RANCANG-BANGUN *RESTU'S AN ENGINE FOR SYNTHETIC THESPIAN UNITS* (RESTU): SUBSISTEM PENENTUAN POSISI PENGGUNA BERDASARKAN SINYAL SUARA (SPEACAL)

TESIS

Karya tulis sebagai salah satu syarat
untuk memperoleh gelar Magister dari
Institut Teknologi Bandung

Oleh

ADITYA ARIE NUGRAHA

NIM: 23209346

Program Magister Teknik Elektro

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

2011

RANCANG-BANGUN *RESTU'S AN ENGINE FOR SYNTHETIC THESPIAN UNITS* (RESTU): SUBSISTEM PENENTUAN POSISI PENGGUNA BERDASARKAN SINYAL SUARA (SPEACAL)

Oleh
Aditya Arie Nugraha

Bandung, ...

Menyetujui

Pembimbing I,	Pembimbing II,
Dr. Ary Setijadi Prihatmanto	**Ir. Tunggal Mardiono, M.Sc.**

ABSTRAK

RANCANG-BANGUN *RESTU'S AN ENGINE FOR SYNTHETIC THESPIAN UNITS* (RESTU): SUBSISTEM PENENTUAN POSISI PENGGUNA BERDASARKAN SINYAL SUARA (SPEACAL)

Oleh

Aditya Arie Nugraha

NIM: 23209346

PROGRAM MAGISTER TEKNIK ELEKTRO

Sebagai sebuah *engine* untuk membangun sistem *embodied conversational agent* (ECA), *RESTU's an Engine for Synthetic Thespian Units* (RESTU) membutuhkan subsistem yang dapat memberikan informasi posisi pengguna. Informasi tersebut dapat dimanfaatkan untuk meningkatkan interaksi agen virtual dengan pengguna, misalnya dengan terjadinya kontak mata antara agen virtual dan pengguna. Posisi pengguna dapat diperkirakan berdasarkan sinyal suara yang tertangkap oleh mikrofon atau citra yang tertangkap oleh kamera.

Penelitian ini bertujuan merancang, mengimplementasikan, dan menguji subsistem penentuan posisi pengguna berdasarkan sinyal suara, serta mengintegrasikannya dengan RESTU. Untuk memperkirakan posisi pengguna, subsistem yang kemudian diberi nama SpeaCal ini mencoba memanfaatkan parameter *time difference of arrival* (TDOA) dan *peak-to-peak amplitude ratio* (PtPAR) yang digunakan sebagai masukan jaringan syaraf tiruan (JST).

Perangkat keras yang digunakan untuk mengimplementasikan SpeaCal adalah komputer atau laptop, *USB sound card* dengan satu kanal masukan dan *sample rate* 24 KHz, serta mikrofon. SpeaCal menggunakan sebuah *microphone array* yang tersusun dari empat buah mikrofon.

Dari proses pengambilan data untuk JST, diketahui bahwa TDOA yang dihasilkan oleh SpeaCal tidak dapat digunakan sebagai masukan JST karena tingkat konsistensinya sangat rendah dan cenderung tidak valid. Hal ini disebabkan oleh syarat *time-constraint* yang penting bagi perhitungan TDOA tidak dapat dipenuhi. Oleh karena itu, penentuan posisi pengguna pada SpeaCal hanya menggunakan parameter PtPAR.

Dua set data yang menyusun 240 data latih dan 60 data uji digunakan untuk melatih JST. JST yang dihasilkan kemudian diuji dengan tiga set data lain yang secara total memuat 200 data. JST terbaik yang diperoleh memiliki 3 lapisan dengan 16 neuron tersembunyi. *Mean squared error* (MSE) pelatihan dan pengujian JST tersebut mencapai 0,0001499649 dan 0,005309. Pengujian dengan tiga set data lain menghasilkan MSE 0,139543; 0,210295; dan 0,464500.

JST tersebut kemudian digunakan dalam pengujian SpeaCal yang telah diintegrasikan dengan RESTU. Dalam proses pengujian, diketahui bahwa SpeaCal mampu menghasilkan informasi perkiraan posisi pengguna (sumber suara) untuk RESTU secara *real-time*, meskipun hasilnya seringkali masih tidak akurat.

Kata kunci: penentuan posisi pewicara, *time difference of arrival*, *peak-to-peak amplitude ratio*, jaringan syaraf tiruan, *microphone array*

ABSTRACT

THE DESIGN AND IMPLEMENTATION OF *RESTU'S AN ENGINE FOR SYNTHETIC THESPIAN UNITS* (RESTU): SPEAKER LOCALIZATION SUBSYSTEM (SPEACAL)

By

Aditya Arie Nugraha

NIM: 23209346

ELECTRICAL ENGINEERING MASTER PROGRAM

As an engine for building an embodied conversational agent (ECA) system, RESTU's an Engine for Synthetic Thespian Units (RESTU) required at least one subsystem that can provides user (speaker) position information. Such information can be used to enhance the interaction between virtual agent and user, for example to make eye contact between them. The user position can be estimated using the sound signals captured by the microphones or the images captured by the cameras.

This research aimed to design, implement, and test the speaker localization subsystem (named SpeaCal), then integrate the subsystem with RESTU. To estimate the user position, SpeaCal tried to utilize Time Difference Of Arrival (TDOA) and Peak-to-Peak Amplitude Ratio (PtPAR) parameters as the Artificial Neural Network (ANN) input.

The hardwares used to implement SpeaCal were a computer or laptop, USB sound cards which had single input channel and only support 24 KHz sample rate, and microphones. SpeaCal used a microphone array that consisted of four microphones.

From the ANN data acquisition process, it is known that the TDOA value generated by SpeaCal could not be used as ANN input because its consistency is very poor and likely to be invalid. This was caused by the time-constraint which was essential for TDOA calculation could not be met. Therefore, SpeaCal only used PtPAR parameter to estimate user position.

A training data that consisted of 240 data and a testing data that consisted of 60 data are used in ANN training process. Both training and testing data were constructed using

two data sets. Then, the resulting ANN is tested using three other data sets which consisted of 200 data in total. The best ANN obtained from training processes had 3 layers with 16 hidden neurons. The mean squared error (MSE) of its training process reached 0.000149964, while the testing reached 0.005309. The MSE of three other testing processes reached 0.139543, 0.210295, and 0.464500.

Then, the ANN was used in the testing of SpeaCal which had been integrated with RESTU. The testing showed that SpeaCal could provide user position estimation for RESTU in real-time, although the estimated user positions were often still inaccurate.

Keywords: speaker localization, time difference of arrival, peak-to-peak amplitude ratio, artificial neural network, microphone array

PEDOMAN PENGGUNAAN TESIS

Tesis S2 yang tidak dipublikasikan, terdaftar dan tersedia di Perpustakaan Institut Teknologi Bandung, dan terbuka untuk umum dengan ketentuan bahwa hak cipta ada pada pengarang. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau peringkasan hanya dapat dilakukan seizin pengarang dan harus disertai dengan kebiasaan ilmiah untuk menyebutkan sumbernya.

Memperbanyak atau menerbitkan sebagian atau seluruh tesis haruslah seizin Direktur Program Pascasarjana, Institut Teknologi Bandung.

Perpustakaan yang meminjam tesis ini untuk keperluan anggotanya harus mengisi nama dan tanda tangan peminjam dan tanggal pinjam.

KATA PENGANTAR

Alhamdulillahi rabbil ‘alamin. Puji syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat dan karunia yang dilimpahkan sehingga penulis dapat menyelesaikan tesis ini. Shalawat dan salam tercurah kepada Rasulullah Muhammad SAW beserta keluarganya.

Selama melaksanakan tesis ini, penulis mendapat bantuan dan dukungan dari berbagai pihak. Untuk itu, penulis mengucapkan terima kasih kepada:

1. bapak Dr. Ary Setijadi Prihatmanto, ST., MT., selaku Pembimbing I, yang telah memberikan bimbingan dan dorongan semangat selama pelaksanaan tesis, terutama dalam hal penyelesaian produk RESTU;
2. bapak Ir. Tunggal Mardiono, M.Sc., selaku Pembimbing II, yang telah memberikan bimbingan dan perhatian bahkan terhadap hal-hal yang tidak terkait dengan tesis;
3. ibu Dr. Ir. Aciek Ida Wuryandari, MT., bapak Dr. Ir. Hilwadi Hindersyah, M.Sc., dan bapak Dr. Pranoto Hidaya Rusmin, ST., MT., selaku Tim Penguji Sidang Tesis, yang telah memberikan kesempatan bagi penulis untuk mempresentasikan hasil penelitian dan mengemukakan pendapat;
4. Departemen Pendidikan Nasional atas bantuan Beasiswa Unggulan yang diterima penulis selama menjalani pendidikan program magister;
5. papa, mama, tante, dan kakak-kakak, beserta seluruh keluarga yang senantiasa memberikan semangat dan do’anya;
6. Brenda Ariesty Kusumasari, seorang *supporter* setia dan sahabat seperjuangan untuk ’mengejar’ Juli;
7. rekan Andik Taufiq, Nur Ichsan Utama, Erik Prabowo Kamal, Willy Derbyanto, dan Rio Andita Setiabakti (Alm.), tim utama dalam pengembangan RESTU;
8. rekan M. Hakim Adiprasetya dan Syarif Rousyan Fikri, tim *support* dalam pengembangan RESTU;
9. rekan-rekan *Microsoft Innovation Centre* (MIC) dan *Digital Signal Processing Research and Technology Group* (DSP-RTG);

10. rekan-rekan Teknologi Media Digital dan Game (TMDG) angkatan 2009;
11. seluruh staf dan karyawan Laboratorium Sistem Kendali dan Komputer; dan
12. semua pihak yang membantu, yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa tesis ini bukanlah tanpa kelemahan, baik dari segi ilmu yang disampaikan maupun teknik penulisannya. Oleh karena itu, penulis mengharapkan adanya kritik dan saran yang dapat disampaikan melalui *email* dengan alamat `aa.nugraha@yahoo.com`. Akhir kata, penulis berharap tesis ini dapat memberikan ilmu dan manfaat bagi yang membacanya.

Bandung, Juni 2011

Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Halaman

Halaman

DAFTAR SINGKATAN

Singkatan	Nama	Pemakaian pertama kali pada halaman
AGI	*Artificial General Intelligence*	58
AI	*Artificial Intelligence*	19
AIML	*Artificial Intelligence Markup Language*	1
ALICE	*Artificial Linguistic Internet Computer Entity*	1
CA	*Conversational Agent*	1
CCC	*Classical Cross-Correlation*	13
DFT	*Discrete Fourier Transform*	14
ECA	*Embodied Conversational Agent*	2
FFT	*Fast Fourier Transform*	14
GCC	*Generalized Cross-Correlation*	13
GUI	*Graphical User Interface*	19
HRTF	*Head-Related Transfer Function*	8
IDFT	*Inverse Discrete Fourier Transform*	14
IID	*Interaural Intensity Difference*	8
ILD	*Interaural Level Difference*	8
ITB	Institut Teknologi Bandung	3
ITD	*Interaural Time Difference*	8
JST	Jaringan Syaraf Tiruan	6
LSKK	Laboratorium Sistem Komputer dan Kendali	5
MSE	*Mean Squared Error*	46
PtPAR	*Peak-to-Peak Amplitude Ratio*	14
PUSPA	*PUSPA's an Understanding Synthespian that Provides Assistance*	3
RESTU	*RESTU's an Engine for Synthetic Thespian Units*	3
SAMsON	*Smart Assistant for Museum's Objects Navigation*	3
SpeaCal	*Speaker Localization*	19
SPL	*Sound Pressure Level*	9

Singkatan	Nama	Pemakaian pertama kali pada halaman
STEI	Sekolah Teknik Elektro dan Informatika	3
TDE	*Time Delay Estimation*	11
TDOA	*Time Difference of Arrival*	11
TOA	*Time of Arrival*	11
UI	*User Interface*	19
XML	*Extensible Markup Language*	1

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Istilah *dialog system*, *chatterbot*, atau *conversational agent* (CA) digunakan untuk menyebut sistem komputer yang mampu bercakap-cakap dengan manusia menggunakan bahasa alami [36]. Sistem ini dikenal sejak Alan Turing mendeskripsikan metode pengujian kecerdasaan buatan (kemudian dikenal sebagai *Turing Test*) yang dilakukan dengan cara melakukan percakapan antara manusia dan komputer menggunakan bahasa alami [39]. Meskipun demikian, *dialog system* sendiri baru terwujud pada pertengahan dekade 1960, ketika Joseph Weizenbaum mengembangkan ELIZA. Program yang ditujukan untuk mempelajari komunikasi antara manusia dan komputer menggunakan bahasa alami ini menganalisis dan melakukan dekomposisi teks masukan berdasarkan kata kunci yang ditemukan di dalamnya. Program kemudian menyusun tanggapan menggunakan aturan-aturan yang sesuai dengan kata kunci dan proses dekomposisi yang sebelumnya dilakukan [20,43]. Sejak kemunculan ELIZA, beberapa *chatterbot* dengan pendekatan-pendekatan yang berbeda bermunculan, antara lain: PARRY, SHRDLU, MegaHAL, CONVERSE, Elizabeth, Hexbot, dan ALICE [36, 37]. Salah satu *chatterbot* yang kemudian cukup berpengaruh pada perkembangan aplikasi *chatterbot* dalam satu dekade terakhir adalah *Artificial Linguistic Internet Computer Entity* (ALICE). *Chatterbot* yang meraih Loebner Prize, sebuah penghargaan yang diberikan kepada *chatterbot* yang dinilai menyerupai manusia oleh para juri pada kontes tahunan Turing Test, pada tahun 2000, 2001, dan 2004 ini menggunakan metode pencocokan pola seperti halnya ELIZA [24, 42]. Selain mengembangkan sebuah *chatterbot*, Richard Wallace, pengembang ALICE, juga mengembangkan *Artificial Intelligence Markup Language* (AIML). Skema *Extensible Markup Language* (XML) ini mempermudah pengembangan *chatterbot*, terutama dalam hal membangun basis pengetahuannya. Hal ini memicu semakin banyaknya aplikasi *chatterbot*, terutama yang berbasis internet, untuk berbagai keperluan, dengan berbagai basis pengetahuan, dan dalam berbagai bahasa [13, 41, 44].

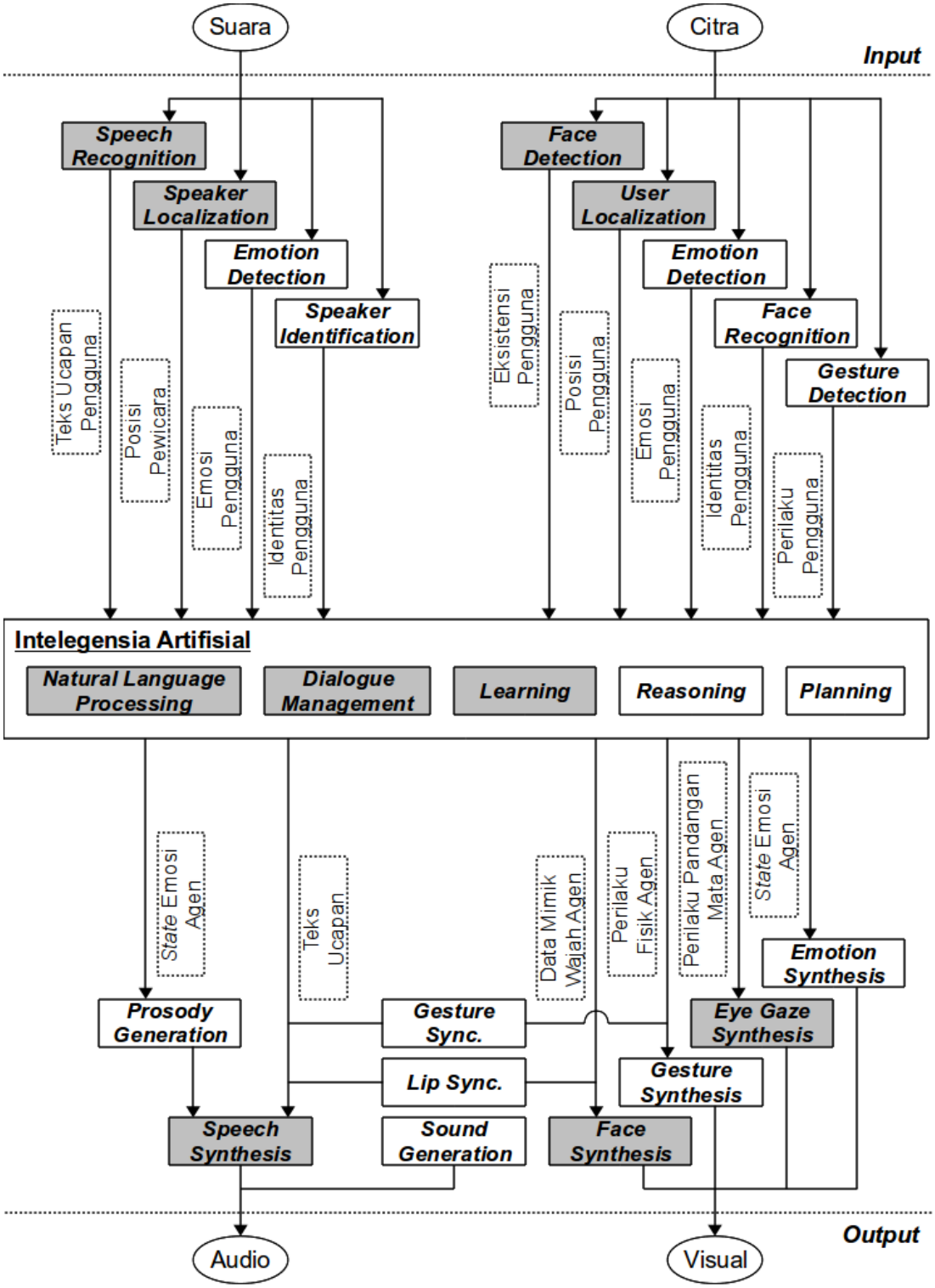
Perkembangan teknologi ucapan (*speech technology*) memungkinkan digunakannya suara ucapan manusia sebagai masukan *dialog system* dengan memanfaatkan teknologi pengenalan ucapan. Sistem yang dikenal dengan nama *spoken dialog system* ini kemudian memberi tanggapan dalam suara ucapan yang dimungkinkan oleh teknologi sintesis ucapan. Singkatnya, komunikasi dengan bahasa alami antara manusia dan komputer pada *spoken dialog system* berbasis ucapan, tidak lagi berbasis teks seperti pada *dialog system* terdahulu. Selain perkembangan antarmuka tersebut, perkembangan juga terjadi pada metode penyusunan tanggapan. Pada *spoken dialog system*, dikenal adanya *dialog manager*. *Dialog manager* berperan seperti bagian pencocokan pola pada ELIZA. Akan tetapi, berbeda dengan metode pencocokan pola pada ELIZA yang hanya mempertimbangkan masukan yang akan ditanggapi, *dialog manager* pada *spoken dialog system* mempertimbangkan konteks percakapan, yang disimpulkan dari percakapan yang terjadi sebelumnya, dalam menyusun tanggapan [10, 20, 27].

Conversational agent, yang sebelumnya hanya mampu menerima masukan dan memberi keluaran dalam bentuk teks atau suara saja, kemudian berevolusi menjadi *conversational agent* yang memiliki wujud, misalnya berupa karakter manusia dalam bentuk 3-dimensi (3D), yang dikenal sebagai *embodied conversational agent* (ECA) atau *embodied conversational interface agent*. Wujud yang dimiliki oleh agen memungkinkan lebih banyak cara yang dapat digunakan dalam komunikasi antara manusia dan komputer, misalnya arah tatapan mata, gerakan badan, dan mimik muka. Dengan kata lain, agen memiliki kemampuan untuk melakukan komunikasi non-verbal dengan pengguna, seperti yang dilakukan dalam komunikasi tatap muka antar manusia, sehingga berpotensi menciptakan komunikasi yang lebih alami dan pengalaman interaksi yang lebih kaya bagi penggunanya [3, 5, 6, 8, 12, 34, 35]. Beberapa contoh implementasi ECA, antara lain: agen serba guna Greta [30], agen properti REA [8], tutor pembelajaran bahasa Baldi [26], dan pemandu museum Ada dan Grace [38]. Ada dan Grace, yang namanya diambil dari Ada Lovelace dan Grace Hopper (dua tokoh wanita di sejarah perkembangan komputer), merupakan agen pemandu museum virtual yang ada di Museum of Science, Boston, tepatnya di bagian ”InterFaces” yang memamerkan benda-benda bersejarah di bidang komputer, robotika, dan teknologi komunikasi. Ada dan Grace dapat menjawab pertanyaan pengunjung, menjelaskan dan menyarankan suatu benda museum, serta menjelaskan bagaimana sistem dialog mereka bekerja (Gambar I.1). Pengetahuan dan karakter kedua agen didesain berbeda untuk memungkinkan terjadinya dialog yang lebih hidup dan memberikan pengalaman yang lebih kaya bagi pengunjung saat menjelajah museum [47, 48].

Gambar I.1 Interaksi pemandu museum virtual Ada dan Grace dengan pengunjung di Museum of Science, Boston [49].

Secara umum, implementasi ECA belum terlihat di Indonesia. Padahal, ECA dapat dimanfaatkan di berbagai bidang, misalnya di bidang pendidikan dan bidang pariwisata. ECA dapat dimanfaatkan sebagai tutor dalam proses pembelajaran. ECA juga dapat dimanfaatkan untuk membangun kios informasi pariwisata yang kemudian dapat ditempatkan di lokasi-lokasi yang mudah dijangkau oleh wisatawan, seperti di bandara, stasiun, terminal, hotel, atau bahkan di objek pariwisatanya sendiri. Implementasi ECA sebagai pemandu museum seperti Ada dan Grace dapat juga dilakukan di Indonesia untuk menarik lebih banyak orang untuk mengunjungi museum mengingat jumlah rata-rata pengunjung museum-museum di Indonesia pada kurun waktu 2006-2008 hanya mencapai 4,3 juta atau hanya 1,8 persen dari 237 juta lebih penduduk Indonesia (data tahun 2010) [45, 46].

Oleh karena itu, tim PUSPA dan SAMsON dari Program Magister Teknik Elektro, Sekolah Teknik Elektro dan Informatika (STEI), Institut Teknologi Bandung (ITB) bekerjasama dalam mengembangkan *RESTU's an Engine for Synthetic Thespian Units* (RESTU), yang merupakan *engine* untuk membangun ECA. Sebagai sebuah *engine*, RESTU terdiri dari bagian interaksi dan bagian kognisi. Secara umum, bagian interaksi memanfaatkan monitor untuk menampilkan wujud agen, mikrofon sebagai telinga, kamera sebagai mata, dan speaker sebagai mulut. Sedangkan, bagian kognisi dapat dipilih untuk memanfaatkan AIML atau menggunakan *artificial general intelligence* (AGI). Tim PUSPA sendiri memiliki tujuan membangun *PUSPA's an Understanding Synthespian that Provides Assistance* (PUSPA), yaitu sebuah ECA yang berperan sebagai asisten pribadi. Sedangkan, tim SAMsON memiliki tujuan membangun *Smart Assistant for Museum's Objects Navigation* (SAMsON), yaitu sebuah ECA yang berperan sebagai pemandu museum [31].



Gambar I.2 Diagram blok teknologi penyusun RESTU.

Gambar I.2 menunjukkan berbagai teknologi terkait ECA yang dapat diimplementasikan pada RESTU. Pada saat ini, teknologi yang diimplementasikan masih terbatas pada

teknologi-teknologi yang dalam Gambar I.2 diberi *background* berwarna abu-abu. Di luar teknologi-teknologi yang sudah tercantum, masih banyak teknologi yang berpotensi meningkatkan kemampuan interaksi ECA. Sebagai contoh, pada pengolahan masukan berupa suara (ucapan), terdapat teknologi *speaker segmentation* dan *speaker tracking*. Implementasi kedua teknologi ini pada RESTU akan memberi kemampuan pada ECA yang dihasilkan untuk mengelola komunikasi dengan pengguna lebih dari satu (*multi-speaker environment*) [25].

