



**TUGAS AKHIR - TF 181801**

**IDENTIFIKASI FORENSIK SUARA MENGGUNAKAN METODE  
*JOINT FACTOR ANALYSIS* DAN *I-VECTOR***

**ROUDHOTUL JANNAH ROUF  
NRP. 0231164000011**

Dosen Pembimbing:  
Dr. Dhany Arifianto, S.T., M.Eng

Departemen Teknik Fisika  
Fakultas Teknologi Industri Dan Rekayasa Sistem  
Institut Teknologi Sepuluh Nopember  
Surabaya  
2020

*Halaman ini sengaja dikosongkan*



**FINAL PROJECT - TF 181801**

***SPEAKER FORENSIC IDENTIFICATION USING JOIN  
FACTOR ANALYSIS AND I-VECTOR***

**ROUDHOTUL JANNAH ROUF  
NRP. 0231164000011**

**Supervisors:  
Dr. Dhany Arifianto, S.T., M.Eng**

***Department Of Engineering Physics  
Faculty of Industrial Technology and System Engineering  
Institut Teknologi Sepuluh Nopember  
Surabaya  
2020***

*Halaman ini sengaja dikosongkan*

## PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan di bawah ini.

Nama : Roudhotul Jannah Rouf  
NRP : 02311640000011  
Departemen / Prodi : Teknik Fisika / S1 Teknik Fisika  
Fakultas : Fakultas Teknologi Industri & Rekayasa Sistem (FTIRS)  
Perguruan Tinggi : Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul "*IDENTIFIKASI FORENSIK SUARA MENGGUNAKAN METODE JOINT FACTOR ANALYSIS DAN I-VECTOR*" adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada Tugas Akhir ini, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenarnya-benarnya.

Surabaya, 21 Agustus 2020

Yang membuat pernyataan,



Roudhotul Jannah Rouf

NRP. 02311640000011

*Halaman ini sengaja dikosongkan*

**LEMBAR PENGESAHAN  
TUGAS AKHIR**

**IDENTIFIKASI FORENSIK SUARA MENGGUNAKAN METODE  
JOINT FACTOR ANALYSIS DAN I-VECTOR**

Oleh:

**Roudhotul Jannah Rouf**  
**NRP. 0231164000011**

Surabaya, 21 Agustus 2020

**Menyetujui,  
Pembimbing**



**Dr. Dhany Arifianto, S.T., M.Eng**  
**NIP. 19731007 199802 1 001**

**Mengetahui,**

**Kepala Departemen  
Teknik Fisika FT-IRS ITS**



**Dr. Suvanto, S.T., M.T**  
**NIP. 1971111131995121002**

*Halaman ini sengaja dikosongkan*



# LEMBAR PENGESAHAN

## IDENTIFIKASI FORENSIK SUARA MENGGUNAKAN METODE *JOINT FACTOR ANALYSIS* DAN *I-VECTOR*

### TUGAS AKHIR

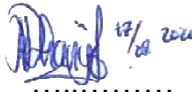


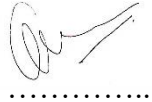
Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Teknik  
pada  
Program Studi S-1 Departemen Teknik Fisika  
Fakultas Teknologi Industri & Rekayasa Sistem (FTIRS)  
Institut Teknologi Sepuluh Nopember

Oleh:

**ROUDHOTUL JANNAH ROUF**

**NRP. 0231164000011**

Disetujui oleh Tim Penguji Tugas Akhir:

- |                                      |   |                 |
|--------------------------------------|---|-----------------|
| 1. Dr. Dhany Arifianto, S.T., M.Eng. | <br>..... | (Pembimbing)    |
| 2. Ir. Wiratno Argo Asmoro, M.Sc.    | <br>..... | (Ketua Penguji) |
| 3. Dr. Irwansyah, S.T., M.T.         | <br>..... | (Penguji I)     |
| 4. Dr. Gunawan Nugroho, S.T., M.T.   | <br>..... | (Penguji II)    |

