh	tok	tol	tom	ton	tong	top	tor	tos	tot
tow	toy	tra	trang	tre2	tre	treng	treng	tri	tring
tro	trong	tru	trung	tu	tu_ptuh	tuk	tul	tum	tun
tung	tup	tur	tus	tut	tuw	tuy	u	u_p	u_p
ub	uc	ud	uf	ug	uh	uj	uk	ul	um
un	ung	uny	up	ur	us	ut	uw	ux	uy
uz	wa	wa_p	wah	wak	wal	wam	wan	wang	wap
war	was	wat	waw	way	we2	we2_p	we	we_p	weh2
weh	wek	wel	wel2	wem2	wem	weng	weng2	wep2	wep
wer2	wer	wes2	wes	wet2	wet	wew2	wew	wey2	wey
wi	wi_p	wih	wik	wil	wim	win	wing	wip	wir
wis	wit	wiw	wiy	wo	wo_p	who	wok	wol	wom
won	wong	wop	wor	wos	wot	wow	woy	wu	wu_p
wuh	wuk	wul	wum	wun	wung	wup	wur	wus	wut
wuw	wuy	ya	ya_p	yah	yak	yal	yam	yan	yang
yap	yar	yas	yat	yaw	yay	ye2_p	yeh2	yeh	yek
yel2	yel	yem2	yem	yen2	yen	yeng	yeng2	yep2	yep

yer2	yer	yes2	yes	yet2	yet	yew2	yew	yey2	yey
yi	yi_p	yih	yik	yil	yim	yin	ying	yip	yir
yis	yit	yiw	yy	yo	yo_p	yoh	yok	yol	yom
yon	yong	yop	yor	yos	yot	yow	yoy	yu	yu_p
yuh	yuk	yul	yum	yun	yung	yup	yur	yus	yut
yuw	yuy	za	za_p	zah	zak	zal	zam	zan	zang
zap	zar	zas	zat	zaw	zay	ze2_p	ze	ze_p	zeh2
zeh	zek2	zek	zel2	zel	zem2	zem	zen	zeng	zeng2
zep2	zep	zer2	zer	zes	zes2	zet2	zet	zew2	zew
zey2	zey	zi	zi_p	zik	zih	zil	zim	zin	zing
zip	zir	zis	zit	zit	ziw	ziy	zo	zo_p	zoh
zok	zol	zom	zon	zong	zop	zor	zos	zot	zow
zoy	zu	zu_p	zuh	zuk	zul	zum	zun	zung	zup
zur	zus	zut	zuw	zuy					

LAMPIRAN C: TABEL DETEKSI FONEM BAHASA INDONESIA

Pada lampiran ini ditulis daftar fonem beserta kata yang mengandung fonem tersebut. Lampiran ini bertujuan untuk menghilangkan fonem fonem yang sebetulnya tidak pernah digunakan dalam bahasa Indonesia. Akan tetapi, karena berbagai keterbatasan, deteksi fonem pada skripsi ini masih belum lengkap. Untuk Fonem berformat V, dan KV sudah cukup lengkap. Namun untuk fonem berformat VK, dan KVK masih jauh dari lengkap.

Tabel Daftar Fonem V Bahasa Indonesia Hasil Pendeteksian

Fonem	Kata	Fonem	kata
a	apel	o	obat
i	ibu	e2	elang
u	udang	au (aw)	auto
e	enak		

Tabel Daftar Fonem VK Bahasa Indonesia Hasil Pendeteksian

Fonem	Kata	Fonem	Kata
ab	abjad	am	ambil
ad	admiral	an	andai
af	afdol	ang	angkuh
ah	ahli	ar	arti
ak	aksara	as	asma
al	alga	at	atmosfer

Tabel Daftar Fonem KV Bahasa Indonesia Hasil Pendeteksian

Fonem	Kata	Fonem	Kata
ba	babak	ku	kutu
be	bebek	ke	keong
bi	bilang	ko	kotor
bu	burung	ke2	keran
bo	borong	la	lagu
be2	besar	li	limbah
ca	cari	lu	luput
ya	yakin	le	lempar
cu	curi	lo	loncat
co	coba	le2	lepas
ce2	cepat	ma	makan