Seperti yang dapat dilihat pada Gambar I.2, pada dasarnya komunikasi antara ECA dengan penggunanya hanya memanfaatkan media suara (audio) dan gambar (visual). Meskipun demikian, dari media audiovisual ini, berbagai komunikasi non-verbal dapat berlangsung, misalnya emosi pengguna dapat diketahui dari intonasi suaranya, ekspresi wajahnya, sikap tubuhnya, atau tatapan matanya. Di sisi lain, agen virtual juga dapat menunjukkan emosi yang beragam dengan memberi intonasi tertentu pada suara yang dihasilkan, ekspresi wajah tertentu, sikap tubuh serta pergerakan anggota badan tertentu, atau bahkan dengan tatapan mata tertentu [17, 28].

Dalam komunikasi tatap muka antar manusia, tatapan mata berperan penting dalam komunikasi non-verbal. Contoh yang paling sederhana adalah tatapan mata dapat memberi petunjuk kepada siapa seseorang sedang berbicara dan dapat memberi petunjuk kondisi emosi seseorang. Oleh karena ECA memiliki wujud, termasuk mata, tatapan mata sebagai salah satu metode komunikasi non-verbal ini juga dapat diimplementasikan untuk menciptakan komunikasi antara manusia dan komputer yang sama alaminya dengan komunikasi antar manusia. Implementasi tatapan mata dan pergerakan mata yang berbeda pada ECA juga telah terbukti memberikan kesan dan tingkat kepuasan yang berbeda bagi penggunanya [23, 40]. Oleh karena itu, sebagai *engine* ECA, RESTU juga harus mengakomodasi fitur ECA yang berupa tatapan dan pergerakan mata agen.

Informasi yang dibutuhkan oleh agen dalam menentukan ke arah mana mata agen harus menatap adalah lokasi pengguna atau lawan bicara agen. Dalam RESTU, penentuan lokasi pengguna mungkin dilakukan dengan memanfaatkan informasi yang didapatkan dari: (1) sinyal suara yang ditangkap oleh mikrofon; dan/atau (2) citra yang ditangkap oleh kamera. Mempertimbangkan persyaratan *modularity* dan *extensibility* terkait modalitas masukan dan keluaran yang termasuk dalam persyaratan arsitektur ECA seperti yang tercantum dalam [7], penentuan lokasi pengguna dalam RESTU juga harus dapat dilakukan hanya berdasarkan sinyal suara saja atau gambar saja. Meskipun demikian, hasil penentuan lokasi kedua metode tersebut juga dapat dikombinasikan

untuk memperoleh lokasi pengguna yang lebih akurat. Penentuan lokasi menggunakan citra yang ditangkap kamera cenderung menghasilkan nilai posisi yang lebih akurat. Akan tetapi, area dimana metode ini dapat digunakan lebih sempit dibandingkan area dimana metode penentuan lokasi menggunakan suara dapat digunakan. Analogi sederhananya adalah telinga manusia dapat mendengar suara yang berasal dari suatu posisi yang terletak di luar jangkauan pandangan mata, misalnya sebuah posisi yang terletak di belakang kepala.

1.2 Tujuan Penelitian

Secara umum, tujuan penelitian ini adalah merancang sebuah subsistem penentuan lokasi pengguna, yang merupakan komponen dari sistem ECA, berdasarkan sinyal suara yang ditangkap oleh mikrofon. Secara lebih khusus, penelitian ini bertujuan: (1) merancang subsistem penentuan lokasi pengguna untuk RESTU dengan memanfaatkan perangkat keras yang murah dan mudah didapatkan, serta perangkat lunak *open source*; (2) mengimplementasikan rancangan yang dihasilkan pada RESTU; serta (3) mengujicoba subsistem penentuan lokasi pengguna yang telah diintegrasikan dalam RESTU.

1.3 Batasan Masalah

Dalam perancangan subsistem penentuan lokasi pengguna berdasarkan sinyal suara ini, beberapa asumsi penting yang digunakan adalah sebagai berikut.

1. Hanya ada satu sumber suara yang dideteksi pada satu waktu.
2. Tidak ada derau dengan intensitas yang tinggi sedemikian sehingga suara ucapan yang tertangkap oleh mikrofon masih terdengar dengan jelas dan dominan.
3. Tidak ada efek akustik dari ruangan, misalnya gema atau gaung. Dengan kata lain, mikrofon hanya menangkap sinyal yang bersifat *direct path* dari sumber suara.

Subsistem penentuan lokasi pengguna yang dirancang memanfaatkan perangkat keras yang murah dan mudah didapatkan, serta perangkat lunak *open source*. Perangkat

keras utama yang dibutuhkan oleh subsistem ini adalah komputer, *sound card*, dan mikrofon. Komputer yang digunakan memiliki prosesor Intel® Core™ 2 Duo T8100 2,1 GHz dan memori 2 GB. *Sound card* yang digunakan adalah *USB sound card* tanpa merk yang memiliki dua kanal keluaran (hanya mendukung *sample rate* 48 KHz) dan satu kanal masukan (hanya mendukung *sample rate* 24 KHz). Sedangkan, mikrofon yang digunakan adalah mikrofon *omnidirectional* merk Genius. Subsistem dirancang dan diimplementasikan pada sistem operasi Ubuntu (Linux). Oleh karena kompatibilitas terhadap sistem operasi belum dipertimbangkan, subsistem tidak dapat dipastikan berjalan pada sistem operasi selain Ubuntu (Linux).

Subsistem penentuan lokasi pengguna yang dirancang akan diintegrasikan ke dalam RESTU. Sebagai *showcase*, RESTU akan digunakan untuk membangun sebuah ECA yang berperan sebagai pemandu atau pusat informasi Laboratorium Sistem Komputer dan Kendali (LSKK), ITB. Perangkat keras yang akan digunakan untuk menampilkan pemandu virtual tersebut adalah perangkat *multitouch screen* vertikal milik LSKK yang berdimensi 139 x 60 x 180 centimeter (panjang x lebar x tinggi) seperti yang terlihat pada Gambar III.4 dan Gambar III.5. Layar berada di bagian atas dari salah satu sisi terluas perangkat. Pengguna diasumsikan manusia dewasa dengan tinggi badan 150-170 cm yang berdiri di depan layar pada jarak yang wajar diambil saat melakukan komunikasi tatap muka antar manusia (kurang lebih 1 meter). Pengguna juga diasumsikan menghadap ke layar saat berbicara, sedemikian sehingga sinyal suara yang dihasilkan dapat ditangkap oleh mikrofon-mikrofon yang ditempelkan pada perangkat.

1.4 Sistematika Pembahasan

BAB I menguraikan sejarah singkat dan potensi-potensi pemanfaatan CA, serta proyek pengembangan RESTU, sebuah *engine* ECA, yang kemudian menjadi latar belakang dan motivasi perancangan subsistem penentuan lokasi pengguna berdasarkan sinyal suara. Bab ini juga mendefinisikan batasan-batasan yang digunakan dalam perancangan subsistem tersebut.

BAB II membahas landasan teori terkait subsistem penentuan lokasi pengguna berdasarkan sinyal suara yang dirancang dalam penelitian ini. Pembahasan mencakup metode-metode penentuan lokasi sumber suara berdasarkan sinyal suara yang ditangkap.

BAB III menjelaskan arsitektur RESTU secara umum dan hubungan subsistem yang dirancang dalam penelitian ini dengan subsistem-subsistem lain yang ada di dalam RESTU, serta menguraikan proses perancangan subsistem penentuan lokasi pengguna berdasarkan sinyal suara, yang meliputi perancangan perangkat keras dan lunak, proses pengambilan dan pengolahan data yang digunakan untuk melatih jaringan syaraf tiruan, serta proses pengujian dan analisis hasil pengujiannya.

BAB IV membahas proses integrasi subsistem penentuan lokasi pengguna berdasarkan sinyal suara ke dalam RESTU dan proses pengujiannya.

Sedangkan, BAB V memuat kesimpulan dan saran berdasarkan penelitian yang dilakukan.

BAB II

TINJAUAN PUSTAKA

2.1 Gelombang Suara Ucapan Manusia

Gelombang bunyi adalah gelombang longitudinal yang dihasilkan oleh suatu sumber bunyi dan merambat melalui suatu medium. Cepat rambat gelombang bunyi di udara kira-kira $331.5 + 0.6T_c\, m/s$, dengan T_c adalah temperatur udara dalam satuan Celcius [18].

Suara ucapan merupakan gelombang bunyi yang dihasilkan oleh pita suara manusia. Dalam menghasilkan ucapan, pita suara dapat bergetar dan menghasilkan gelombang dengan frekuensi yang disebut sebagai frekuensi fundamental (F_0) atau *pitch*. Frekuensi fundamental yang dapat dihasilkan oleh pita suara manusia berbeda antara satu orang dengan orang lainnya. Frekuensi fundamental yang umumnya dapat dihasilkan oleh pita suara manusia adalah sebesar 100 Hz pada pria hingga 250 Hz pada wanita dan anak-anak [33]. Sumber lain menyebutkan bahwa frekuensi fundamental yang umumnya dapat dihasilkan adalah sebesar 60 Hz pada pria hingga 300 Hz pada wanita dan anak-anak [18].

Pada umumnya, energi dari sebuah sinyal berada pada frekuensi fundamental hingga harmonik kesepuluhnya. Oleh karena itu, untuk mendapatkan representasi sinyal suara ucapan manusia yang baik, jangkauan frekuensi yang harus dapat ditangkap adalah F_0 hingga $10\, F_0$ [1]. Sebagai contoh, apabila mengacu pada frekuensi fundamental yang dinyatakan dalam [18], jangkauan frekuensi yang harus dapat ditangkap agar sinyal suara ucapan baik pria, wanita, mau pun anak-anak dapat direpresentasikan dengan baik adalah 60 Hz hingga 3000 Hz. Di bidang telefoni, kanal suara pada jaringan telepon memiliki lebar pita 4000 Hz (frekuensi 0-4000 Hz), meski transmisi suara hanya memanfaatkan frekuensi dalam jangkauan 300-3300 Hz (lebar pita 3000 Hz) [11].

Berdasarkan teorema Nyquist, untuk memastikan suatu sinyal analog yang dikonversi menjadi sinyal digital dapat direproduksi dengan baik, frekuensi *sampling* (*sampling*

rate) sekurang-kurangnya dua kali frekuensi tertinggi dari sinyal analog yang di-*sampling* [1, 11]. Oleh karena itu, apabila diasumsikan bahwa frekuensi tertinggi dari sinyal suara ucapan manusia adalah 4000 Hz, maka frekuensi *sampling* paling rendah yang boleh digunakan adalah sebesar 8000 Hz.

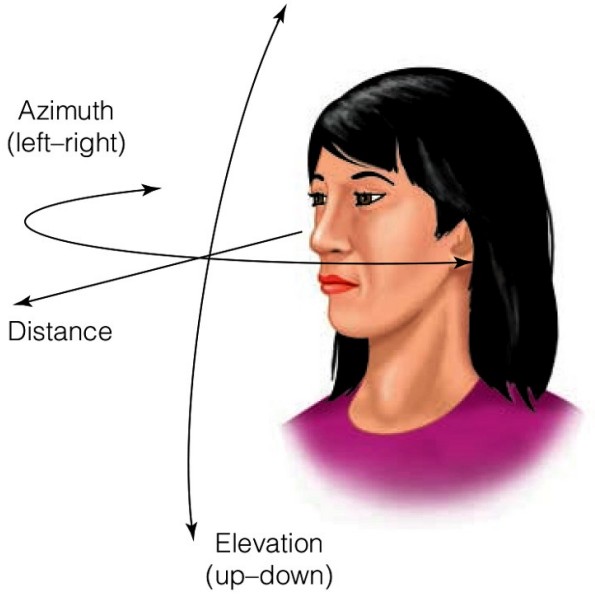
2.2 Penentuan Lokasi Sumber Suara

Manusia dengan indra pendengaran yang berfungsi dengan baik dapat memperkirakan lokasi sumber suara. Ketika seseorang dipanggil oleh rekannya dari belakang, orang tersebut dapat memperkirakan bahwa suara panggilan berasal dari belakang sehingga orang tersebut akan menengok ke arah belakang. Selain itu, seseorang juga dapat memperkirakan arah (sudut) datangnya suara dengan indra pendengaran, yang lokasi sumber suaranya kemudian dapat dikonfirmasi oleh indra penglihatan setelah orang tersebut menengok ke arah perkiraan datangnya suara. Kemampuan menentukan lokasi sumber suara (*sound localization*) yang dimiliki secara alami oleh manusia ini juga akan diimplementasikan pada agen virtual yang dihasilkan oleh RESTU.

2.2.1 Penentuan Lokasi Sumber Suara Pada Manusia

Menurut para peneliti bidang psikoakustik, manusia menggunakan tiga dimensi dalam menentukan lokasi sumber suara [15]. Dimensi-dimensi tersebut adalah: (1) *azimuth*, (2) elevasi, dan (3) jarak (Gambar II.1). Untuk menentukan *azimuth*, manusia menggunakan parameter-parameter binaural, yaitu: (1) *interaural time difference* (ITD); (2) *interaural level difference* (ILD), yang dikenal juga sebagai *interaural intensity difference* (IID); dan (3) perubahan spektral (*coloration*) dari bunyi yang mencapai telinga bagian dalam [14, 15, 21]. Untuk menentukan elevasi, manusia menggunakan parameter monaural berupa *head-related transfer function* (HRTF). Sedangkan, untuk menentukan jarak, belum ada parameter yang pasti [14, 15]. Pada Subbab 2.2 ini, pembahasan terkait metode penentuan lokasi sumber suara pada manusia hanya mencakup parameter-parameter binaural, yaitu ITD dan ILD.

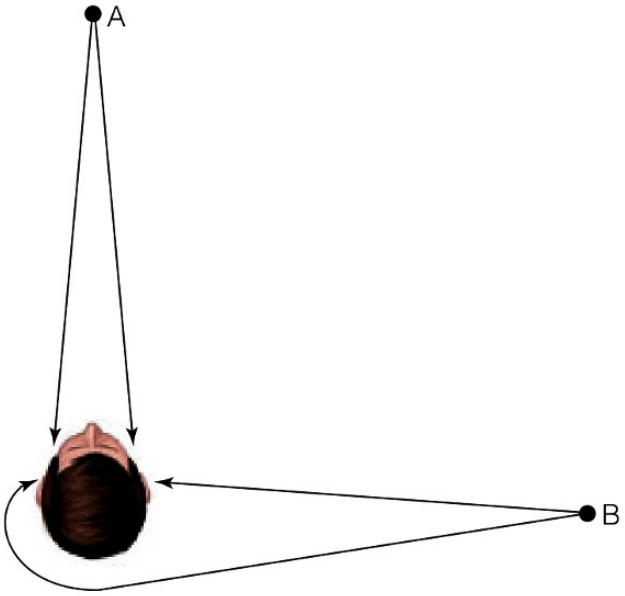
Parameter-parameter binaural didapatkan dengan membandingkan gelombang bunyi yang ditangkap oleh telinga kanan dan kiri. Pada ITD, variabel yang dibandingkan adalah waktu kedatangan gelombang bunyi pada kedua telinga tersebut. Apabila jarak sumber bunyi terhadap kedua telinga sama, misalnya sumber bunyi berada tepat di



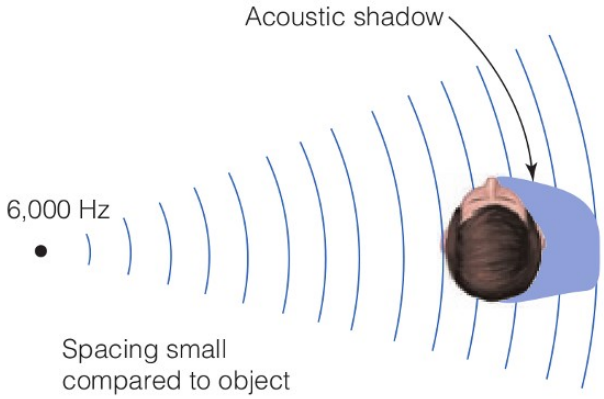
Gambar II.1 Tiga dimensi yang digunakan manusia untuk menentukan lokasi sumber suara [15].

depan atau di belakang kepala pendengar, parameter ITD bernilai nol (titik A pada Gambar II.2). Parameter ITD akan bernilai tidak nol apabila sumber bunyi berada pada *azimuth* tertentu relatif terhadap arah depan kepala. Sebagai contoh, gelombang bunyi yang bersumber pada titik B (Gambar II.2) akan diterima oleh telinga kanan terlebih dahulu. Pada gelombang sinusoidal, ITD akan bersifat ambigu antara *leading* atau *lagging* saat perbedaan fasa antara dua gelombang sinyal mencapai 180^o. Oleh karena itu, ITD rentan menghasilkan nilai yang tidak tepat pada gelombang dengan frekuensi tinggi.

Pada ILD atau IID, variabel yang dibandingkan adalah *sound pressure level* (SPL) yang diterima oleh telinga kanan dan kiri. Perbedaan SPL yang diterima oleh kedua telinga disebabkan karena kepala membentuk sebuah *acoustic shadow* yang meredam gelombang bunyi yang melaluinya, sehingga intensitas gelombang bunyi yang diterima oleh telinga yang jauh lebih kecil. Meskipun demikian, peredaman ini hanya signifikan pada gelombang dengan frekuensi tinggi (Gambar II.3 dan Gambar II.4). Oleh karena itu, penentuan *azimuth* lokasi sumber suara pada manusia mengombinasikan ITD dan ILD. ITD untuk gelombang bunyi dengan frekuensi rendah dan ILD untuk gelombang bunyi dengan frekuensi tinggi.

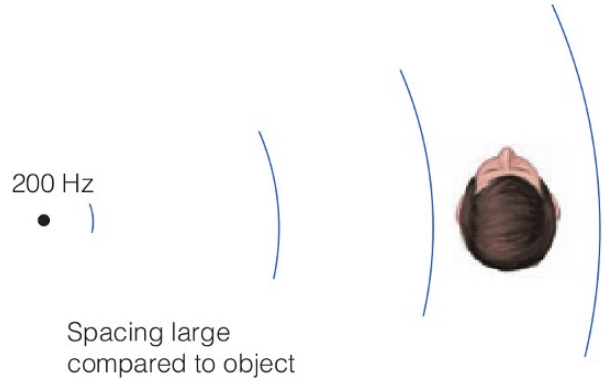


Gambar II.2 Ilustrasi *interaural time difference* (ITD) [15].



Gambar II.3 Ilustrasi *interaural level difference* (ILD) pada gelombang dengan frekuensi tinggi [15].

Penentuan lokasi sumber suara yang digunakan RESTU untuk agen virtual (ECA) akan menggunakan konsep yang sama dengan ITD dan ILD, yaitu membandingkan waktu kedatangan dan amplitudo dari gelombang suara ucapan yang tertangkap oleh sepasang mikrofon.

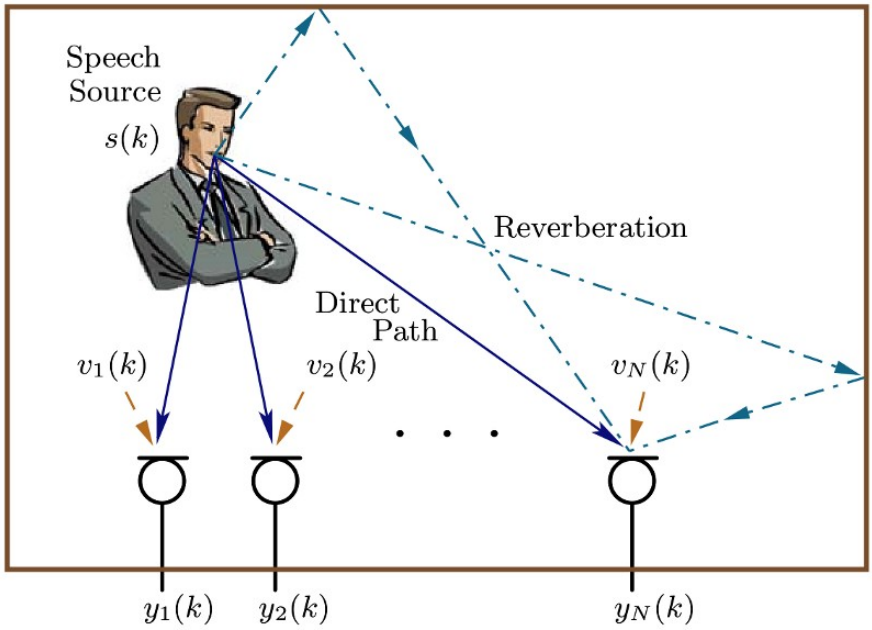


Gambar II.4 Ilustrasi *interaural level difference* (ILD) pada gelombang dengan frekuensi rendah [15].

2.2.2 *Time Difference of Arrival* (TDOA)

Parameter *time difference of arrival* (TDOA) digunakan untuk mengukur perbedaan waktu kedatangan antara suara yang ditangkap oleh mikrofon yang satu dengan mikrofon yang lain dalam sebuah konfigurasi *microphone array*. Pengukuran TDOA dikenal juga sebagai *time delay estimation* (TDE), meskipun sebenarnya TDE sendiri dapat dibagi menjadi pengukuran TDOA dan pengukuran *time of arrival* (TOA) [19]. Pada Subbab 2.2 ini, pembahasan metode TDE terbatas mengacu pada asumsi-asumsi yang telah didefinisikan pada Subbab 1.3.

Secara umum, dalam metode TDE terdapat dua model sinyal, yaitu model *free-field* ideal dan model *reverberant* [2, 19]. Dalam [9], terdapat satu model lagi yang disebut model *multipath*. Tidak seperti model *free-field* ideal yang hanya memperhitungkan sinyal yang bersifat *direct path*, model *multipath* dan model *reverberant* memperhitungkan efek akustik dari ruangan, misalnya pantulan gelombang bunyi (Gambar II.5). Perbedaan di antara keduanya adalah pada model *multipath* jumlah pantulan dapat diperkirakan, sehingga jumlah *path* gelombang dari sumber bunyi ke mikrofon dapat diketahui dan dapat dihitung. Pada model ini, terdapat variabel τ_{lm} yang menyatakan waktu tunda relatif antara mikrofon ke-l dengan mikrofon ke-0 untuk *path* ke-m, dengan $\tau_{01} = 0$. Sedangkan, pada model *reverberant* jumlah *path* sangat banyak sehingga nilai τ_{lm} sangat sulit untuk dihitung.



Gambar II.5 Ilustrasi model propagasi ideal dan model *reverberant* [2].

Model *free-field* ideal, juga disebut sebagai model propagasi ideal, mengasumsikan bahwa mikrofon hanya menangkap sinyal yang bersifat *direct path*. Dengan demikian, mikrofon diasumsikan menangkap sinyal yang teredam dan tertunda karena pengaruh propagasi dari sinyal asli yang dihasilkan oleh sumber suara, serta menangkap derau yang bersifat aditif. Mengacu ke skenario yang tergambar pada Gambar II.5, sinyal yang ditangkap mikrofon n dari sinyal s pada waktu ke-k dapat dinyatakan sebagai Persamaan II.1.

$$y_n(k) = \alpha_n s(k - \tau_n) + v_n(k), \qquad n = 1, 2, \cdots, N \tag{II.1}$$

Pada Persamaan II.1 tersebut, α_n adalah faktor redaman karena propagasi ($0 \leq \alpha_n \leq 1$), τ_n adalah waktu yang dibutuhkan untuk propagasi, dan v_n adalah derau aditif. Sinyal derau v_n diasumsikan tidak berkorelasi dengan sinyal dari sumber suara dan derau yang tertangkap oleh mikrofon yang lain.

TDOA antara mikrofon ke-i dan ke-j dapat didefinisikan sebagai Persamaan II.2.

$$\tau_{ij} \triangleq \tau_i - \tau_j, \qquad i, j = 1, 2, \cdots, N \tag{II.2}$$

Metode yang paling sederhana dan paling umum digunakan dalam menghitung TDOA adalah *classical cross-correlation* (CCC) [2,9,19]. Mengacu ke skenario yang tergambar pada Gambar II.5, CCC antara sinyal $y_1(k)$ dan $y_2(k)$ dapat didefinisikan sebagai Persamaan II.3.

$$r_{y_1y_2}^{CCC}(p) = E[y_1(k)y_2(k+p)] \tag{II.3}$$

Kemudian, TDOA dapat diperoleh dari Persamaan II.4.

$$\widehat{\tau}_{y_1y_2}^{CCC} = \arg\max_{p} r_{y_1y_2}^{CCC}(p) \tag{II.4}$$

Pada Persamaan II.3 dan Persamaan II.4, $p \in [-\tau_{max}, \tau_{max}]$ dan τ_{max} adalah TDOA maksimum yang mungkin terjadi.

CCC sendiri merupakan kasus khusus dari *generalized cross-correlation* (GCC) [2, 9, 19]. TDOA pada GCC diperoleh dari Persamaan II.5.

$$\widehat{\tau}_{y_1y_2}^{GCC} = \arg\max_{\tau} r_{y_1y_2}^{GCC}(\tau) \tag{II.5}$$

Sedangkan, persamaan GCC didefinisikan sebagai Persamaan II.6.

$$\begin{aligned} r_{y_1y_2}^{GCC}(p) &= F^{-1}[\Psi_{y_1y_2}(f)] \\ &= \int_{-\infty}^{\infty} \Psi_{y_1y_2}(f) e^{j2\pi fp} df \\ &= \int_{-\infty}^{\infty} \vartheta(f)\phi_{y_1y_2}(f) e^{j2\pi fp} df \end{aligned} \tag{II.6}$$

Pada Persamaan II.6 tersebut, $F^{-1}[\cdot]$ menyatakan *inverse discrete-time Fourier transform* (IDTFT), $\Psi_{y_1y_2}(f)$ adalah *generalized cross-spectrum*, dan $\phi_{y_1y_2}(f)$ adalah *cross-spectrum*.

Cross-spectrum didefinisikan sebagai Persamaan II.7, dengan $Y_n(f)$ didefinisikan sebagai Persamaan II.8.

$$\phi_{y_1 y_2}(f) = E[Y_1(f)Y_2^*(f)] \tag{II.7}$$

$$Y_n(f) = \sum_k y_n(k)e^{-j2\pi fk}, \qquad n = 1, 2 \tag{II.8}$$

Pada Persamaan II.8, $(\cdot)^*$ menyatakan konjugat kompleks.

Generalized cross-spectrum didefinisikan sebagai Persamaan II.9.

$$\Psi_{y_1 y_2}(f) = \vartheta(f)\phi_{y_1 y_2}(f) \tag{II.9}$$

Pada Persamaan II.9 tersebut, $\vartheta(f)$ adalah fungsi pembobotan pada domain frekuensi. Pada CCC, $\vartheta(f) = 1$.

Persamaan-persamaan GCC di atas dan hubungannya dengan CCC menunjukkan bahwa perhitungan CCC dapat dilakukan menggunakan *discrete Fourier transform* (DFT) dan *inverse* DFT (IDFT), yang dapat diimplementasikan secara efisien memanfaatkan *fast Fourier transform* (FFT).

2.2.3 *Peak-to-Peak Amplitude Ratio* (PtPAR)

Parameter *peak-to-peak amplitude ratio* (PtPAR) digunakan untuk mengukur perbedaan amplitudo gelombang antara sinyal suara yang ditangkap oleh mikrofon yang satu dengan mikrofon yang lain.

Intensitas gelombang suara pada sebuah ruang dapat dirumuskan sebagai Persamaan II.10.

$$I = \frac{P_{av}}{A} = \frac{P_{av}}{4\pi r^2} \tag{II.10}$$

Pada Persamaan II.10 tersebut, P_{av} adalah daya rata-rata yang dihasilkan oleh sumber suara, A adalah luas permukaan bola, dan r adalah jari-jari bola [16]. Dengan demikian, hubungan antara intensitas gelombang suara I dengan sebuah titik yang berjarak r dari sumber suara dapat dinyatakan sebagai Persamaan II.11.

$$I \propto \frac{1}{r^2} \tag{II.11}$$

Sedangkan, daya suatu gelombang dapat dirumuskan sebagai Persamaan II.12.

$$P = \frac{1}{2}\mu\omega^2 A^2 v \tag{II.12}$$

Pada Persamaan II.12 tersebut, μ adalah rapat massa per satuan panjang medium, ω adalah frekuensi angular gelombang, A adalah amplitudo gelombang, dan v adalah cepat rambat gelombang [16]. Dengan demikian, hubungan antara intensitas gelombang suara I dengan amplitudo gelombang suara A dapat dinyatakan sebagai Persamaan II.13.

$$I \propto A^2 \tag{II.13}$$

Dari Persamaan II.11 dan Persamaan II.13, hubungan antara amplitudo gelombang suara A dengan sebuah titik yang berjarak r dari sumber suara dapat dinyatakan sebagai Persamaan II.14.

$$A \propto \frac{1}{r} \tag{II.14}$$

Dengan demikian, hubungan perbandingan jarak titik 1 dan 2 dari sumber suara (r_1 dan r_2) dengan perbandingan amplitudo gelombang yang tertangkap pada kedua titik tersebut (A_1 dan A_2) dapat dinyatakan sebagai Persamaan II.15.

$$\frac{A_1}{A_2} = \frac{r_2}{r_1} \tag{II.15}$$

Mengacu pada Persamaan II.14 dan skenario yang tergambar pada Gambar II.5, sinyal yang ditangkap mikrofon n dari sinyal s pada waktu ke-k dengan mengabaikan waktu yang dibutuhkan oleh propagasi gelombang dapat dinyatakan dalam persamaan matematika sebagai Persamaan II.16.

$$y_n(k) = \frac{s(k)}{r_n} + v_n(k), \qquad n = 1, 2, \cdots, N \tag{II.16}$$

Pada Persamaan II.16 tersebut, r_n adalah jarak antara mikrofon n dari sumber suara dan v_n adalah derau aditif [4].

Berdasarkan eksperimen yang dilakukan dalam [4], dengan menggunakan parameter ILD saja, penentuan lokasi sumber suara dapat dilakukan secara akurat.

BAB III

PERANCANGAN SPEACAL UNTUK RESTU

3.1 *RESTU's an Engine for Synthetic Thespian Units* (RESTU)

Pengembangan *engine* ECA yang diberi nama *RESTU's an Engine for Synthetic Thespian Units* (RESTU) ini berangkat dari konsep yang terdapat pada *conversational agent*, yaitu gagasan akan mampunya komputer melakukan perbincangan dengan manusia dalam bahasa alami manusia. *Conversational agent*, yang sebelumnya hanya mampu menerima masukan dan memberi keluaran dalam bentuk teks atau suara saja, kemudian berevolusi menjadi *conversational agent* yang memiliki wujud, misalnya berupa karakter manusia dalam bentuk 3-dimensi (3D), yang dikenal sebagai *embodied conversational agent* (ECA) atau *embodied conversational interface agent*. Wujud yang dimiliki oleh agen memungkinkan lebih banyak cara yang dapat digunakan dalam komunikasi antara manusia dan komputer, misalnya arah tatapan mata, gerakan badan, dan mimik muka. Dengan kata lain, agen memiliki kemampuan untuk melakukan komunikasi non-verbal dengan pengguna, seperti yang dilakukan dalam komunikasi tatap muka antar manusia, sehingga berpotensi menciptakan komunikasi yang lebih alami dan pengalaman interaksi yang lebih kaya bagi pengguna.

Fitur dasar dari ECA yang dibangun dengan *engine* ini adalah mampu melakukan perbincangan dengan penggunanya dalam Bahasa Indonesia. Fitur dasar produk ECA yang lain akan sangat bergantung pada peran agen virtual. Sebagai contoh, apabila produk ECA ditujukan untuk berperan sebagai asisten, agen virtual akan mampu memahami permintaan penggunanya dan berusaha mewujudkannya. Apabila produk ECA ditujukan untuk berperan sebagai pemandu museum, agen virtual akan mampu memahami pertanyaan pengguna dan berusaha untuk menjawab serta menjelaskannya. Meskipun demikian, perlu diingat bahwa tindakan-tindakan tersebut sangat bergantung pada pengetahuan yang dimiliki oleh agen virtual. Sebagai contoh, agen virtual yang berperan sebagai pemandu museum hanya dapat menjawab pertanyaan-pertanyaan terkait museum. Fitur yang lain adalah agen virtual diwujudkan dalam tampilan 3D yang dilengkapi dengan kemampuan menggerakkan anggota badannya. Selain dapat

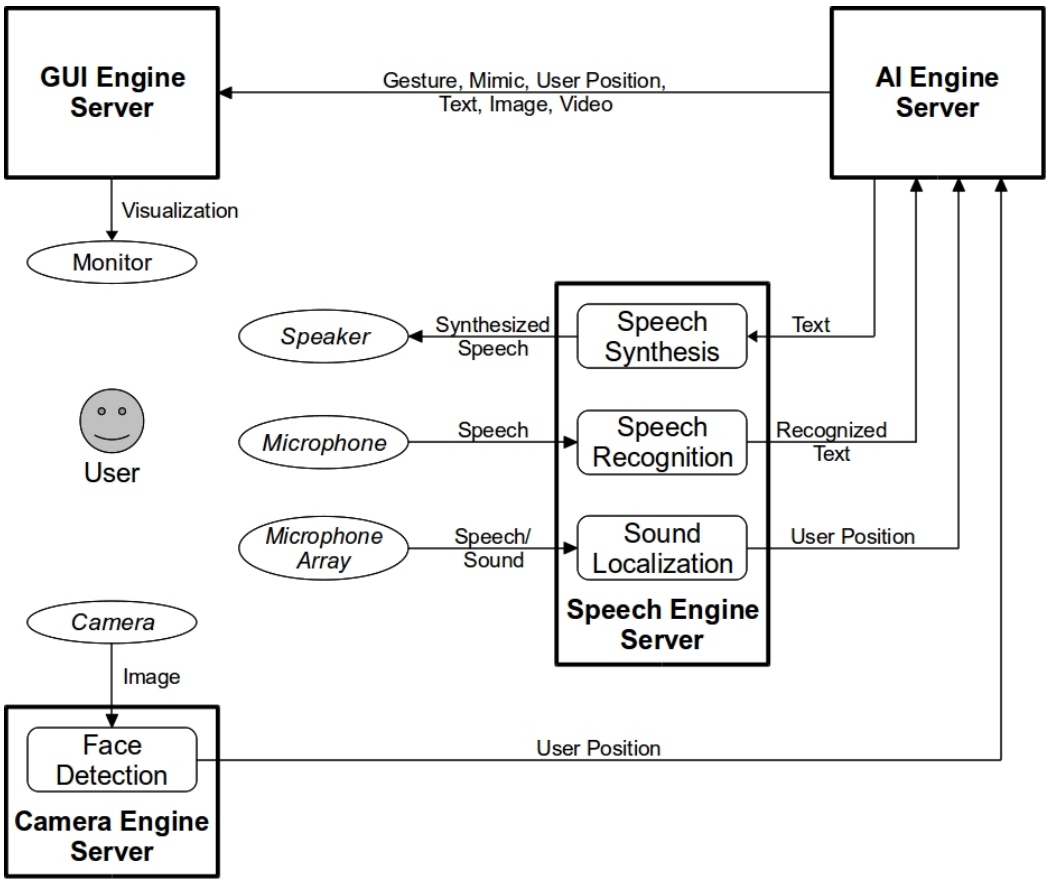
menggerakkan bibir saat berbicara, agen virtual dapat menengokkan kepalanya untuk memandang lawan bicaranya. Agen juga dapat menunjukkan pose tertentu untuk mencapai tujuannya, misalnya menunjukkan suatu gambar ke pengguna, atau sekedar untuk membangun suasana percakapan yang tidak kaku.

Gambar III.1 Interaksi pengguna dengan prototipe pemandu virtual LSKK yang dibangun dengan RESTU.

Secara umum, RESTU disusun oleh teknologi pemrosesan bahasa alami, pemrosesan suara, pemrosesan citra, grafis 3D, dan kecerdasan artifisial. Teknologi pemrosesan bahasa alami, mencakup teknologi pengenalan suara dan sintesis suara. Teknologi pemrosesan suara digunakan dalam fungsi penentuan lokasi pengguna berdasarkan suara dan teknologi pemrosesan citra digunakan dalam fungsi pengenalan wajah untuk menentukan lokasi pengguna berdasarkan citra. Teknologi grafis digunakan untuk mewujudkan karakter virtual, perilakunya, dan lingkungannya. Sedangkan, teknologi kecerdasan artifisial dimanfaatkan untuk menyusun respons terhadap masukan dari pengguna yang diterima oleh agen berdasarkan pengetahuan yang dimilikinya.

Fungsi-fungsi yang menyusun RESTU tersebut dapat dikelompokkan menjadi dua bagian, yaitu bagian kognitif dan bagian interaksi. Fungsi kecerdasan artifisial menyusun bagian kognitif. Sedangkan, bagian interaksi tersusun dari fungsi pengenalan

suara, sintesis suara, penentuan lokasi pengguna baik menggunakan suara mau pun citra, dan antarmuka grafis. Dengan mengikuti pengelompokan tersebut, di sisi implementasi RESTU dibangun dengan menggunakan dua kelompok *server*, yaitu *Artificial Intelligence (AI) Engine Server* dan *User Interface (UI) Engine Server*. UI Engine Server dapat dibagi menjadi *Camera Engine Server*, *Speech Engine Server*, dan *Graphical User Interface (GUI) Engine Server*. Diagram aliran data antar *server*, atau modul yang ada di dalamnya, ditunjukkan oleh Gambar III.2.



Gambar III.2 Diagram aliran data RESTU.

3.2 *Speaker Localization* (SpeaCal)

SpeaCal, yang berasal dari istilah *Speaker Localization*, merupakan subsistem penentuan posisi pengguna berdasarkan sinyal suara, yang dalam Gambar III.2 disebut sebagai *Sound Localization*. Subbab 3.2 ini akan menguraikan proses perancangan SpeaCal yang menggunakan model perancangan iteratif. Perancangan melalui tahap

pendefinisian kebutuhan, penentuan spesifikasi, pembuatan desain, pengimplementasian desain, serta pengujian dan evaluasi. Proses iterasi dilakukan pada tiga tahap yang disebutkan terakhir.

Tahap pembuatan desain meliputi pembuatan desain perangkat lunak dan desain perangkat keras, yang kemudian akan diimplementasikan pada tahap implementasi. Sedangkan, tahap pengujian dan evaluasi meliputi pengambilan data latih dan data uji yang akan digunakan untuk melatih dan menguji JST. Analisis kemudian dilakukan terhadap semua data yang diperoleh, tanpa membedakan antara data latih dan data uji, untuk menentukan apakah desain perangkat lunak dan perangkat keras dapat menghasilkan data yang cukup konsisten sehingga dapat digunakan untuk melatih JST. Apabila data tidak cukup konsisten, proses perancangan akan kembali pada tahap pembuatan desain.

3.2.1 Definisi Kebutuhan

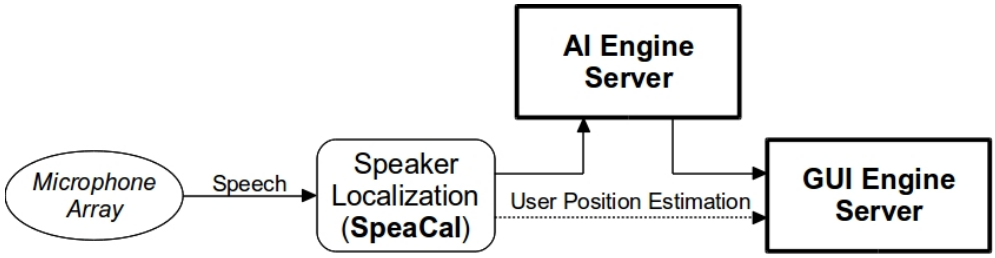
SpeaCal dikembangkan spesifik untuk RESTU. Dari Bab I dan Subbab 3.1, fungsi penentuan posisi pengguna dibutuhkan untuk mengetahui ke arah mana mata atau kepala agen virtual harus menatap atau menoleh. Dalam komunikasi tatap muka antar manusia, kontak mata diperlukan untuk menunjukkan perhatian terhadap lawan bicara dan topik pembicaraan. meskipun demikian, kontak mata tidak berarti mata seseorang *selalu* menatap lawan bicara *tepat* di matanya selama percakapan.

Dengan demikian, SpeaCal harus dapat menghasilkan informasi *perkiraan* posisi pengguna berdasarkan suara pengguna yang ditangkap oleh mikrofon. Informasi perkiraan posisi yang menjadi prioritas utama adalah informasi *azimuth*. Informasi ini kemudian harus dapat digunakan oleh *AI Engine*, atau *GUI Engine* secara langsung, untuk menentukan ke arah mana mata atau kepala agen virtual harus menatap atau menoleh.

Definisi kebutuhan di atas dapat digambarkan oleh Gambar III.3.

3.2.2 Spesifikasi

Dari definisi kebutuhan di atas, SpeaCal harus mampu memperkirakan posisi pengguna berdasarkan suara pengguna yang ditangkap oleh mikrofon. Untuk melakukannya,



Gambar III.3 Ilustrasi definisi kebutuhan SpeaCal.

SpeaCal akan menggunakan parameter ITD dan/atau ILD yang membandingkan dua sinyal. Oleh karena itu, diperlukan lebih dari satu mikrofon untuk menangkap suara pengguna. Pada manusia, dua indera pendengaran dapat digunakan untuk memperkirakan *azimuth* posisi sumber suara. Mengacu pada fakta ini, SpeaCal akan menggunakan empat buah mikrofon yang dipasang di bagian atas, bawah, kanan, dan kiri perangkat. Dengan mikrofon-mikrofon tersebut (selanjutnya disebut *microphone array*) diharapkan informasi posisi sumber suara yang diperoleh tidak hanya *azimuth*, tetapi juga *elevation*.

Untuk menangkap sinyal dari mikrofon dibutuhkan *sound card*. Sinyal suara yang akan ditangkap oleh mikrofon adalah suara manusia, sehingga satu kanal suara (mono) saja cukup untuk merepresentasikan sinyal yang ditangkap dengan baik. Dengan demikian, untuk menangkap sinyal dari empat mikrofon dibutuhkan empat kanal suara. Dari hasil survei, *sound card* yang dapat diakses (dibeli) dengan mudah dan harganya murah adalah *USB sound card*. Perangkat seharga Rp 28.000,00 ini memiliki satu kanal masukan yang dapat dimanfaatkan untuk menerima sinyal dari mikrofon.

Untuk mengelola empat *sound card* ini dibutuhkan sebuah program (pustaka) yang mampu mengakses *buffer* yang dimiliki *sound card*, sehingga data sinyal yang tertangkap oleh mikrofon dapat disimpan dalam sebuah file. File-file representasi sinyal dari setiap mikrofon kemudian akan diolah untuk menghitung TDOA, sebagai parameter ITD, dan PtPAR, sebagai parameter ILD. Perhitungan TDOA akan menggunakan DFT agar komputasi dapat dilakukan dengan lebih cepat. Oleh karena itu, dibutuhkan pustaka yang mampu melakukan FFT.

Parameter TDOA dan/atau PtPAR inilah yang kemudian digunakan oleh JST untuk memperkirakan posisi pengguna. Untuk membangun sebuah JST diperlukan kumpulan data latih dan data uji yang memuat nilai masukan, berupa nilai TDOA dan/atau PtPAR, dan nilai keluaran, berupa posisi. Data latih digunakan untuk membangun

(melatih) JST, sedangkan data uji digunakan untuk menguji jaringan yang telah dilatih. JST tersebut kemudian digunakan untuk proses penentuan posisi pengguna saat SpeaCal telah diintegrasikan dalam RESTU. Oleh karena itu, dibutuhkan pustaka pustaka yang mampu membangun dan menggunakan JST.

Dengan demikian, SpeaCal harus mampu:

1. menangkap suara pengguna dengan menggunakan *microphone array*, yang terdiri dari empat mikrofon, secara bersamaan;
2. menghitung TDOA dari sinyal suara yang ditangkap oleh *microphone array*;
3. menghitung PtPAR dari sinyal suara yang ditangkap oleh *microphone array*;
4. menyimpan data TDOA, PtPAR, dan posisi untuk membuat data latih dan data uji JST;
5. melatih JST dengan nilai TDOA dan PtPAR yang didapatkan selama pengambilan data latih;
6. menguji JST dengan nilai TDOA dan PtPAR yang didapatkan selama pengambilan data uji;
7. menggunakan JST untuk menghasilkan informasi perkiraan posisi pengguna (sumber suara); dan
8. mengirimkan informasi perkiraan posisi pengguna ke subsistem lain (*AI Engine* atau *GUI Engine*).

Perangkat keras yang akan digunakan meliputi:

- perangkat *multitouch screen* vertikal milik LSKK yang berdimensi 139 x 60 x 180 centimeter (panjang x lebar x tinggi) (Gambar III.4, Gambar III.5);
- komputer:
  - untuk perancangan serta pengambilan data latih dan uji: laptop Dell XPS M1330 dengan prosesor Intel® Core™ 2 Duo T8100 2,1 GHz dan memori 2 GB, dan

  - untuk demo: komputer rakitan dengan prosesor Intel® Core™ i7 2,67 GHz dan memori 2 GB;

- *USB sound card* tanpa merk (4 buah), dengan dua kanal keluaran yang hanya mendukung *sample rate* 48 KHz dan satu kanal masukan yang hanya mendukung *sample rate* 24 KHz (Gambar III.6(a));
- *USB hub* merk Belkin, yang memiliki empat *port* (Gambar III.6(a));
- mikrofon (4 buah) merk Genius seri MIC-01A, yang bagian tangkainya dipotong dan bagian kepalanya ditancapkan pada *styrofoam* yang ditempelkan pada perangkat *multitouch screen* vertikal (Gambar III.6(b)); dan
- *speaker* merk Genius seri SP-i150;

Sedangkan, perangkat lunak yang akan digunakan meliputi:

- sistem operasi: Ubuntu 10.04.2 LTS;
- bahasa pemrograman: C, C++;
- IDE: CodeBlocks;
- pustaka: C Std Lib, C POSIX Lib, C++ Std Lib, Portaudio, FFTW3, FANN, Boost, wxWidgets;
- pengolah data: gedit, OpenOffice Calc;
- pendukung: Audacity, Octave.