Fonem	Kata	Fonem	Kata
da	dadu	mi	minum
di	diam	mu	music
du	duren	me	merah
de	desa	mo	mobil
do	dosa	me2	mega
de2	depan	na	nanas
fa	fakir	ni	nilai
fi	fisika	nu	nusa
we	wesel	ne	nenek
ze	zebra	no	nomor
fo	forum	ne2	negeri
ga	gajah	pa	patuh
gi	gigi	pi	pisang
gu	gula	pu	pusat
ge	getah	pe	pesek
go	gotong	po	pohon
ge2	gelang	pe2	perang
ha	hari	ra	rasa
hi	hidup	ri	rindu
hu	hujan	ru	rumah
he	hemat	re	rela
ho	hotel	ro	rotan
he2	hening	re2	rendah
ja	jari	sa	sakit
ji	jinak	si	silau
ju	jual	su	suka
zo	zodiac	se	sehat
jo	jodoh	so	soal
je2	jenis	se2	sepuh
ka	kali	ta	tarik
ki	kidal	ti	tikus
tu	tutup	yu	yuris
te	tekad	ye	yel
to	tolong	yo	yoga
te2	teman	za	zakat
wa	wanita	zi	zikir
we2	wedang	zu	zuhur
wu	wujud		

Tabel Daftar Fonem KVK Bahasa Indonesia Hasil Pendeteksian

Fonem	Kata	Fonem	Kata
bah	bahtera	del	delman

Fonem	Kata	Fonem	Kata
bad	abad	dem	idem
bab	bab	den	dendeng
bak	tebak	deng	dendeng
bal	sambal	duh	seduh
bam	lebam	duk	duduk
ban	bantal	dul	sundul
bang	bangku	dung	bendung
bar	lebar	dup	hidup
bas	tebas	dur	tidur
bat	hebat	dus	tandus
baw	kerbau	dut	badut
bay	lambai	dok	sendok
bih	tasbih	dol	cendol
bib	tabib	don	tendon
bik	cabik	dong	brondong
bil	sambil	dor	mandor
bim	bimbingan	dot	sedot
bin	bintara	dek2	ledek
bing	kambing	del2	bandel
bir	gelambir	dem2	tandem2
bis	habis	deh	lodeh
bit	bit	fus	tifus
bek	sobek	gang	gan
bel	gembel	gap	tegap
ben	benteng	gar	tegar
bes	asbes	gas	gas
buh	bubuh	gis	bugis
buk	bubuk	git	gigit
bul	gembul	jang	kijang
bun	tambun	jing	anjing
bur	gembur	jir	banjir
bus	gabus	jon	jontor
bung	tabung	ding	pudding
but	lembut	dip	kedip
dil	bedil	dir	pander
dim	dimsum	dis	kudis
bek2	ngambek	cat	cat
bel2	sambel	caw	kacau
bes2	asbes	cek	cek
bet2	ribet	cen	kuncen
boh	heboh	cet	pacet
bol	bobol	coh	kecoh

Fonem	Kata	Fonem	Kata
bom	bom	cong	bencong
bos	bos	cor	bocor
bot	robot	cot	bekicot
cah	pecah	cuh	acuh
ceh	rekeh	cung	kacung
cak	becak	cur	mancur
cam	kecam	cek2	cek2
can	cantik	cel2	pecel
cang	kacang	cem2	bacem
cap	kecap	cen2	cendawan
car	pacar	cer2	cermat
dak	bedak	dah	sudah
dal	sandal	dih	didih
dam	gendam	duh	seduh
dan	medan	duk	duduk
dang	tendang	dul	gundul
dap	sedap	dung	bendung
dar	pudar	dup	hidup
das	pedas	dur	tidur
dat	sendat	dus	kardus
dik	didik	dut	badut

LAMPIRAN D: FORM SURVEY

Form Uji singing synthesizer

Nama :

Jenis Kelamin :

Usia :

1. Diantara lagu 1, lagu 2 dan lagu 3 mana yang terdengar paling jelas? urutkan dari yang paling jelas ke yang paling tidak jelas

2. Diantara lagu 1, lagu 2 dan lagu 3 mana yang paling enak didengar ? urutkan dari yang paling enak didengar ke yang paling tidak enak

3. kata berbakti, berjanji, dan mengabdikan pada lagu 1, lagu 2, dan lagu 3 mana yang paling jelas? urutkan berdasarkan yang paling jelas ke yang paling tidak jelas

berjanji:

berbakti:

mengabdikan:

4. dari suara yang anda dengar, coba tebak kriteria dari pemilik suara tersebut

jenis kelamin :

usia :

bangsa :

Tanda Tangan

()

form uji speech synthesizer

Nama:

Usia :

Jenis Kelamin:

Tentukan urutan kalimat dari yang paling jelas terdengar sampai yang paling tidak jelas

kalimat	1	2	3	4
selamat dikejar				
maling kambing				
timun emas				
taring macan				