3.2.3 Iterasi I: Desain

Perangkat Lunak

Dari Subbab 3.2.2 dapat didefinisikan tiga perangkat lunak (program) utama yang dibutuhkan dalam perancangan subsistem penentuan posisi pengguna berdasarkan sinyal suara, yaitu:

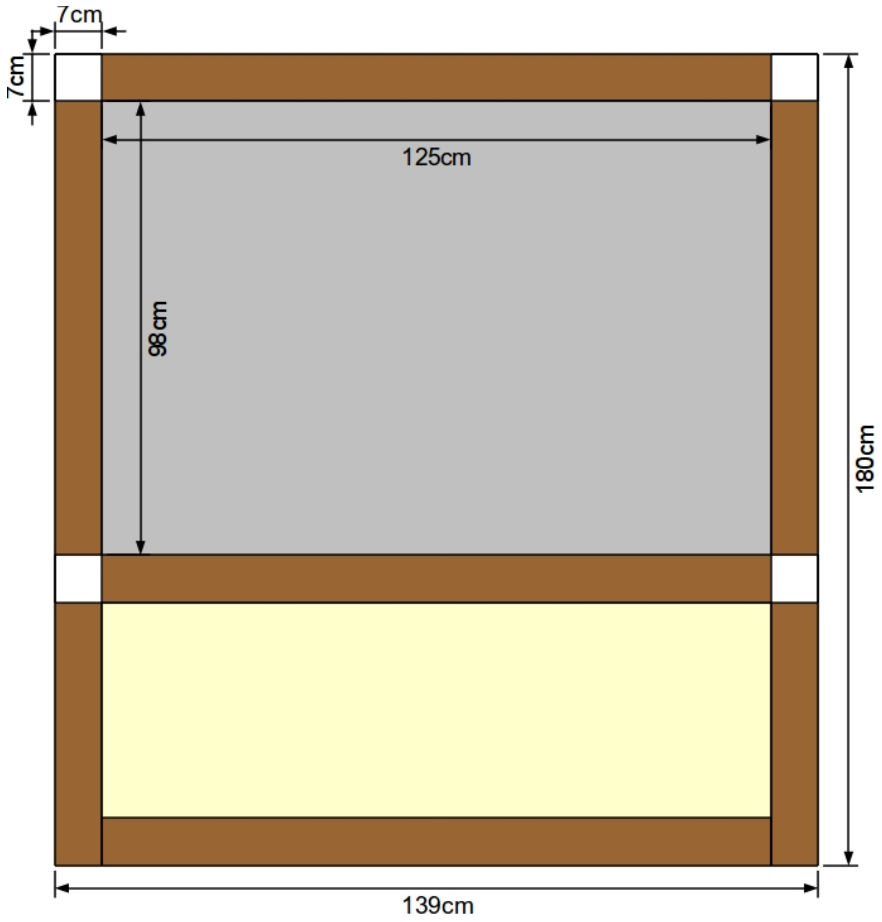
Gambar III.4 Foto perangkat *multitouch screen* vertikal yang akan digunakan untuk demo RESTU.

1. SpeaCalTrain, yaitu perangkat lunak (program) yang digunakan untuk memperoleh data latih dan data uji untuk JST;

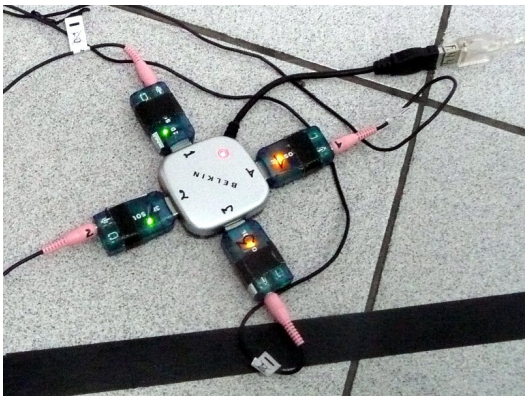
2. SpeaCal, yaitu perangkat lunak (program) yang digunakan untuk memperkirakan posisi pengguna memanfaatkan JST; dan

3. perangkat lunak (program) pendukung yang digunakan untuk melatih JST dengan memanfaatkan data latih dan mengujinya dengan memanfaatkan data uji.

Dalam Subbab 3.2.2 telah dijelaskan bahwa penentuan posisi pengguna akan memanfaatkan parameter TDOA dan PtPAR. Kedua parameter tersebut akan dihitung dari empat sinyal suara yang ditangkap oleh empat mikrofon. Oleh karena itu, fungsi penangkap sinyal suara serta fungsi penghitung parameter TDOA dan PtPAR menjadi inti dari SpeaCalTrain dan SpeaCal.



Gambar III.5 Dimensi bagian muka perangkat *multitouch screen* vertikal yang akan digunakan untuk demo RESTU.



(a) Foto *USB sound card* dan *USB hub*.

(b) Foto instalasi mikrofon pada perangkat *multitouch screen* vertikal.

Gambar III.6 Foto sebagian perangkat keras yang digunakan dalam perancangan.

Gambar III.8 menunjukkan alur program yang digunakan untuk mendapatkan data latih dan data uji untuk JST. Gambar III.11 menunjukkan alur program yang digunakan untuk memperkirakan posisi pengguna memanfaatkan JST. Sedangkan, Gambar III.14 menunjukkan alur program yang digunakan untuk melatih JST.

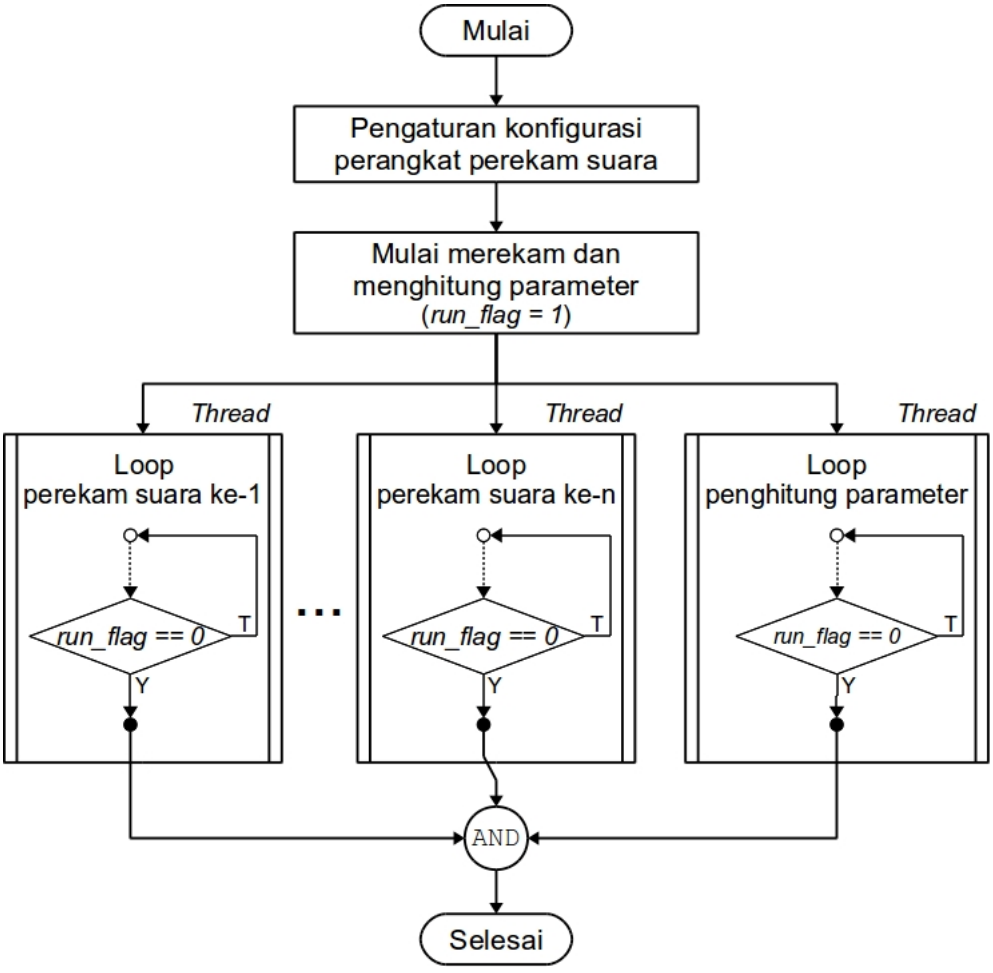
Gambar III.9 dan Gambar III.12 menunjukkan alur fungsi penangkap sinyal pada SpeaCalTrain dan SpeaCal. Sedangkan, Gambar III.10 dan Gambar III.13 menunjukkan alur fungsi penghitung parameter TDOA dan PtPAR. Pada prinsipnya, fungsi penangkap sinyal suara dan penghitung parameter yang digunakan pada kedua program adalah sama. Perbedaannya terletak pada adanya proses menyimpan data pada program SpeaCalTrain dan adanya proses menjalankan JST pada program SpeaCal.

Durasi sampel sinyal suara yang direkam oleh fungsi penangkap sinyal dapat diatur secara fleksibel. Akan tetapi, perlu diingat bahwa durasi sampel sinyal suara berpengaruh pada banyaknya *frame* yang harus diolah oleh fungsi penghitung parameter. Dari pengamatan, durasi sampel berbanding lurus dengan waktu yang dibutuhkan oleh fungsi penghitung parameter untuk menyelesaikan operasinya. Oleh karena itu, apabila durasi sampel sinyal diset x detik, fungsi penangkap sinyal harus diset ke *idle* selama x detik setiap kali selesai merekam sinyal selama x detik untuk memastikan bahwa fungsi penghitung parameter telah menyelesaikan operasinya. Dalam perancangan ini, durasi yang digunakan adalah 0,5 dan 1 detik, sehingga nilai perkiraan posisi pengguna diperbarui setiap 1 dan 2 detik.

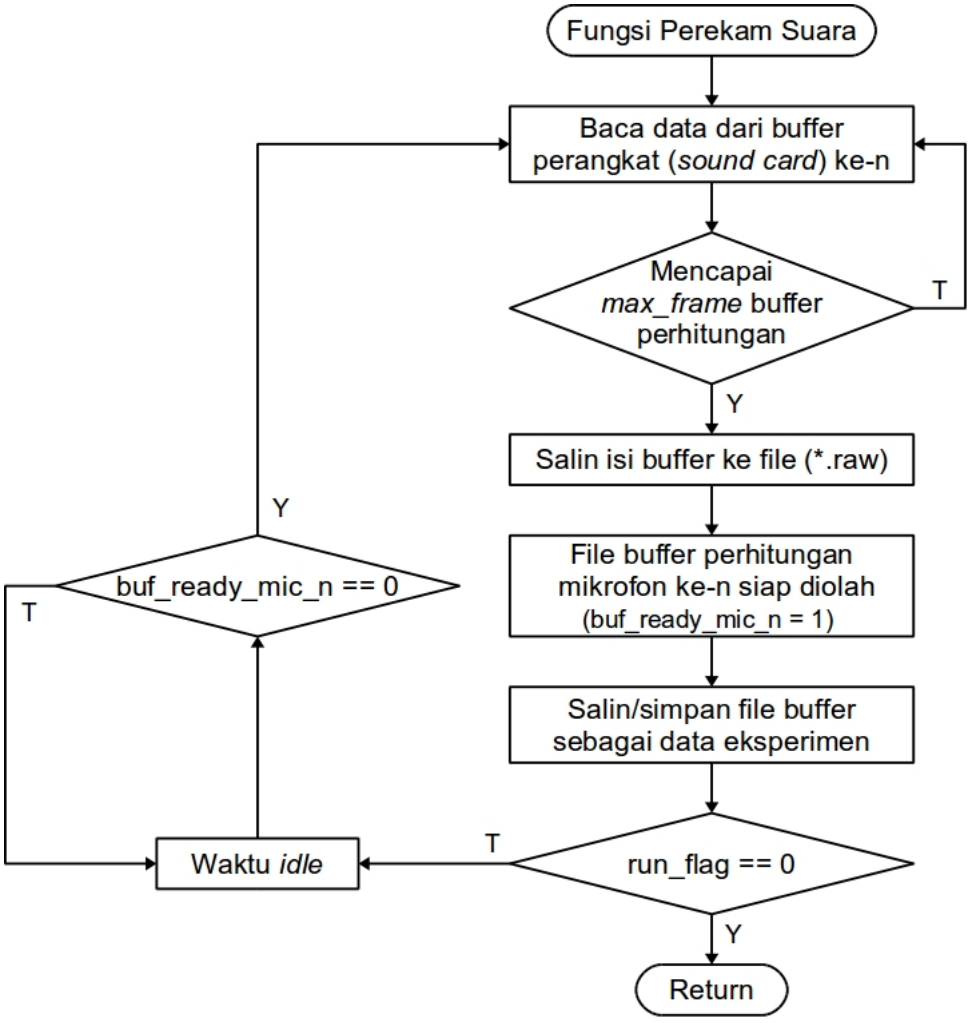
Data yang disimpan pada program SpeaCalTrain terdiri dari data suara dalam file berekstensi `raw` dan data TDOA serta PtPAR yang tertulis dalam sebuah file teks. Format penulisan file teks yang memuat data TDOA dan PtPAR disesuaikan dengan format data latih JST. Contoh penulisan file teks data tersebut dapat dilihat pada Gambar III.7. Baris pertama dalam file menunjukkan jumlah data (pasangan masukan dan keluaran) yang tercantum dalam file tersebut, jumlah masukan, dan jumlah keluaran. Dalam contoh yang tercantum pada Gambar III.7, jumlah data adalah 120, jumlah masukan 12, dan jumlah keluaran 3. Baris-baris selanjutnya terbagi atas dua macam, yaitu baris genap menuliskan data masukan dan baris ganjil menuliskan data keluaran yang bersesuaian. Dalam contoh yang tercantum pada Gambar III.7, baris data masukan mencantumkan enam data TDOA yang disusul dengan enam data PtPAR dan baris data keluaran mencantumkan representasi sebuah titik dalam koordinat Cartesian.

```
120 12  3
1.375 2.833 3.208 1.417 1.792 0.417 -7.214  -11.371 -15.296
    -4.156  -8.081  -3.925
190 30  60
-1.458  2.500 1.292 3.833 2.667 -1.125  -7.244  -11.366
    -15.586 -4.121  -8.341  -4.220
190 30  60
0.000 7.292 2.667 2.917 2.667 -0.250  -7.097  -11.227 -15.222
    -4.130  -8.125  -3.995
190 30  60
...
```

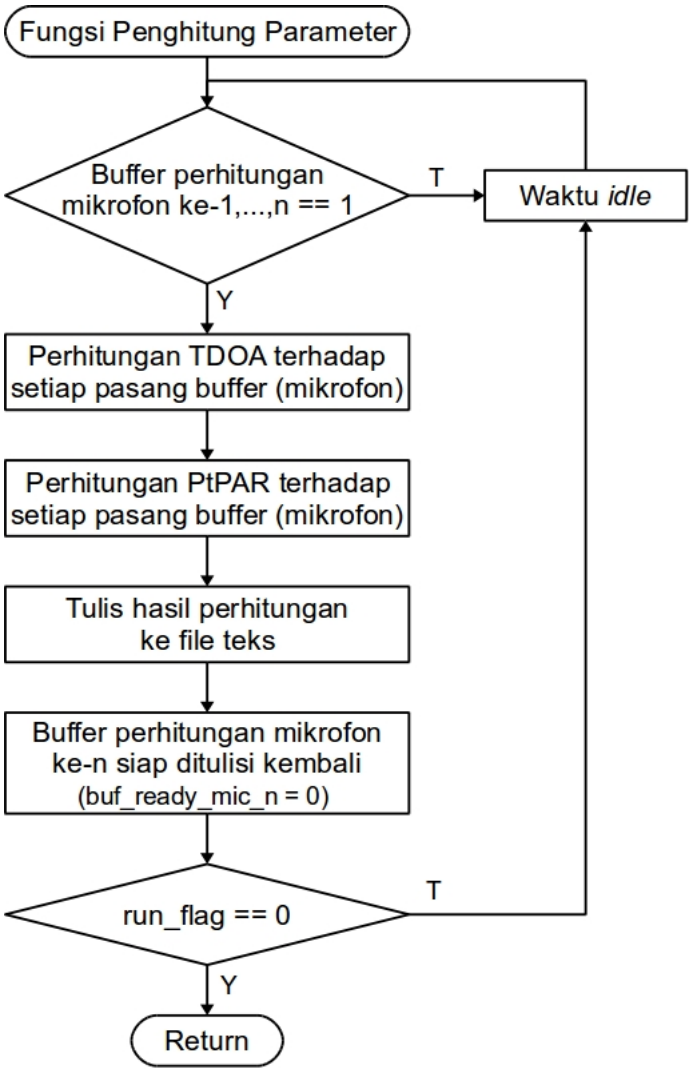
Gambar III.7 Contoh penulisan data TDOA dan PtPAR dalam file teks data.



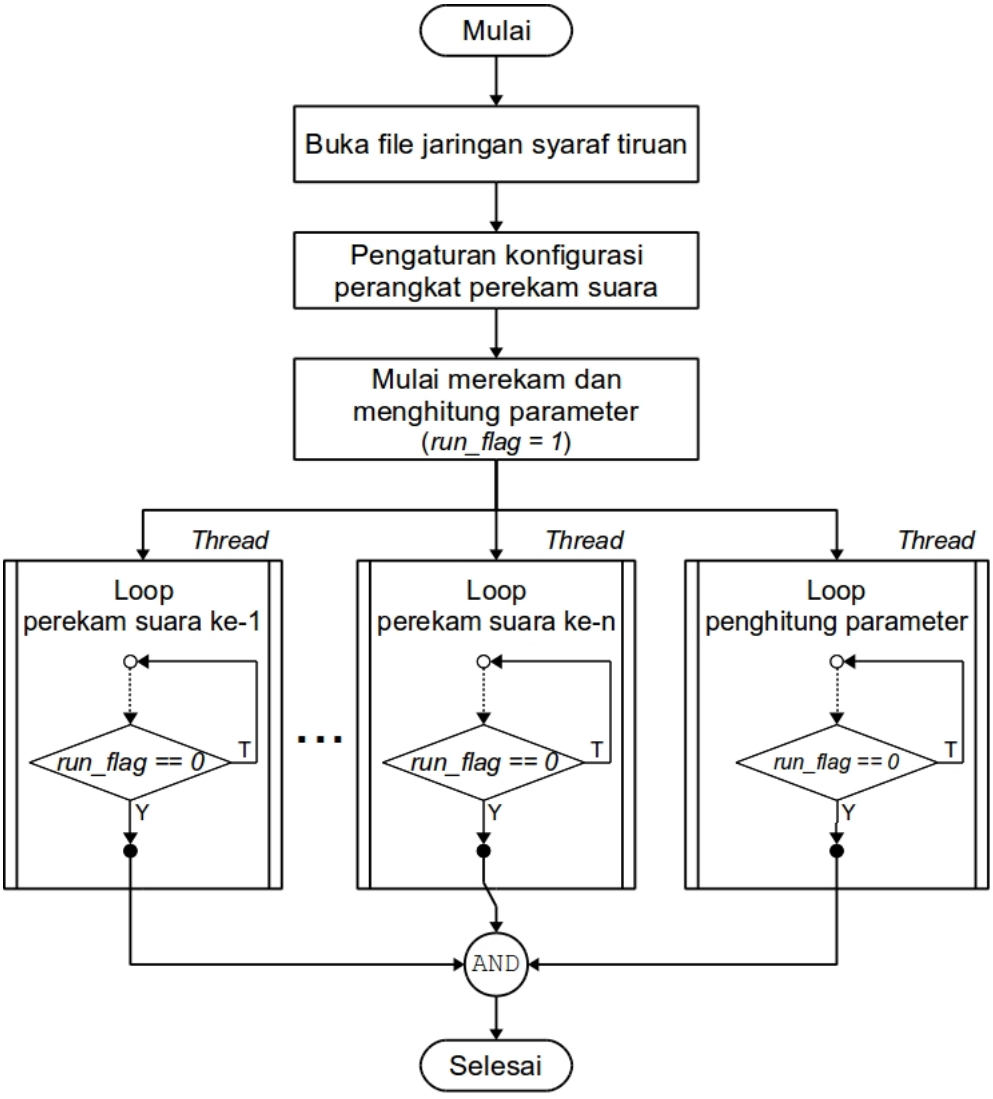
Gambar III.8 Diagram alir SpeaCalTrain.



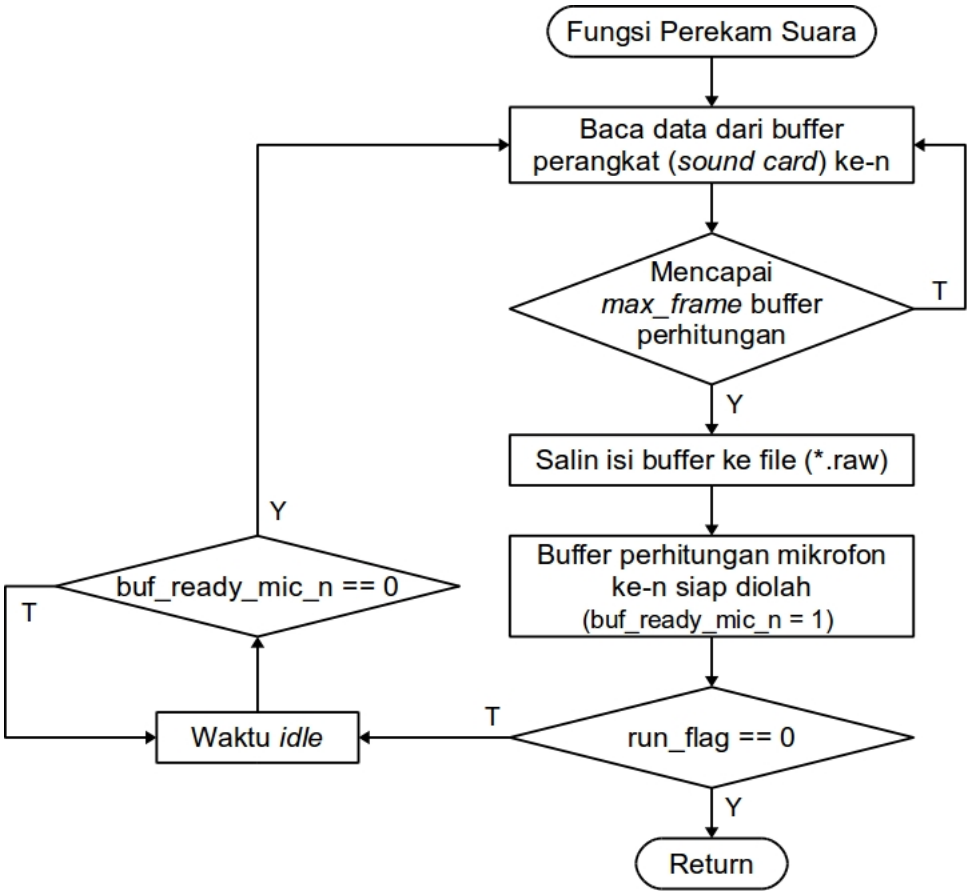
Gambar III.9 Diagram alir fungsi perekam suara pada SpeaCalTrain.



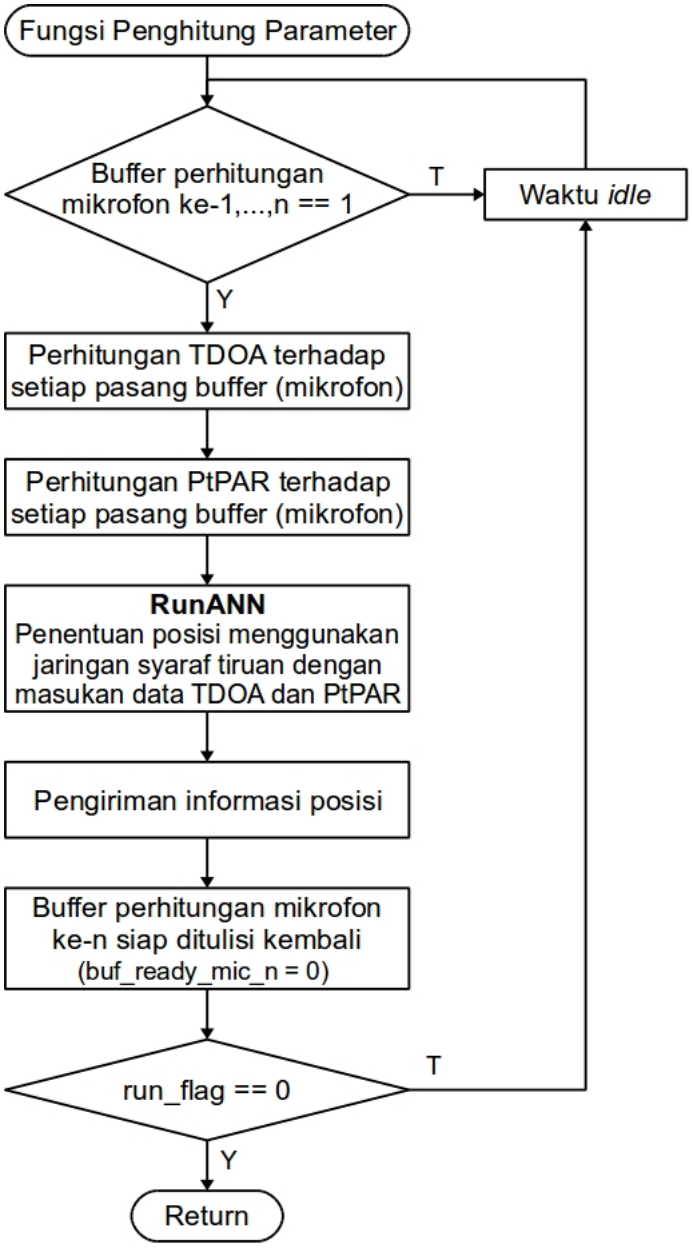
Gambar III.10 Diagram alir fungsi penghitung parameter TDOA dan PtPAR pada SpeaCalTrain.



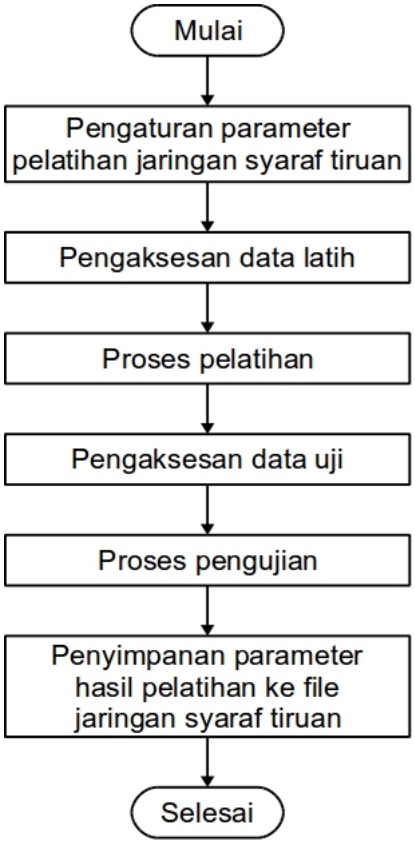
Gambar III.11 Diagram alir SpeaCal.



Gambar III.12 Diagram alir fungsi perekam suara pada SpeaCal.



Gambar III.13 Diagram alir fungsi penghitung parameter TDOA dan PtPAR pada SpeaCal.



Gambar III.14 Diagram alir program untuk melatih JST.

Perangkat Keras

Dalam penentuan posisi pengguna berdasarkan TDOA dan PtPAR, penempatan mikrofon akan sangat berpengaruh. Empat mikrofon yang digunakan akan ditempatkan pada perangkat *multitouch* vertikal seperti yang ditunjukkan oleh Gambar III.15. Dengan desain tersebut diharapkan posisi pengguna dalam ruang dapat diperkirakan. Apabila manusia dapat memperkirakan *azimuth* posisi sumber suara dengan dua telinga, secara logis *azimuth* posisi pengguna dapat diperkirakan dengan membandingkan sinyal yang tertangkap oleh mikrofon kanan dan kiri, sedangkan *elevation* dapat diperkirakan dengan sinyal dari mikrofon atas dan bawah. Selain itu, dengan adanya perbedaan jarak mikrofon kanan-kiri dan mikrofon atas-bawah terhadap permukaan layar, diharapkan jarak pengguna terhadap layar juga dapat diperkirakan memanfaatkan parameter TDOA dan PtPAR yang membandingkan kombinasi mikrofon kanan-kiri dan mikrofon atas-bawah, misalnya parameter TDOA dan PtPAR pasangan mikrofon atas dan kanan.

meskipun demikian, prioritas utama adalah memperolah *azimuth* posisi pengguna relatif terhadap perangkat *multitouch*.

Pengambilan Data

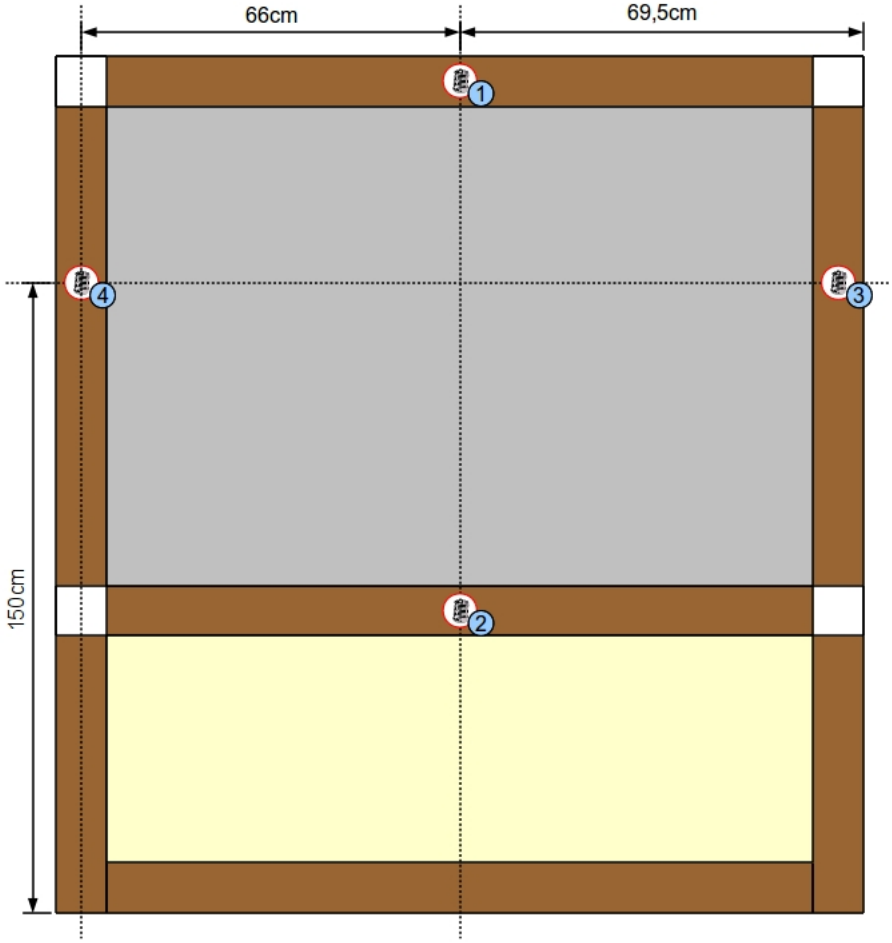
Pengambilan data latih dan uji untuk JST dilakukan dengan menempatkan sumber suara (*speaker*) pada koordinat tertentu relatif terhadap perangkat *multitouch* vertikal. Titik 0 dari koordinat Cartesian yang digunakan adalah ujung kiri atas perangkat *multitouch*. Pengguna diasumsikan merupakan manusia dewasa dengan tinggi 160-170 cm. Dengan demikian, dapat diasumsikan bahwa posisi mulut berada pada ketinggian 150 cm dari tanah. Oleh karena itu, sumber suara ditempatkan pada koordinat $z = -30$, sedangkan koordinat x dan y merupakan variabel. Jangkauan nilai kedua koordinat tersebut adalah $x = \{-50, 10, 70, 130, 190\}$ dan $y = \{60, 120, 180\}$ (ditandai dengan tanda X pada Gambar III.16).

3.2.4 Iterasi I: Implementasi

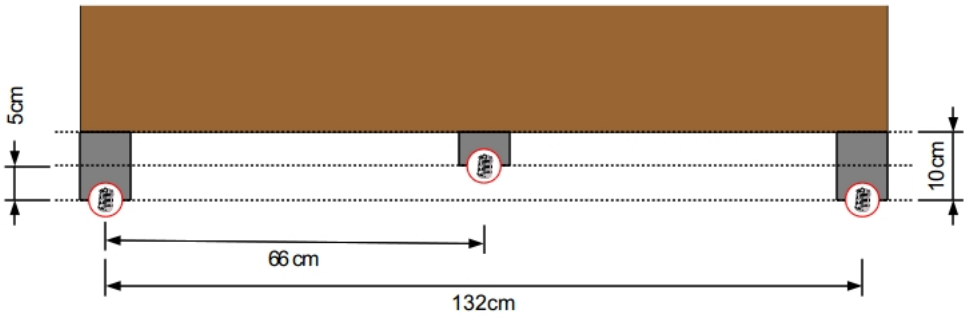
Perangkat Lunak

Implementasi fungsi perekam suara memanfaatkan pustaka Portaudio untuk mengakses *buffer* yang dimiliki *sound card* dan menyalinnya ke sebuah file yang berekstensi `raw`. File-file yang berisi representasi sinyal suara dari empat mikrofon inilah yang kemudian akan diolah oleh fungsi penghitung parameter.

Fungsi penghitung parameter terdiri dari penghitung parameter TDOA dan PtPAR. Metode CCC digunakan untuk menghitung TDOA dari sepasang data sinyal. Untuk memperoleh operasi yang cepat, metode CCC diimplentasikan menggunakan DFT. Oleh karena itu, fungsi ini memanfaatkan pustaka FFTW3 dalam proses penghitungan TDOA. Gambar III.19 menunjukkan implementasi metode CCC. Parameter TDOA dapat diperoleh dengan mencari nilai maksimum dari hasil operasi CCC.

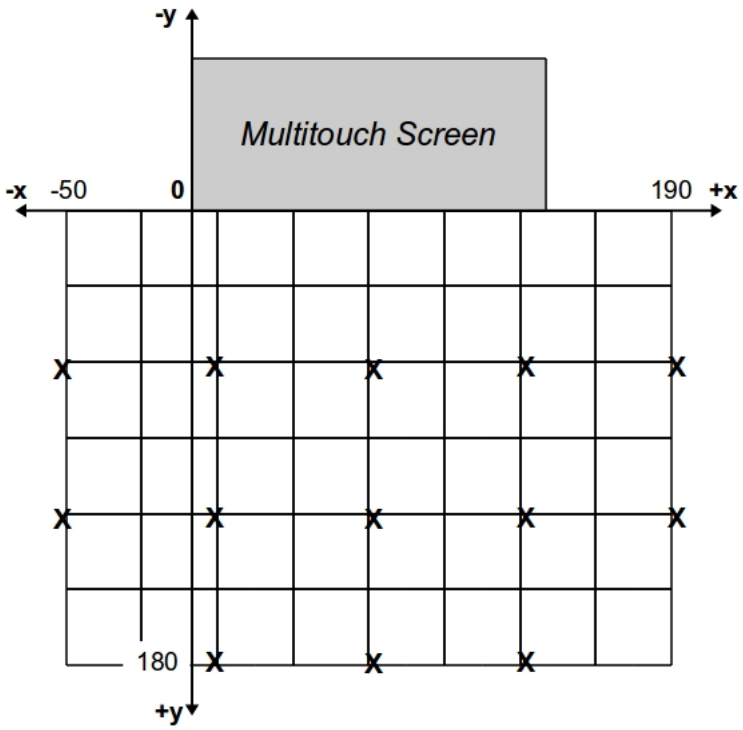


(a) Tampak muka



(b) Tampak atas

Gambar III.15 Rancangan penempatan mikrofon.



Gambar III.16 Rancangan titik pengambilan data.

```
if(inputBuffer == NULL)
{
  for(i=0; i<framesPerBuffer; i++)
    {
    output_data_buffer[i] = SAMPLE_SILENCE;  /* left */
    if(NUM_CHANNELS == 2)
      output_data_buffer[i] = SAMPLE_SILENCE;  /* right */
  }
}
else
{
  for(i=0; i<framesPerBuffer; i++)
  {
    output_data_buffer[i] = *input++;
  }
}
```

Gambar III.17 Kode program operasi pembacaan data dari *buffer* yang dimiliki *sound card*.

```
sprintf(recordFileName,"dev-%d-buf-rec-%d.raw", userDataFile->
    devNum, pubBufFileID);
buf_fid = fopen(recordFileName, "wb");
if(buf_fid == NULL)
{
  printf("Could not open file for saving the buffer.");
  exit(1);
}
else
{
  fwrite(output_data_buffer, NUM_CHANNELS * sizeof(SAMPLE),
      framesPerBuffer, buf_fid);
  fclose(buf_fid);
}
```

Gambar III.18 Kode program operasi penyimpanan data ke file yang berekstensi `raw`.

```
pa = fftw_plan_dft_1d((N << 1) - 1, signala_ext, outa,
    FFTW_FORWARD, FFTW_ESTIMATE);
pb = fftw_plan_dft_1d((N << 1) - 1, signalb_ext, outb,
    FFTW_FORWARD, FFTW_ESTIMATE);
px = fftw_plan_dft_1d((N << 1) - 1, out, out_shifted,
    FFTW_BACKWARD, FFTW_ESTIMATE);

for (i = 0; i < (N << 1) - 1; i++) {
  if (i < N) {
    signala_ext[i] = signala[i];
    signalb_ext[i] = signalb[i];
  }
  else {
    signala_ext[i] = 0;
    signalb_ext[i] = 0;
  }
}

fftw_execute(pa);
fftw_execute(pb);

for (i = 0; i < (N << 1) - 1; i++)
  out[i] = outa[i] * conj(outb[i]);

fftw_execute(px);

for (i = 0; i < (N << 1) - 1; i++)
  result[i] = out_shifted[(i + N) % ((N << 1) - 1)] / ((N <<
      1) - 1);
```

Gambar III.19 Kode program operasi CCC terhadap dua data sinyal.

```
maxTemp1 = minTemp1 = creal(first_xcorr_signal[0]);;
maxTemp2 = minTemp2 = creal(second_xcorr_signal[0]);

for(l = 0; l < FramesPerBuffer; l++) {
  if(creal(first_xcorr_signal[l]) > maxTemp1)
    maxTemp1 = creal(first_xcorr_signal[l]);
  if(creal(first_xcorr_signal[l]) < minTemp1)
    minTemp1 = creal(first_xcorr_signal[l]);
  if(creal(second_xcorr_signal[l]) > maxTemp2)
    maxTemp2 = creal(second_xcorr_signal[l]);
  if(creal(second_xcorr_signal[l]) < minTemp2)
    minTemp2 = creal(second_xcorr_signal[l]);
}

return 20*log10((maxTemp1-minTemp1)/(maxTemp2-minTemp2));
```

Gambar III.20 Kode program operasi perhitungan PtPAR.

Perangkat Keras

Implementasi rancangan penempatan mikrofon dapat dilihat pada Gambar III.21. Mikrofon ditancapkan pada *styrofoam* yang ditempelkan pada perangkat *multitouch* vertikal (Gambar III.27(b)). Selain sebagai tempat untuk meletakkan mikrofon, *styrofoam* juga dapat berfungsi sebagai peredam suara. Oleh karena itu, diharapkan mikrofon hanya menangkap sinyal suara dari arah depan.

3.2.5 Iterasi I: Pengujian dan Evaluasi

Pengambilan data dilakukan dengan memutar secara berulang-ulang file rekaman suara FAN_1A dan MAA_1A yang berisi ucapan kata ”satu”. Masing-masing rekaman menghasilkan sebuah set data yang memuat 78 data dari 13 titik pengambilan data yang telah didefinisikan pada Subbab 3.2.3 (6 data per titik). Gambar III.22 dan Gambar III.23 menunjukkan parameter TDOA dan PtPAR pasangan mikrofon 3 dan 4 dari set data FAN_1A dan MAA_1A. Untuk memudahkan pembacaan grafik, jangkauan nilai koordinat $x = \{-50, 10, 70, 130, 190\}$ dan $y = \{60, 120, 180\}$ diubah menjadi $x = \{-2, -1, 0, 1, 2\}$ dan $y = \{-1, 0, 1\}$.

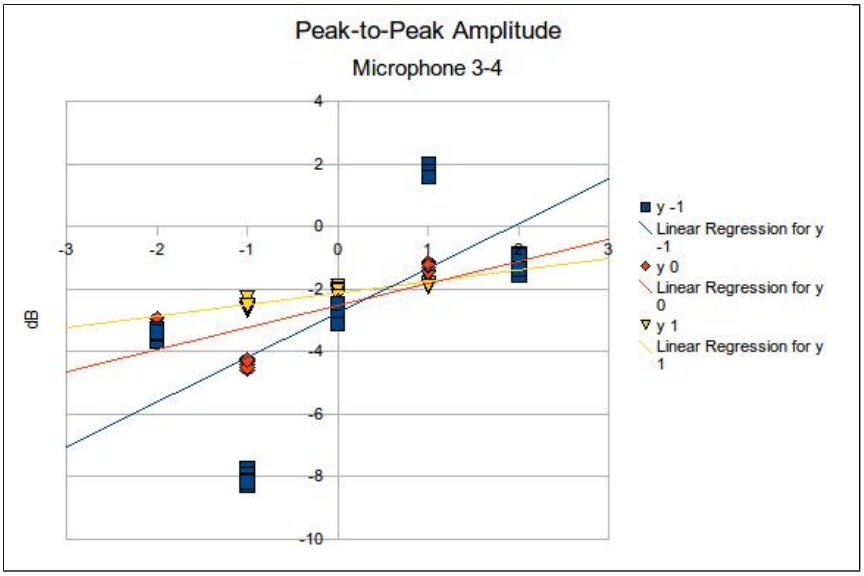
Grafik-grafik dari dua set data tersebut memperlihatkan bahwa nilai parameter PtPAR tidak cukup konsisten dan nilai parameter TDOA sangat tidak konsisten. Dua set data ini tidak dapat digunakan untuk melatih JST yang baik.

Gambar III.21 Foto implementasi rancangan penempatan mikrofon.

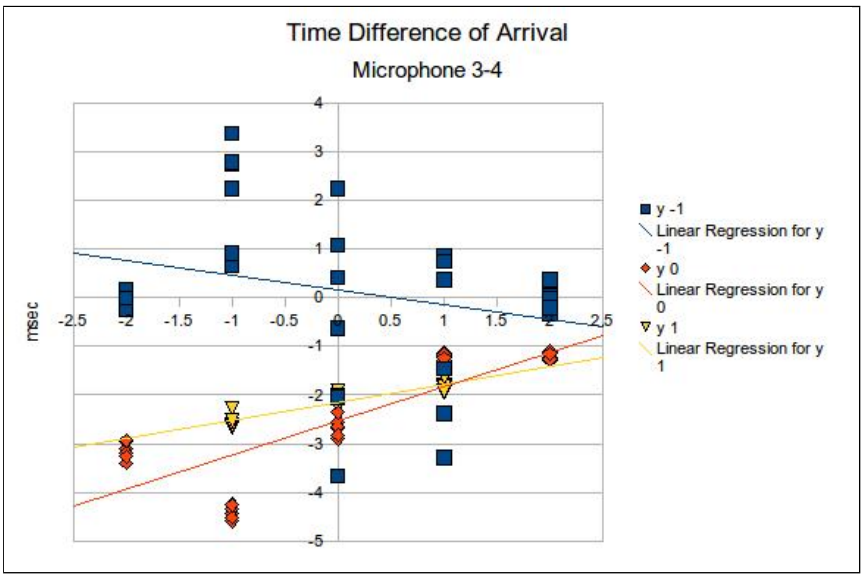
3.2.6 Iterasi II: Desain

Perangkat Keras

Oleh karena prioritas utama adalah menentukan *azimuth* posisi pengguna relatif terhadap layar, penempatan mikrofon diubah menjadi seperti yang terlihat pada Gambar III.24. Dengan perubahan desain ini, pendekatan terhadap masalah TDOA berubah dari seperti yang tergambar pada Gambar III.25(a) menjadi seperti yang tergambar pada Gambar III.25(b) dan Gambar III.25(c). Selain itu, lebih banyak pasangan mikrofon yang dapat digunakan untuk memperkirakan *azimuth* posisi pengguna relatif terhadap layar. Dengan demikian, terdapat enam parameter yang berpotensi sebagai masukan JST, yaitu dua nilai TDOA dari dua pasang mikrofon yang saling berdekatan dan empat nilai PtPAR dari empat pasang mikrofon yang saling berjauhan. Dua nilai PtPAR dari dua pasang mikrofon yang saling berdekatan dapat diabaikan karena perbedaan amplitudo sinyal yang tertangkap oleh sepasang mikrofon yang letaknya berdekatan tidak signifikan.

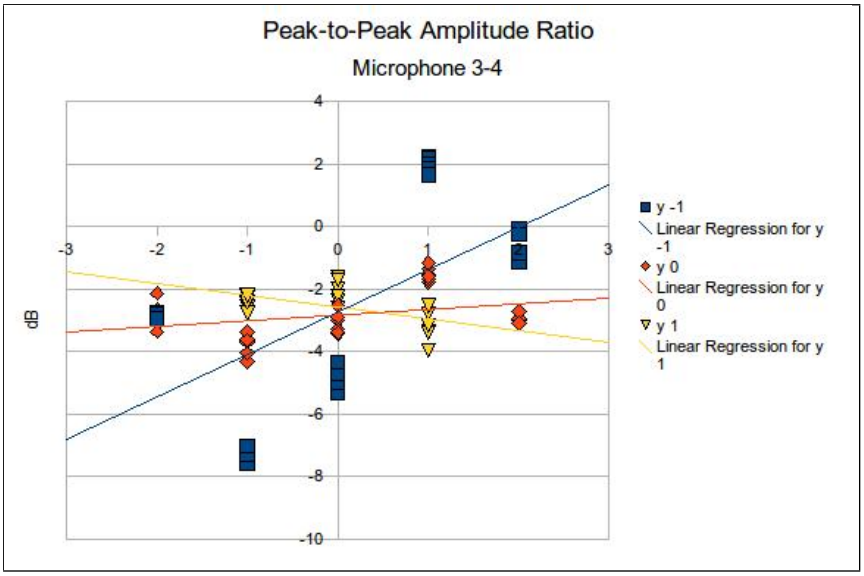


(a) PtPAR

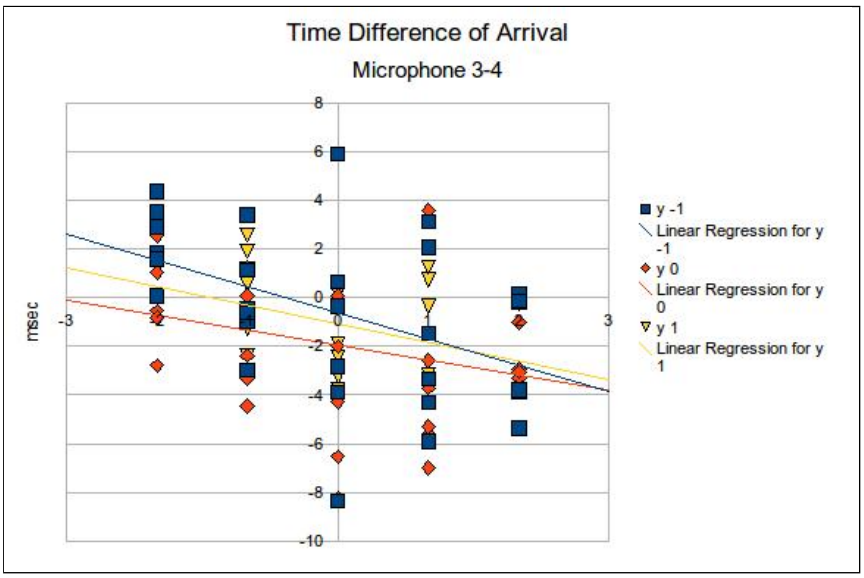


(b) TDOA

Gambar III.22 Grafik TDOA dan PtPAR mikrofon 3-4 dari set data FAN_1A.

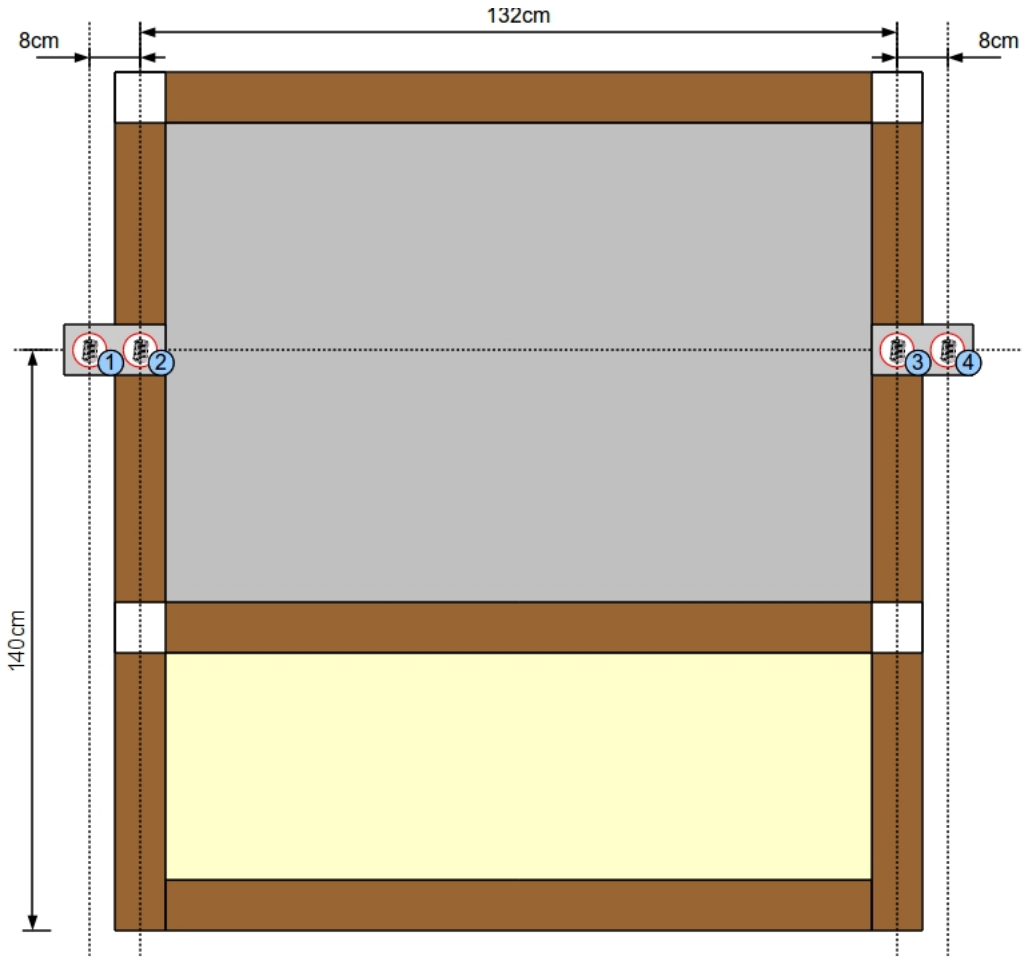


(a) PtPAR

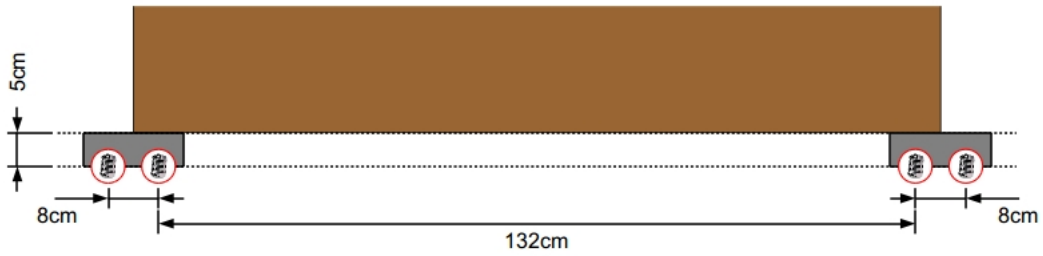


(b) TDOA

Gambar III.23 Grafik TDOA dan PtPAR mikrofon 3-4 dari set data MAA_1A.

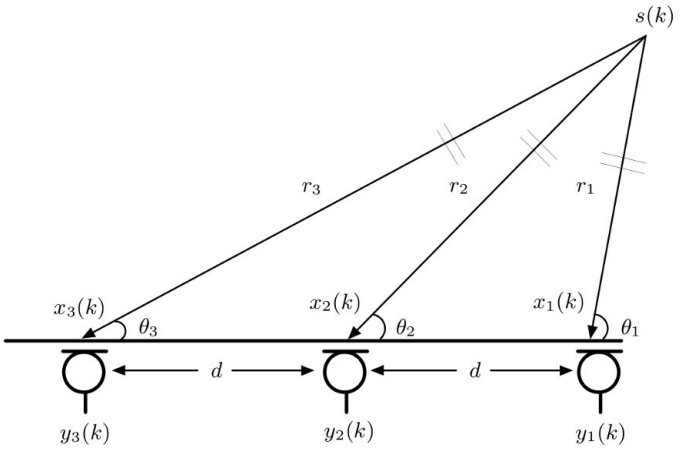


(a) Tampak muka

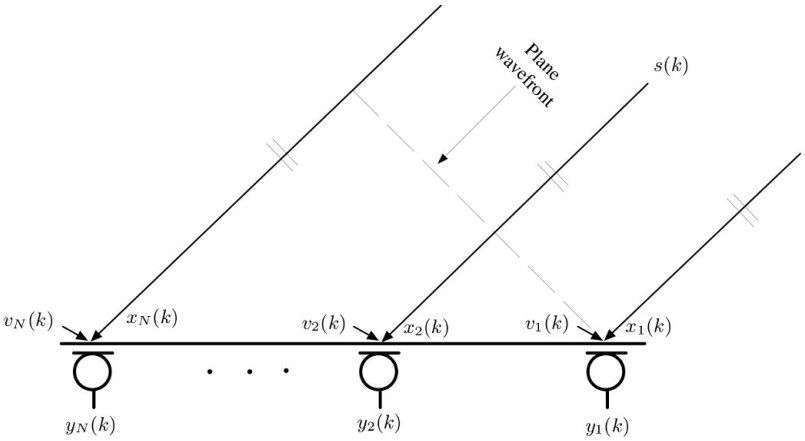


(b) Tampak atas

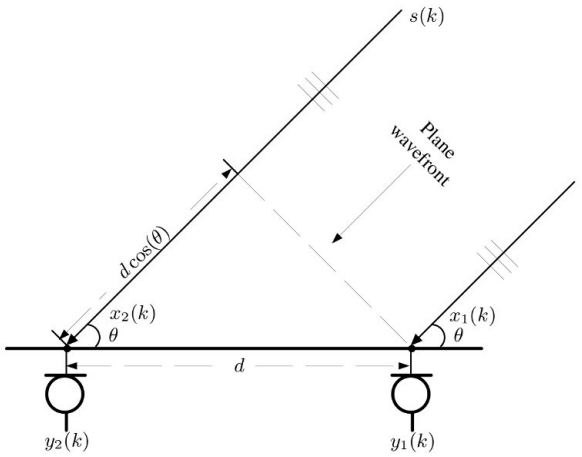
Gambar III.24 Perbaikan rancangan penempatan mikrofon.



(a)



(b)

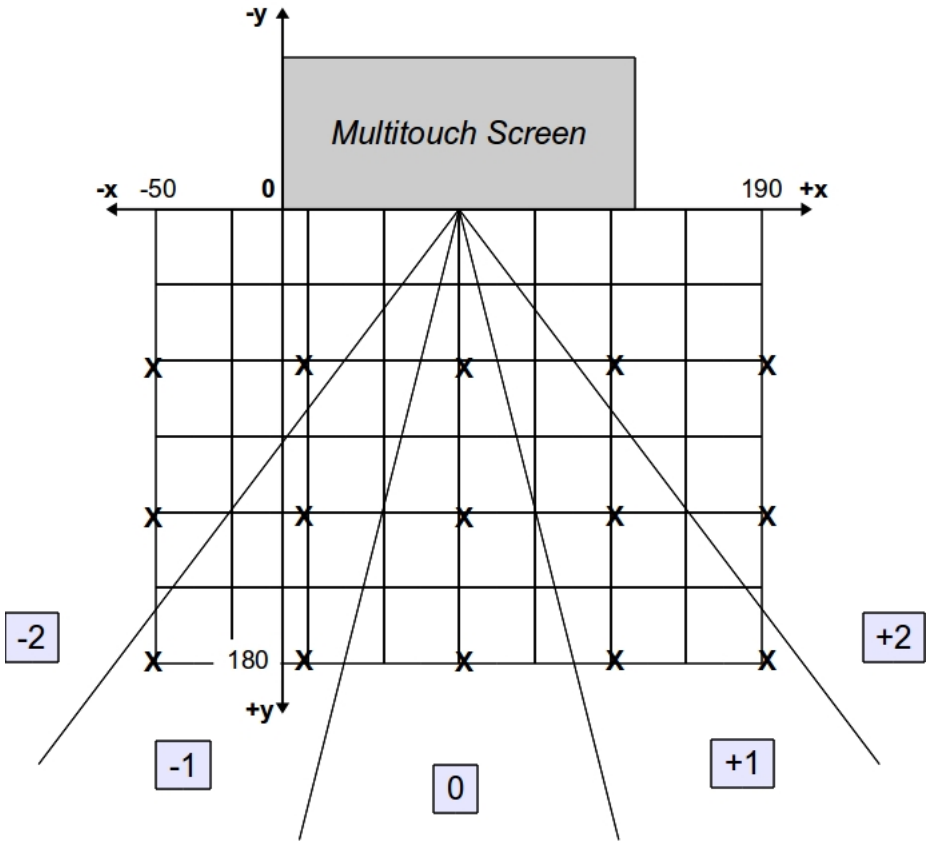


(c)

Gambar III.25 Ilustrasi perubahan pendekatan masalah TDOA [2].

Pengambilan Data

Jangkauan nilai koordinat sumber suara sama dengan yang digunakan sebelumnya (Iterasi I), yaitu $x = \{-50, 10, 70, 130, 190\}$, $y = \{60, 120, 180\}$, dan $z = -30$. Akan tetapi, titik-titik pengambilan data tersebut kemudian dibagi dalam kawasan-kawasan seperti yang ditunjukkan oleh Gambar III.26. Oleh karena itu, apabila sebelumnya kemungkinan keluaran penentuan posisi pengguna adalah sebuah titik dalam koordinat Cartesian tiga dimensi yang relatif terhadap perangkat *multitouch*, dengan penggunaan kawasan kemungkinan keluaran adalah sebuah sudut (*azimuth*) yang relatif terhadap titik tengah perangkat *multitouch*.



Gambar III.26 Perbaikan rancangan titik pengambilan data.

3.2.7 Iterasi II: Implementasi

Perangkat Keras

Implementasi perbaikan rancangan penempatan mikrofon dapat dilihat pada Gambar III.27(a). Gambar III.27(b) menunjukkan bagaimana mikrofon diletakkan dalam *styrofoam* yang ditempelkan pada perangkat *multitouch*.

(a)

(b)

Gambar III.27 Foto implementasi perbaikan rancangan penempatan mikrofon.

3.2.8 Iterasi II: Pengujian dan Evaluasi

Dalam tahap ini, pengambilan data menggunakan empat file rekaman suara, yaitu FAN_9B (ucapan kata ”sembilan”), FAW_7B (ucapan kata ”tujuh”), MAF_25A (ucapan frase ”dua lima”), dan MSD_5B (ucapan kata ”lima”). Untuk memudahkan

pembacaan grafik, jangkauan nilai koordinat $x = \{-50, 10, 70, 130, 190\}$ dan $y = \{60, 120, 180\}$ diubah menjadi $x = \{-4, -2, 0, 2, 4\}$ dan $y = \{-2, -4, -6\}$.

Dari pengamatan terhadap empat set data, nilai parameter PtPAR cukup konsisten, sedangkan nilai parameter TDOA masih saja tidak konsisten. Dengan data tersebut, penentuan posisi pengguna diputuskan hanya akan menggunakan parameter PtPAR saja. Secara lebih spesifik, parameter PtPAR yang akan digunakan adalah parameter PtPAR mikrofon 1-3, 1-4, 2-3, dan 2-4.

Grafik TDOA dan PtPAR yang akan ditampilkan secara lengkap hanya set data FAN_9B yang cukup merepresentasikan set data yang lain (Gambar III.28 sampai dengan Gambar III.33). Gambar III.34 dan Gambar III.35 menunjukkan grafik PtPAR mikrofon 1-3, 1-4, 2-3, dan 2-4 dari set data FAW_7B untuk memberikan gambaran lebih lanjut mengenai data latih dan data uji JST.

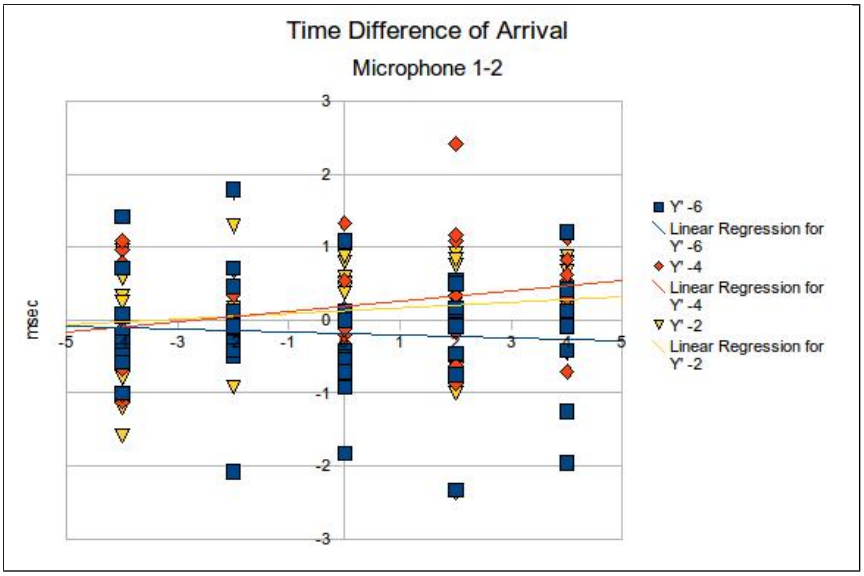
3.2.9 Pelatihan dan Pengujian Jaringan Syaraf Tiruan

JST dilatih dengan menggunakan set data yang diperoleh pada Iterasi II. Data latih dan data uji yang digunakan berasal dari set data FAN_9B dan FAW_7B yang memiliki 240 data latih dan 60 data uji. Selain melakukan pelatihan dengan parameter yang sama secara berulang, parameter jumlah neuron tersembunyi dan *mean squared error* (MSE) pelatihan (*desired error*) juga diubah-ubah untuk memperoleh MSE pelatihan dan pengujian yang baik. Data dari proses pelatihan JST ditampilkan pada Tabel III.1.

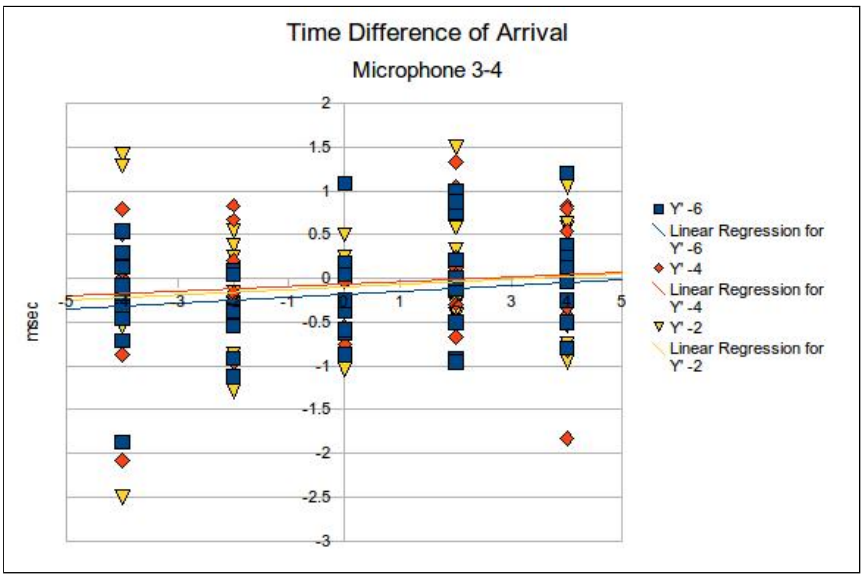
JST yang diperoleh kemudian diuji dengan set data MAF_25A dan MSD_5B yang masing-masing memuat 50 data, serta FAN_9B_2 yang memuat 150 data. Data dari proses pengujian JST ditampilkan pada Tabel III.2. Data pengujian tersebut menunjukkan bahwa MSE rata-rata terendah diperoleh dari penggunaan JST yang dihasilkan pada pelatihan ke-7. JST inilah yang nantinya akan digunakan untuk memperkirakan posisi pengguna setelah subsistem diintegrasikan ke dalam RESTU.

3.2.10 Analisis Hasil

Dari proses perancangan yang telah dilakukan, beberapa pencapaian yang telah diperoleh antara lain sebagai berikut.

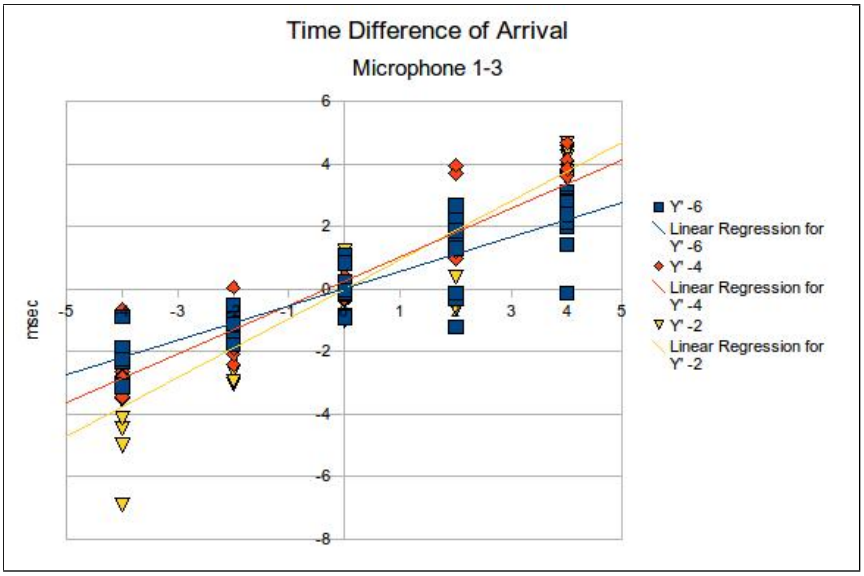


(a) TDOA 1-2

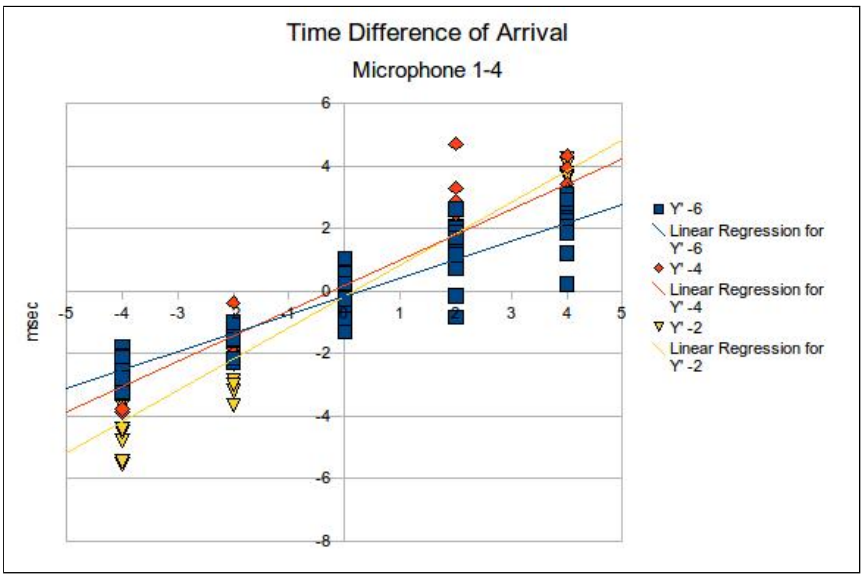


(b) TDOA 3-4

Gambar III.28 Grafik TDOA dari set data FAN_9B (1).

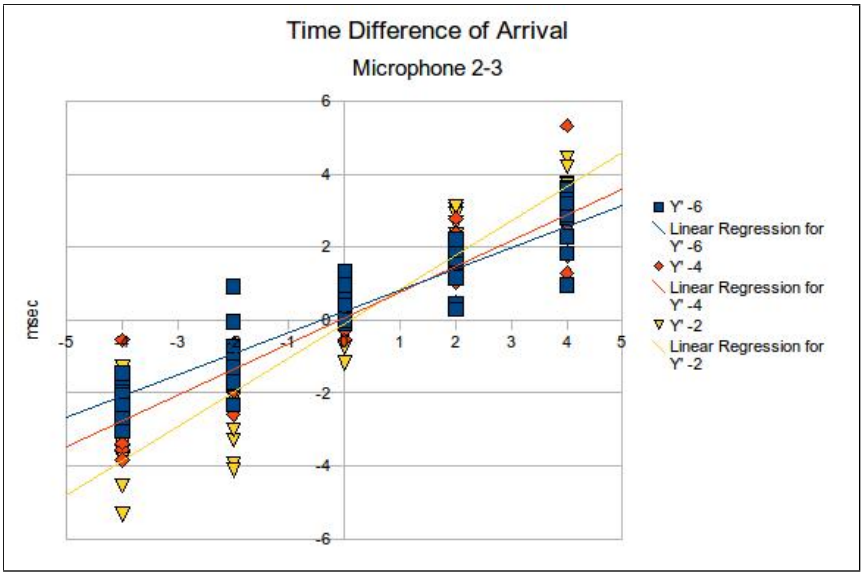


(a) TDOA 1-3

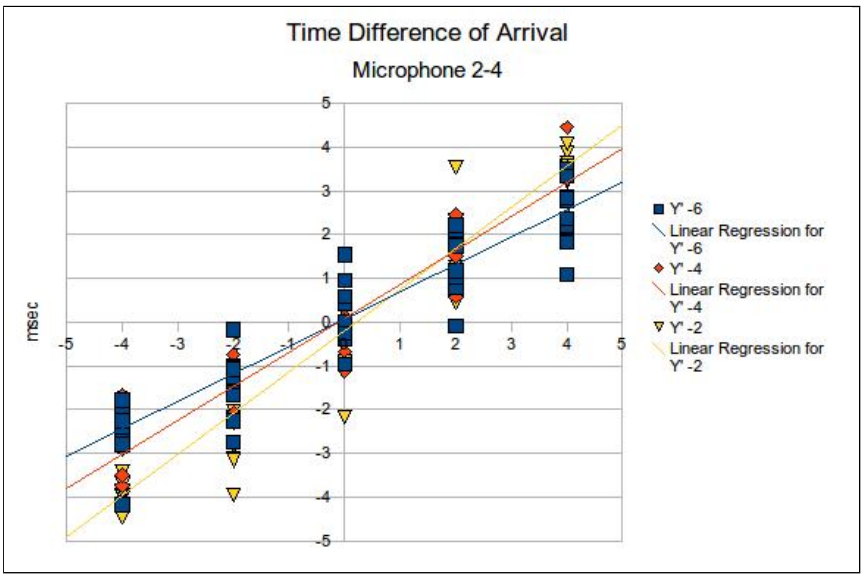


(b) TDOA 1-4

Gambar III.29 Grafik TDOA dari set data FAN_9B (2).

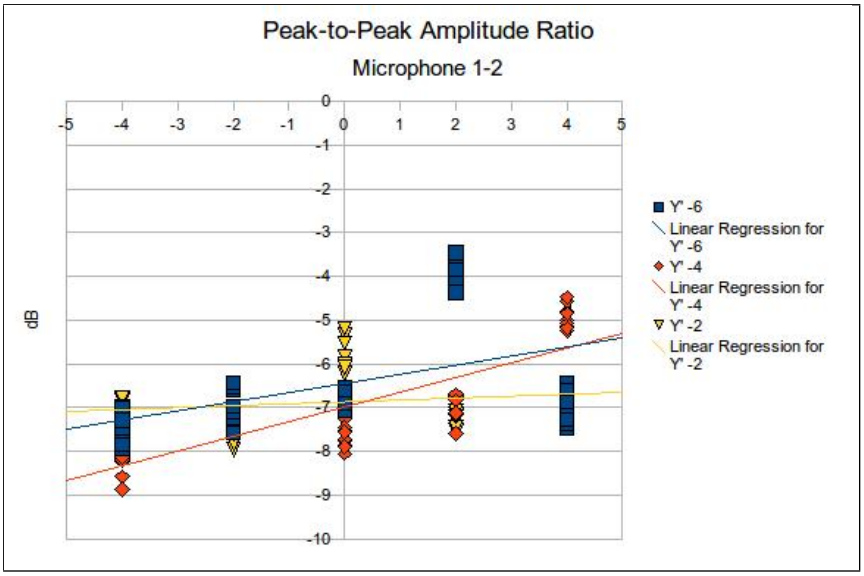


(a) TDOA 2-3

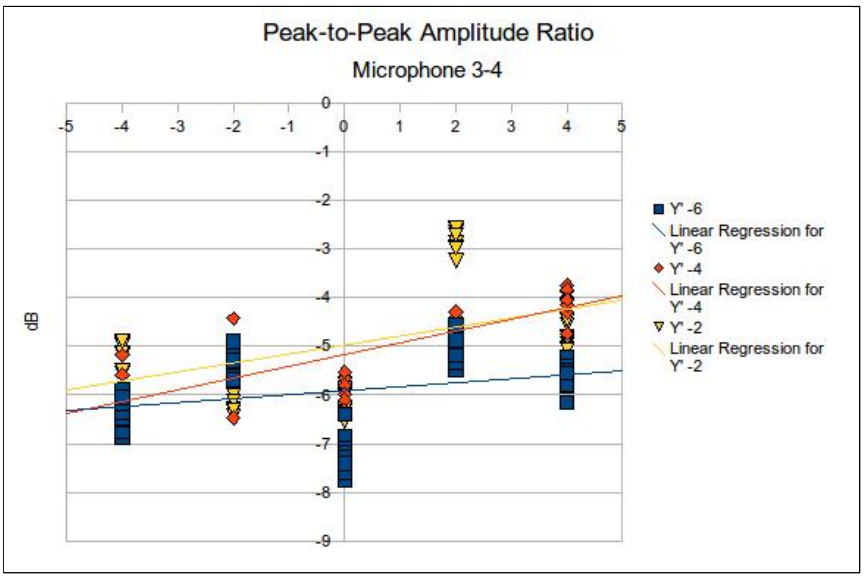


(b) TDOA 2-4

Gambar III.30 Grafik TDOA dari set data FAN_9B (3).

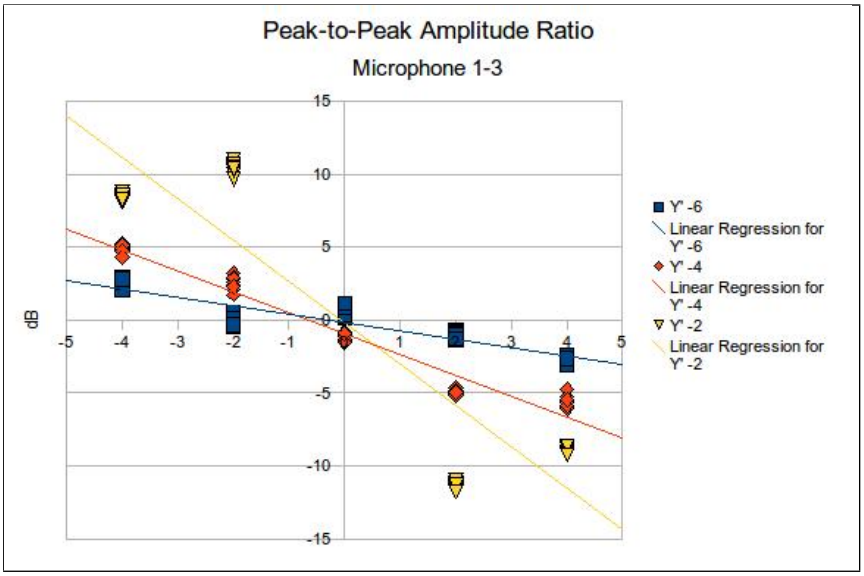


(a) PtPAR 1-2

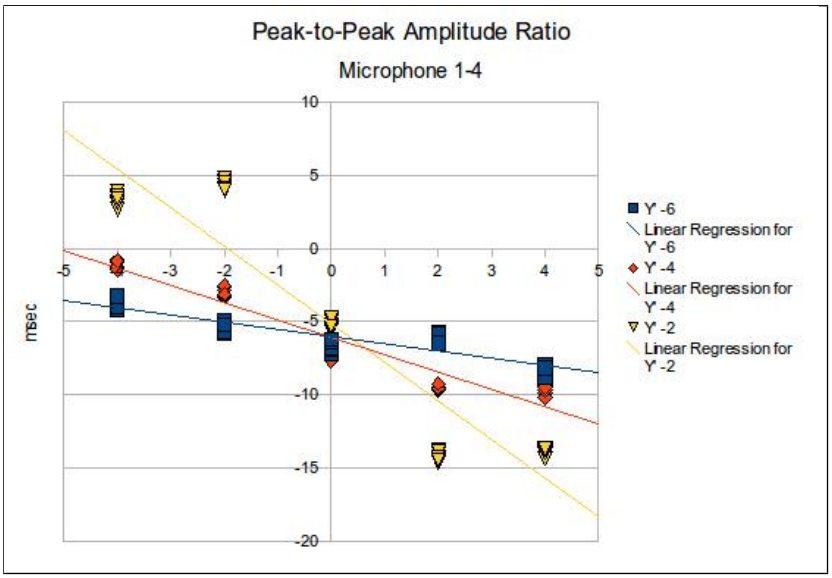


(b) PtPAR 3-4

Gambar III.31 Grafik PtPAR dari set data FAN_9B (1).

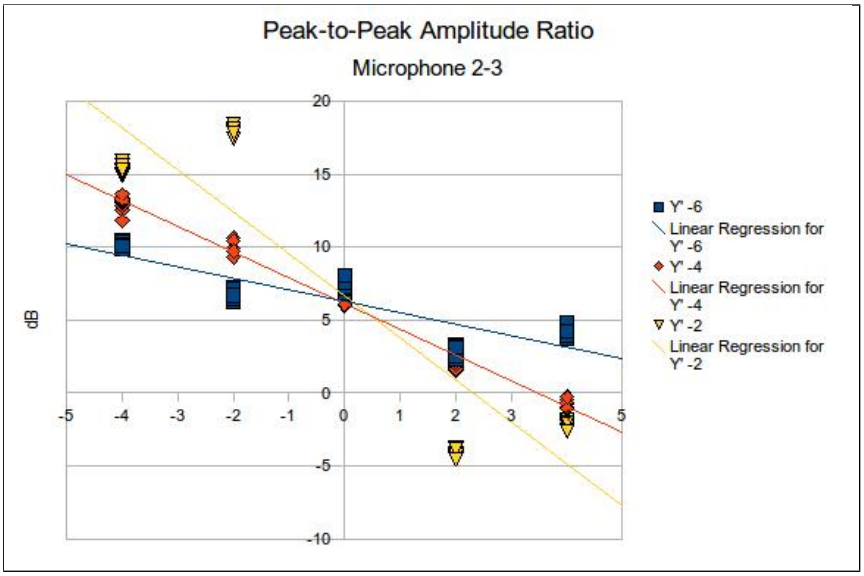


(a) PtPAR 1-3

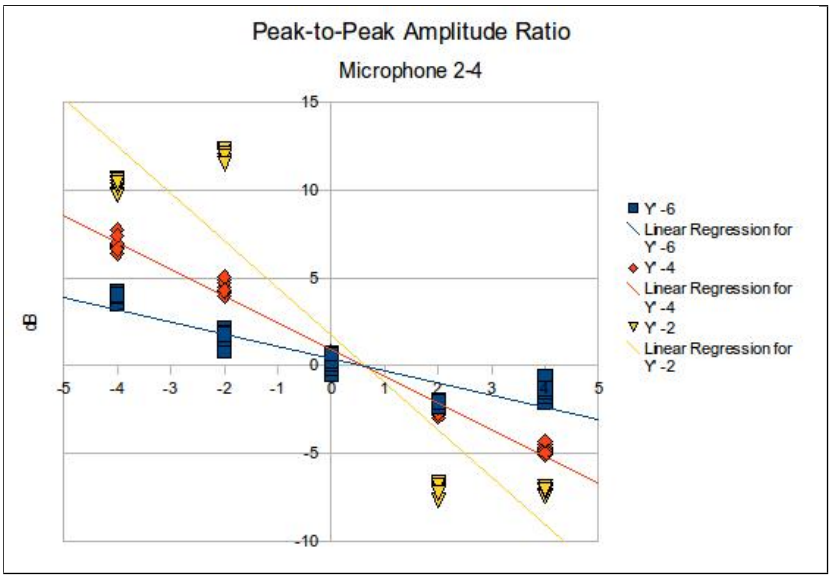


(b) PtPAR 1-4

Gambar III.32 Grafik PtPAR dari set data FAN_9B (2).

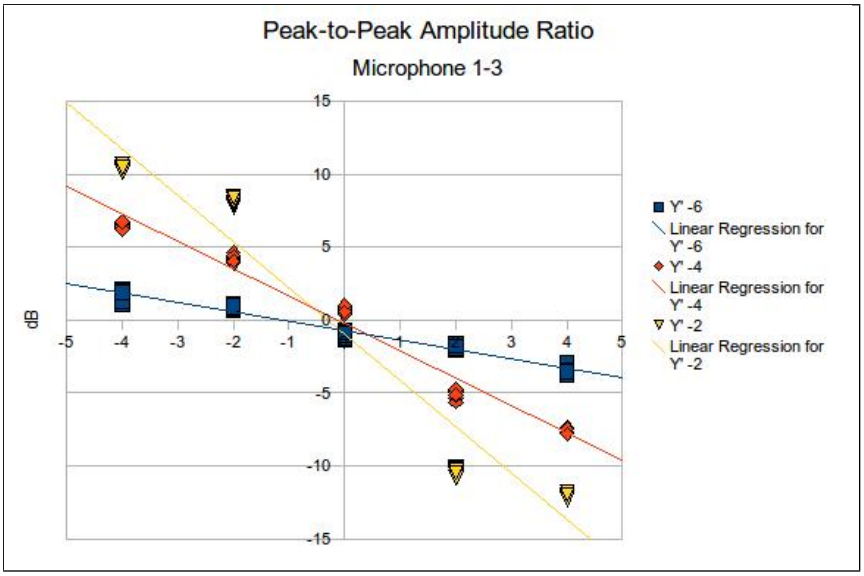


(a) PtPAR 2-3

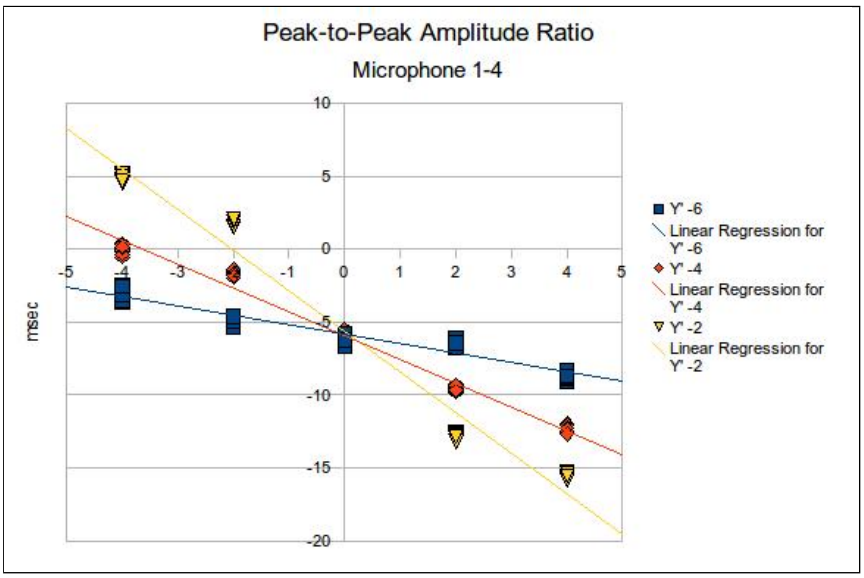


(b) PtPAR 2-4

Gambar III.33 Grafik PtPAR dari set data FAN_9B (3).

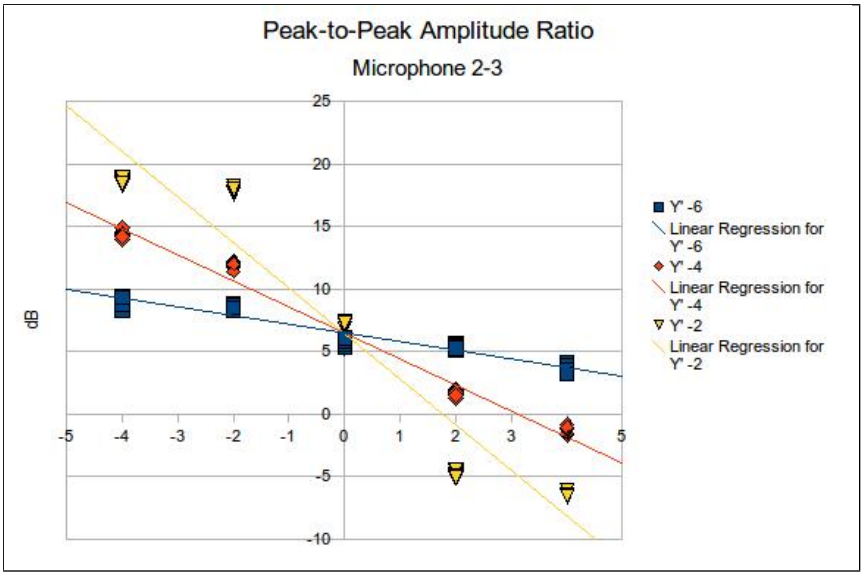


(a) PtPAR 1-3

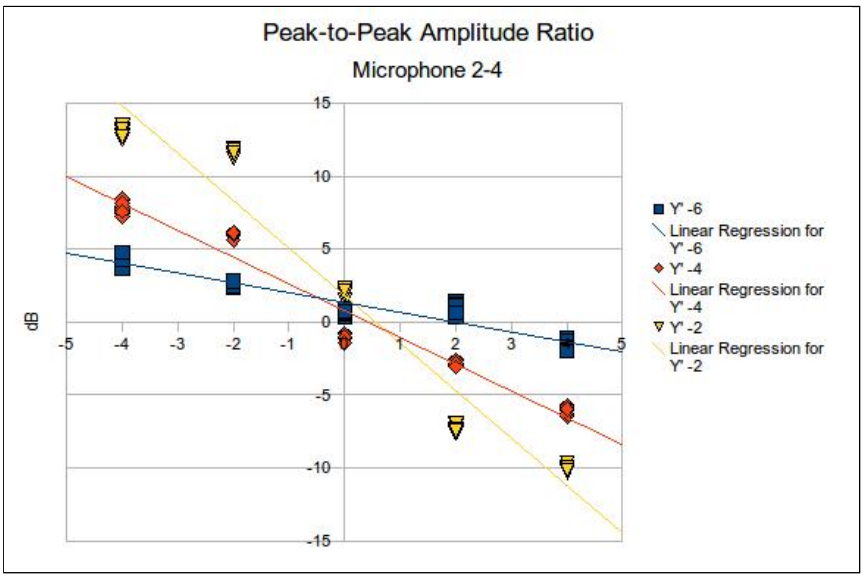


(b) PtPAR 1-4

Gambar III.34 Grafik PtPAR dari set data FAW_7B (1).



(a) PtPAR 2-3



(b) PtPAR 2-4

Gambar III.35 Grafik PtPAR dari set data FAW_7B (2).

Tabel III.1 Data pelatihan JST.

Pelatihan ke-	Jumlah layer	Jumlah neuron tersembunyi	MSE pelatihan	MSE pengujian
1	3	64	0,0009989900	0,010317
2	3	32	0,0009997976	0,004737
3	3	16	0,0009985201	0,000379
4	3	16	0,0007484510	0,002396
5	3	16	0,0004987848	0,004765
6	3	16	0,0002499711	0,008815
7	3	16	0,0001499649	0,005309
8	3	16	0,0000999883	0,027223

Tabel III.2 Data pengujian JST.

JST Hasil Pelatihan ke-	MSE MAF_25A	MSE MSD_5B	MSE FAN_9B_2	MSE Rata-rata
1	0,174747	0,366794	0,309057	0,283533
2	0,191638	0,400196	0,467885	0,353240
3	0,201244	0,372975	0,426066	0,333428
4	0,199302	0,219985	0,417764	0,279017
5	0,176784	0,230723	0,445191	0,284233
6	0,172334	0,368824	0,437893	0,326350
7	0,139543	0,210295	0,464500	0,271446
8	0,181249	0,356440	0,602393	0,380027

1. Menangkap suara pengguna dengan menggunakan *microphone array*, yang terdiri dari empat mikrofon, secara bersamaan memanfaatkan pustaka Portaudio. Representasi sinyal yang ditangkap disimpan dalam file yang berekstensi `raw`.

2. Menghitung TDOA dari sinyal suara yang ditangkap oleh *microphone array* menggunakan metode CCC. Implementasi CCC menggunakan DFT memanfaatkan pustaka FFTW3. Implementasi telah diverifikasi kebenarannya dengan cara membandingkan hasil perhitungan yang diperoleh dari implementasi dengan hasil perhitungan yang diperoleh dari fungsi `xcorr` pada program Octave.

3. Menghitung PtPAR dari sinyal suara yang ditangkap oleh *microphone array*. Implementasi telah diverifikasi kebenarannya dengan cara membandingkan hasil perhitungan yang diperoleh dari implementasi dengan hasil pengamatan grafik sinyal yang ditampilkan program Audacity.

4. Menyimpan data TDOA, PtPAR, dan posisi untuk membuat data latih dan data uji JST.

5. Melatih JST dengan nilai TDOA dan/atau PtPAR yang didapatkan selama pengambilan data latih dengan memanfaatkan pustaka FANN.

6. Menguji JST dengan nilai TDOA dan/atau PtPAR yang didapatkan selama pengambilan data uji dengan memanfaatkan pustaka FANN.

Pada awal penelitian, parameter TDOA dan PtPAR diperkirakan dapat digunakan untuk melakukan penentuan posisi sumber suara, atau dalam hal ini adalah posisi pengguna. Parameter PtPAR terbukti dapat digunakan untuk memperkirakan posisi pengguna, sedangkan parameter TDOA tidak dapat. Meskipun demikian, parameter TDOA sebenarnya merupakan parameter yang banyak digunakan dalam penentuan posisi sumber suara sehingga tingkat keakuratannya seharusnya cukup baik. Oleh karena itu, tidak dapat digunakannya parameter tersebut dalam penelitian ini disebabkan oleh desain dan implementasi yang kurang sesuai dengan batasan masalah TDOA.

Faktor utama yang menyebabkan hasil perhitungan TDOA sangat tidak konsisten adalah faktor perangkat keras yang digunakan. Penggunaan *USB sound card* yang kemudian datanya dikelola oleh program yang memanfaatkan pustaka Portaudio tidak dapat menghasilkan representasi data sinyal yang baik karena batasan waktu tidak ditepati dengan baik. Dari pengamatan, proses perekaman yang dilakukan oleh empat *sound card* tidak berjalan secara bersamaan. Pengamatan dilakukan dengan memberikan penanda waktu (*timestamp*) saat proses merekam mulai dan berakhir. Dari penanda waktu tersebut diketahui bahwa durasi perekaman tidak pernah terpenuhi dan durasi *sound card* yang satu dengan yang lain dalam mengambil data tidak sama , meskipun semua *sound card* diset untuk merekam dalam durasi yang sama. Sebagai contoh, apabila semua *sound card* diset untuk merekam selama 1 detik, penanda waktu menunjukkan bahwa *sound card* merekam dalam durasi $1 + \delta t$, dengan δt bervariasi antara satu *sound card* dengan *sound card* yang lain. Apabila δt dari satu proses perekaman ke proses yang lain sama (konsisten), permasalahan mungkin dapat dipecahkan dengan memberikan *offset* pada data. Dari pengamatan δt dari satu proses perekaman ke proses yang lain tidak konsisten. meskipun nilai δt hanya dalam orde milidetik, pada dasarnya file representasi sinyal suara yang diperoleh tidak dapat digunakan untuk menghitung TDOA. Apabila sinyal tersebut digunakan untuk menghitung TDOA, hasil perhitungan merupakan nilai yang tidak valid karena nilai

tersebut sangat dipengaruhi oleh δt sehingga tidak merepresentasikan perbedaan waktu yang dibutuhkan propagasi sinyal dari sumber suara ke mikrofon dengan baik.

Untuk memecahkan permasalahan tersebut dibutuhkan perangkat akuisisi sinyal yang mampu berjalan secara *real-time*, seperti yang digunakan pada [22], [29], dan [32]. Pada prinsipnya, apabila *time-constraint* yang diberikan dapat ditepati dengan baik, dengan asumsi model *free-field* ideal, representasi sinyal akan memenuhi Persamaan II.1.

BAB IV

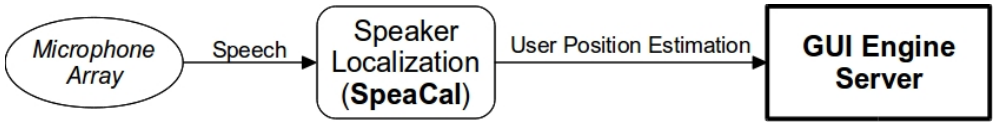
INTEGRASI SPEACAL DALAM RESTU

4.1 Iterasi III: Desain

Perangkat Lunak

Subsistem penentuan posisi pengguna berdasarkan sinyal suara yang telah didesain pada Bab III harus diintegrasikan dengan RESTU. Pada dasarnya, proses integrasi hanya mencakup proses pengiriman data hasil penentuan posisi yang dihasilkan oleh SpeaCal ke salah satu bagian dari RESTU, misalnya *AI Engine* atau *GUI Engine* (Gambar III.3).

Pada konsep *artificial general intelligence* (AGI), semua informasi yang diterima oleh agen, termasuk informasi posisi pengguna, seharusnya masuk ke *AI Engine* terlebih dahulu. Oleh karena bagian kognitif dari RESTU belum menggunakan konsep AGI ini, untuk sementara informasi yang diterima oleh agen diteruskan ke bagian yang langsung membutuhkannya. Dengan demikian, informasi dari subsistem penentuan posisi pengguna berdasarkan sinyal suara dan subsistem penentuan posisi pengguna berdasarkan citra langsung dikirimkan ke *GUI Engine*, tanpa melalui *AI Engine* (Gambar IV.1). Oleh karena itu, desain pengiriman data dari subsistem penentuan posisi pengguna berdasarkan sinyal suara perlu memperhatikan desain masukan data yang digunakan oleh *GUI Engine*.



Gambar IV.1 Diagram aliran data dari SpeaCal ke *GUI Engine*.

GUI Engine akan membuka sebuah *socket* untuk pengiriman data dari subsistem penentuan posisi pengguna berdasarkan sinyal suara dan subsistem penentuan posisi pengguna berdasarkan citra. Format penulisan informasi posisi dari dua subsistem

tersebut dibedakan agar *GUI Engine* mengetahui darimana informasi berasal. Informasi posisi kemudian dituliskan dalam format koordinat Cartesian tiga dimensi dengan urutan penulisan: koordinat z, y, kemudian x. (Gambar IV.2).

```
1|a|-30|120|10|
```

Gambar IV.2 Contoh penulisan informasi posisi yang dikirim dari SpeaCal ke *GUI Engine Server*.

Daerah cakupan subsistem penentuan posisi pengguna berdasarkan citra relatif sempit. Apabila mengacu pada pembagian kawasan yang digunakan oleh subsistem penentuan posisi pengguna berdasarkan sinyal suara (Gambar III.26), subsistem penentuan posisi pengguna berdasarkan citra hanya mencakup kawasan 0 saja.

Prioritas penggunaan informasi posisi pengguna yang berdasarkan citra lebih tinggi daripada informasi yang berdasarkan sinyal suara. Hal ini dikarenakan penentuan posisi pengguna berdasarkan citra menghasilkan posisi yang lebih akurat daripada penentuan posisi berdasarkan sinyal suara. Apabila subsistem penentuan posisi pengguna berdasarkan citra menangkap posisi pengguna melalui dua *webcam* yang digunakannya, *GUI Engine* akan menggunakan informasi posisi dari subsistem tersebut. Apabila tidak ada informasi dari subsistem penentuan posisi berdasarkan citra, *GUI Engine* akan menggunakan informasi dari subsistem penentuan posisi berdasarkan sinyal suara.

Untuk meningkatkan akurasi dalam penentuan posisi pengguna, SpeaCal melakukan tiga kali proses penentuan posisi dengan JST. Hasil dari tiga proses tersebut kemudian diurutkan untuk mencari nilai mediannya. Dengan demikian, apabila sampel sinyal suara yang digunakan untuk menentukan posisi diset $\frac{1}{2}$ detik, informasi posisi pengguna akan diperbarui setiap 3 detik.

Informasi posisi yang diterima oleh *GUI Engine* menggunakan format koordinat Cartesian tiga dimensi. Oleh karena itu, nilai median yang diperoleh sebelumnya perlu dikonversi terlebih dahulu dari format kawasan ke format koordinat Cartesian tiga dimensi. Dalam proses konversi koordinat yang dipilih untuk merepresentasikan kawasan adalah koordinat yang terletak pada $y = 120$. Dengan kata lain, $posisi_{kawasan} = -2, -1, 0, 1, 2$ dikonversi menjadi $posisi_{Cartesian} = (x, 120, -30)$, dengan $x = -50, 10, 70, 130, 190$. Koordinat Cartesian inilah yang kemudian dikirimkan ke *GUI Engine* dengan format seperti yang tercantum pada Gambar IV.2.

4.2 Iterasi III: Implementasi

Perangkat Lunak

Implementasi konversi koordinat dari format kawasan ke format koordinat Cartesian tiga dimensi tertulis pada Gambar IV.3. Sedangkan, implementasi penyusunan data yang akan dikirimkan ke *GUI Engine* sesuai format penulisan yang telah didefinisikan dan pengiriman data melalui *socket* tertulis pada Gambar IV.4. Dua operasi terkait *socket* tersebut diimplementasikan memanfaatkan pustaka Boost.

```
1 speakerPosValue[0] = annResult * 60 + 70;
2 speakerPosValue[1] = 120;
3 speakerPosValue[2] = -30;
```

Gambar IV.3 Kode program operasi konversi koordinat.

4.3 Iterasi III: Pengujian dan Evaluasi

Pengujian menunjukkan bahwa data dari SpeaCal dapat diterima dengan baik oleh *GUI Engine*. Pada saat pengujian, koneksi beberapa kali terputus karena sinyal WLAN yang lemah. Selain itu, tidak ada permasalahan yang dihadapi oleh komunikasi data antara SpeaCal dan *GUI Engine*.

Setelah SpeaCal berhasil diintegrasikan dalam RESTU, pengujian pengguna secara terbatas juga dilakukan. Pengujian melibatkan pihak-pihak yang juga terlibat dalam pengembangan RESTU. Dari proses pengujian, diketahui bahwa SpeaCal dapat menghasilkan informasi posisi pengguna berdasarkan suara pengguna. Meskipun demikian, informasi posisi yang dihasilkan seringkali masih tidak akurat. Dari hasil pengamatan, hasil SpeaCal cenderung berkisar pada nilai -1, 0, dan 1, meskipun pada faktanya pengguna berada pada kawasan -2 atau 2. SpeaCal akan menghasilkan nilai -2 atau 2 apabila jarak pengguna relatif sangat dekat dengan salah satu pasangan mikrofon.

Pengujian menunjukkan bahwa SpeaCal menghasilkan perkiraan posisi yang lebih akurat apabila pengguna memperbesar amplitudo suaranya dengan sedikit berteriak.

```
try
{
  boost::asio::io_service io_service;
  boost::asio::ip::tcp::resolver resolver(io_service);
  boost::asio::ip::tcp::resolver::query query(boost::asio::ip
      ::tcp::v4(), "167.205.56.139", "12137");
  boost::asio::ip::tcp::resolver::iterator endpoint_iterator =
       resolver.resolve(query);
  boost::asio::ip::tcp::resolver::iterator end;
  boost::asio::ip::tcp::socket socket(io_service);
  boost::system::error_code error = boost::asio::error::
      host_not_found;
  while (error && endpoint_iterator != end)
  {
    socket.close();
    socket.connect(*endpoint_iterator++, error);
  }
  if (error)
    throw boost::system::system_error(error);

  std::string pos_info;
  std::string pos_data_start ("a");
  std::string pos_data_separator ("|");

  pos_info.clear();
  pos_info = pos_data_start;
  pos_info += pos_data_separator;
  pos_info += boost::lexical_cast<std::string>(speakerPosValue
      [2]);
  pos_info += pos_data_separator;
  pos_info += boost::lexical_cast<std::string>(speakerPosValue
      [1]);
  pos_info += pos_data_separator;
  pos_info += boost::lexical_cast<std::string>(speakerPosValue
      [0]);
  pos_info += pos_data_separator;

  try
  {
    boost::asio::write(socket, boost::asio::buffer( pos_info )
        );
    std::cout << pos_info << std::endl;
  }
  catch( std::exception e )
    throw std::runtime_error("message send error | " + std::
        string( e.what() ) );
}
catch (std::exception& e)
  std::cerr << e.what() << std::endl;
```

Gambar IV.4 Kode program operasi pengiriman data dari SpeaCal ke *GUI Engine* melalui *socket*.

Selain itu, SpeaCal sensitif terhadap suara yang bersifat *spike*, misalnya suara tepukan tangan dan suara jentikan jari.

Pengujian juga menunjukkan bahwa SpeaCal sangat sensitif terhadap derau. Pada awal pengujian, sebuah *server* diletakkan di samping perangkat *multitouch*, di bawah mikrofon 1-2. Dalam kondisi ini, SpeaCal memiliki kecenderungan menunjukkan nilai -1, bahkan saat tidak ada suara yang dominan di ruangan. Ketika posisi *server* sedikit dimundurkan, kecenderungan menunjukkan nilai -1 berkurang dan SpeaCal kembali menunjukkan nilai 0 saat tidak ada suara pengguna.

4.4 Analisis Hasil

Dari proses perancangan lanjutan yang telah dilakukan, beberapa tambahan pencapaian yang telah diperoleh antara lain sebagai berikut.

1. Menggunakan JST untuk menghasilkan informasi perkiraan posisi pengguna (sumber suara) secara *real-time*, meskipun hasilnya seringkali masih tidak akurat.
2. Mengirimkan informasi perkiraan posisi pengguna ke bagian *GUI Engine* dari RESTU yang kemudian dimanfaatkan untuk menentukan ke arah mana agen virtual menengokkan kepalanya.

Hasil penentuan posisi yang seringkali tidak akurat mungkin disebabkan oleh sensitivitas mikrofon yang kurang memadai. Hal ini ditandai oleh penentuan posisi sering menunjukkan hasil yang akurat apabila penguji (pengguna) mengeluarkan suara yang beramplitudo yang lebih besar dibandingkan dengan saat berbicara normal.

Kurang akuratnya penentuan posisi berdasarkan sinyal suara juga dipengaruhi oleh derau suara yang ada saat penggunaan. Saat sebuah komputer salah satu *server* RESTU diletakkan di bawah mikrofon 1-2, hasil penentuan posisi cenderung menunjukkan nilai -1. Bahkan, dalam kondisi tidak ada pengguna yang bersuara, SpeaCal menunjukkan nilai -1. Hal ini menunjukkan bahwa suara kipas dari komputer tersebut menjadi derau bagi SpeaCal. Dengan demikian, mengganti mikrofon dengan yang lebih baik pun tidak cukup untuk meningkatkan akurasi SpeaCal karena dengan mikrofon yang lebih sensitif berarti semakin banyak pula derau yang tertangkap.

Alternatif solusi yang lain adalah menambahkan operasi pra-pemrosesan sinyal. Sebagai contoh, penggunaan *filter* dimungkinkan untuk memperkecil pengaruh derau yang potensial ada di lingkungan penggunaan RESTU, misalnya derau kipas komputer. Sebenarnya SpeaCal telah mengimplementasikan *biquad band-pass filter* untuk jangkauan frekuensi 300-3300 Hz yang merupakan jangkauan frekuensi suara manusia. Akan tetapi, tampaknya penggunaan *filter* tersebut saja tidak cukup untuk menghasilkan penentuan posisi pengguna yang baik.

Salah satu batasan masalah dalam penelitian ini adalah hanya ada satu sumber suara (pengguna). Di luar batasan tersebut, potensi derau yang terbesar dari SpeaCal adalah suara manusia yang statusnya bukan pengguna. Alternatif solusi dari permasalahan ini adalah pengembangan subsistem penentuan posisi pengguna berdasarkan audiovisual yang mengintegrasikan proses penentuan posisi berdasarkan sinyal suara dan citra, tidak hanya mengombinasikan hasil dari dua proses penentuan posisi berdasarkan sinyal suara dan citra yang terpisah.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

SpeaCal yang merupakan subsistem penentuan posisi pengguna berdasarkan sinyal suara untuk RESTU telah berhasil dirancang, diimplementasikan, diuji, dan diintegrasikan dalam RESTU. SpeaCal telah memenuhi kebutuhan subsistem ini, yaitu memberikan informasi perkiraan *azimuth* posisi pengguna relatif terhadap layar.

SpeaCal telah memenuhi spesifikasi yang sebelumnya didefinisikan, dengan detail sebagai berikut.

1. SpeaCal mampu menangkap suara pengguna dengan menggunakan *microphone array*, yang terdiri dari empat mikrofon, secara bersamaan memanfaatkan pustaka Portaudio. Representasi sinyal yang ditangkap disimpan dalam file yang berekstensi `raw`.
2. SpeaCal mampu menghitung TDOA dari sinyal suara yang ditangkap oleh *microphone array* menggunakan metode CCC. Implementasi CCC menggunakan DFT memanfaatkan pustaka FFTW3. Implementasi telah diverifikasi kebenarannya dengan cara membandingkan hasil perhitungan yang diperoleh dari implementasi dengan hasil perhitungan yang diperoleh dari fungsi `xcorr` pada program Octave.
3. SpeaCal mampu menghitung PtPAR dari sinyal suara yang ditangkap oleh *microphone array*. Implementasi telah diverifikasi kebenarannya dengan cara membandingkan hasil perhitungan yang diperoleh dari implementasi dengan hasil pengamatan grafik sinyal yang ditampilkan program Audacity.
4. SpeaCal mampu menyimpan data TDOA, PtPAR, dan posisi untuk membuat data latih dan data uji JST.
5. SpeaCal mampu melatih JST dengan nilai TDOA dan/atau PtPAR yang didapatkan selama pengambilan data latih dengan memanfaatkan pustaka FANN.

6. SpeaCal mampu menguji JST dengan nilai TDOA dan/atau PtPAR yang didapatkan selama pengambilan data uji dengan memanfaatkan pustaka FANN.
7. SpeaCal mampu menggunakan JST untuk menghasilkan informasi perkiraan posisi pengguna (sumber suara) secara riil, meskipun hasilnya seringkali masih tidak akurat.
8. SpeaCal mampu mengirimkan informasi perkiraan posisi pengguna ke bagian *GUI Engine* dari RESTU yang kemudian dimanfaatkan untuk menentukan ke arah mana agen virtual menengokkan kepalanya.

Pada awal penelitian, parameter TDOA dan PtPAR diperkirakan dapat digunakan untuk melakukan penentuan posisi sumber suara, atau dalam hal ini adalah posisi pengguna. Parameter PtPAR terbukti dapat digunakan untuk memperkirakan posisi pengguna, sedangkan parameter TDOA tidak dapat digunakan karena syarat *time-constraint* tidak dapat terpenuhi oleh desain dan implementasi yang digunakan. Oleh karena itu, SpeaCal hanya menggunakan PtPAR untuk memperkirakan posisi pengguna.

JST terbaik yang diperoleh dalam proses pelatihan memiliki tiga lapisan (terdiri dari satu masukan, satu tersembunyi, dan satu keluaran) dengan 16 neuron tersembunyi. MSE pelatihan dan pengujian JST tersebut mencapai 0,0001499649 dan 0,005309. Sedangkan, pengujian dengan tiga set data lain terhadap JST tersebut menghasilkan MSE 0,139543; 0,210295; dan 0,464500.

5.2 Saran

Beberapa alternatif pengembangan subsistem penentuan posisi pengguna berdasarkan sinyal pengguna yang mungkin dilakukan adalah sebagai berikut.

- Penggunaan perangkat keras akuisisi sinyal yang mampu berjalan secara *real-time*, misalnya sistem *embedded*.
- Penambahan operasi pra-pemrosesan sinyal, misalnya penapisan derau.
- Pengembangan ke arah penentuan posisi pengguna berdasarkan audiovisual.
- Penggunaan model propagasi sinyal suara *reverberant*.

DAFTAR PUSTAKA

[1] Baken, R. J. dan Orlikoff, R. F., *Clinical Measurement of Speech & Voice*, edisi ke-2, Singular, 1999.

[2] Benesty, J., Chen, J., dan Huang, Y., Direction-of-Arrival and Time-Difference-of-Arrival Estimation, *Microphone Array Signal Processing*, 181–215, Springer, 2008.

[3] Beun, R., Vos, E. D., dan Witteman, C., Embodied Conversational Agents: Effects on Memory Performance and Anthropomorphisation, *Intelligent Virtual Agents*, 315–319, Springer, 2003.

[4] Birchfield, S. dan Gangishetty, R., Acoustic localization by interaural level difference, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, **4**, IEEE, 2005.

[5] Cassell, J., Embodied conversational interface agents, *Communications of the ACM*, **43**(4), 70–78, 2000.

[6] Cassell, J., Bickmore, T., Vilhjálmsson, H., dan Yan, H., More than just a pretty face: affordances of embodiment, *Proceedings of the 5th International Conference on Intelligent User Interfaces*, 52–59, ACM, 2000.

[7] Cassell, J., et al., An architecture for embodied conversational characters, *Proceedings of the First Workshop on Embodied Conversational Characters*, 109–120, 1998.

[8] Cassell, J., et al., Embodiment in conversational interfaces: Rea, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 520–527, 1999.

[9] Chen, J., Huang, Y. A., dan Benesty, J., Time Delay Estimation, *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Kluwer Academic Publishers, 2004.

[10] Cole, R. A., Mariani, J., Uszkoreit, H., Zaenen, A., dan Zue, V. (editor), *Survey of State of the Art in Human Language Technology*, Cambrige University Press, 1997, URL `http://cslu.cse.ogi.edu/HLTsurvey/`.

[11] Forouzan, B. A., *Data Communications and Networking*, edisi ke-3, McGraw-Hill, 2003.

[12] Foster, M., Enhancing human-computer interaction with embodied conversational agents, *Universal Access in Human-Computer Interaction. Ambient Interaction*, 828–837, 2007.

[13] Gasperis, G. D., Building an AIML Chatter Bot Knowledge-Base Starting from a FAQ and a Glossary, *Journal of e-Learning and Knowledge Society*, **6**(2), 75–83, 2010.

[14] Gelfand, S. A., *Hearing: An Introduction to Psychological and Physiological Acoustics*, edisi ke-5, Informa Healthcare, 2010.

[15] Goldstein, E. B., *Sensation and Perception*, edisi ke-8, Wadsworth, 2010.

[16] Halliday, D., Resnick, R., dan Walker, J., *Fundamentals of Physics*, edisi ke-7, Wiley, 2004.

[17] Hartmann, B., Mancini, M., dan Pelachaud, C., Implementing expressive gesture synthesis for embodied conversational agents, *Gesture Workshop*, 188–199, Springer, 2005.

[18] Huang, X., Acero, A., dan Hon, H.-W., *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*, edisi ke-1, Prentice Hall, 2001.

[19] Huang, Y. A., Benesty, J., dan Chen, J., Time Delay Estimation and Source Localization, *Springer Handbook of Speech Processing*, Benesty, J., Sondhi, M. M., dan Huang, Y. A., editor, 1043–1063, Springer, 2008.

[20] Jurafsky, D. dan Martin, J. H., *Speech and Language Processing*, edisi ke-2, Prentice-Hall, 2009.

[21] Kollmeier, B., Brand, T., dan Meyer, B., Perception of Speech and Sound, *Springer Handbook of Speech Processing*, Benesty, J., Sondhi, M. M., dan Huang, Y. A., editor, 61–82, Springer, 2008.

[22] Lee, J., Ji, S., Hahn, M., dan Cho, Y., Real-Time Sound Localization Using Time Difference for Human-Robot Interaction, *Proceedings of 16th IFAC World Congress*, 2005.

[23] Lee, S., Badler, J., dan Badler, N., Eyes alive, *ACM Transactions on Graphics (TOG)*, **21**, 637–644, ACM, Juli 2002.

[24] Loebner, H. G., *Home Page of The Loebner Prize in Artificial Intelligence*, URL `http://www.loebner.net/Prizef/loebner-prize.html`, diakses pada 10 Mei 2011, 02:00 WIB.

[25] Martin, A. F. dan Przybocki, M. A., Speaker recognition in a multi-speaker environment, *EUROSPEECH-2001*, 787-790, 2001.

[26] Massaro, D., Liu, Y., Chen, T., dan Perfetti, C., A multilingual embodied conversational agent for tutoring speech and language learning, *Proceedings of the Ninth International Conference on Spoken Language Processing (Interspeech 2006-ICSLP)*, 825–828, 2006.

[27] McTear, M. F., Spoken dialogue technology: enabling the conversational user interface, *ACM Computing Surveys*, **34**, 90–169, Maret 2002.

[28] Morency, L.-P., Christoudias, C. M., dan Darrell, T., Recognizing gaze aversion gestures in embodied conversational discourse, *Proceedings of the 8th international conference on Multimodal interfaces*, 289–294, ACM, 2006.

[29] Nakano, A. Y., *Exploring Spatial Information For Distant Speech Recognition Under Real Environmental Conditions*, Disertasi Doktor, Toyohashi University of Technology, Maret 2010.

[30] Niewiadomski, R., Bevacqua, E., Mancini, M., dan Pelachaud, C., Greta: an interactive expressive ECA system, *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, 1399–1400, International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[31] Nugraha, A. A., Taufiq, A., Utama, N. I., dan Prihatmanto, A. S., Smart Assistant for Museum's Objects Navigation (SAMsON), *Proceedings of The 5th AOTULE International Postgraduate Students Conference on Engineering*, 186–189, November 2010.

[32] Park, Y., Application of 3D sound technology to intelligent robots, *Proceedings of the 3rd International Universal Communication Symposium on - IUCS '09*, 199–204, 2009.

[33] Rabiner, L. R. dan Schafer, R. W., Introduction to digital speech processing, *Foundations and Trends in Signal Processing*, **1**(1-2), January 2007.

[34] Rehm, M., From Chatterbots to Natural Interaction – Face to Face Communication with Embodied Conversational Agents, *IEICE Transactions on Information and Systems*, **E88-D**(11), 2445–2452, November 2005.

[35] Rickel, J., Marsella, S., Gratch, J., Hill, R., Traum, D., dan Swartout, W., Toward a new generation of virtual humans for interactive experiences, *IEEE Intelligent Systems*, **17**(4), 32–38, Juli 2002.

[36] Shawar, B. A. dan Atwell, E., Chatbots: Are they Really Useful? *LDV Forum*, **22**(1), 29–49, 2007.

[37] Stephens, K. R., What Has the Loebner Contest Told Us About Conversant Systems? URL `http://www.behavior.org/resources/325.pdf`, diakses pada 10 Mei 2011, 02:00 WIB.

[38] Swartout, W., et al., Ada and grace: toward realistic and engaging virtual museum guides, *Proceedings of the 10th international conference on Intelligent virtual agents (IVA'10)*, 286–300, Springer-Verlag, 2010.

[39] Turing, A. M., Computing Machinery and Intelligence, *Mind*, **59**(236), 433–460, Oktober 1950.

[40] Van Es, I., Heylen, D., Van Dijk, B., dan Nijholt, A., Making agents gaze naturally-Does it work? *Proceedings of the Working Conference on Advanced Visual Interfaces*, 357–358, ACM, 2002.

[41] Wallace, R. S., *The Elements of AIML Style*, Maret 2003.

[42] Wallace, R. S., The Anatomy of A.L.I.C.E., *Parsing the Turing Test*, Epstein, R., Roberts, G., dan Beber, G., editor, 181–210, Springer, 2009.

[43] Weizenbaum, J., ELIZA-a computer program for the study of natural language comunication between man and machine, *Communications of the ACM*, **9**(1), 36–45, 1966.

[44] _____, *Chat Robot Survey*, ALICE Artificial Intelligence Foundation, URL `http://www.alicebot.org/aimlbots.html`, diakses pada 10 Mei 2011, 02:00 WIB.

[45] _____, *Hasil Sensus Penduduk 2010: Data Agregat per Provinsi*, Badan Pusat Statistik, 2010.

[46] _____, *Jumlah Pengunjung Museum di Indonesia*, Direktorat Kebudayaan, Pariwisata, Pemuda, dan Olahraga, Badan Perencanaan Pembangunan Nasional, 2009, URL `http://kppo.bappenas.go.id/preview/225`, diakses pada 10 Mei 2011, 17:00 WIB.

[47] _____, *Meet Ada and Grace: Virtual Museum Guides*, The Museum of Science, Boston, URL `http://www.mos.org/interfaces/adagrace.php`, diakses pada 10 Mei 2011, 17:00 WIB.

[48] _____, *Responsive Virtual Human Museum Guides*, USC Institute for Creative Technologies, URL `http://ict.usc.edu/projects/responsive_virtual_human_museum_guides/`, diakses pada 10 Mei 2011, 17:00 WIB.

[49] _____, *Virtual Humans Blog on Science and Women's History*, Armed with Science, Maret 2010, URL `http://science.dodlive.mil/2010/03/04/virtual-humans-blog-on-science-and-womens-history/`, diakses pada 10 Mei 2011, 17:00 WIB.