**SURABAYA**

**2020**

*Halaman ini sengaja dikosongkan*

# IDENTIFIKASI FORENSIK SUARA MENGGUNAKAN METODE *JOINT FACTOR ANALYSIS* DAN *I-VECTOR*

Nama : Roudhotul Jannah Rouf  
NRP : 0231164000011  
Departemen : Teknik Fisika FTIRS - ITS  
Dosen Pembimbing : Dr. Dhany Arifianto, S.T., M.Eng

## ABSTRAK

Forensik suara ucap adalah proses untuk menentukan kecocokan identitas antara suara seseorang (*known speaker*) dengan suara yang akan diselidiki (*suspect speaker*). Untuk meningkatkan keakuratan dalam analisa forensik suara digunakan gabungan 2 metode forensik. Penelitian ini bertujuan untuk membuat sistem identifikasi forensik suara ucap dengan metode *Joint Factor Analysis* dan *i-vector*. Pendekatan forensik dilakukan dengan menambahkan sinyal noise dengan nilai SNR tertentu sebagai representasi situasi penyadapan untuk mengukur performansi identifikasi suara. Verifikasi dilakukan dengan 2 pengujian yaitu perbandingan speaker yang sama dan verifikasi antara speaker yang berbeda. Klasifikasi pada verifikasi penutur menggunakan *i-vector* dilakukan dengan membandingkan model *i-vector* tes dan target. Kedua model tersebut dihitung kemiripan vektornya menggunakan *cosine similarity score*. Proses pengujian pada performa program ditunjukkan oleh nilai *error equal rate*. Sedangkan sensitivitas sistem ditunjukkan oleh nilai *threshold*. Hasil yang diperoleh menunjukkan bahwa nilai EER dataset Graz (1,8%) lebih rendah dibandingkan dengan dataset Indonesia (10%).

**Kata Kunci:** Forensik, *Joint Factor Analysis*, *i-vector*, *Cosine Similarity Score*, *Error Equal Rate*

*Halaman ini sengaja dikosongkan*

## ***SPEAKER FORENSIC IDENTIFICATION USING JOINT FACTOR ANALYSIS AND I-VECTOR***

***Name*** : Roudhotul Jannah Rouf  
***NRP*** : 0231164000011  
***Department*** : Engineering Physics FTIRS - ITS  
***Supervisors*** : Dr. Dhany Arifianto, S.T., M.Eng

### **ABSTRACT**

*Voice forensics is the process of determining the identity match between a person's voice (known speaker) with the voice to be investigated (suspect speaker). To improve accuracy in sound forensic analysis a combination of 2 forensic methods is used. This study aims to create a sound forensic identification system using the Joint Factor Analysis and i-vector methods. The forensic approach is carried out by adding noise signals with certain SNR values as a representation of the tapping situation to measure the performance of sound identification. Verification is done by 2 tests namely the comparison of the same speaker and verification between different speakers. Classification of speaker verification using i-vector is done by comparing the i-vector test model and the target. Both of the models in the vector similarity were calculated using a cosine similarity score. The testing process on program performance is indicated by the error equal rate value. While the sensitivity of the system is shown by the value of the threshold. The results obtained indicate that the Graz dataset EER value (1,8%) is lower than the Indonesian dataset (10%).*

***Keywords: Forensic, Joint Factor Analysis, i-vector, Cosine Similarity Score, Error Equal Rate***

*Halaman ini sengaja dikosongkan*

## KATA PENGANTAR

Dengan menyebut nama Allah Yang Maha Pengasih Lagi Maha Penyayang, puji syukur penulis panjatkan kehadirat Allah SWT. Atas rahmat dan hidayahnya sehingga laporan tugas akhir ini yang berjudul “**Identifikasi Forensik Suara Menggunakan Metode *Joint Factor Analysis* dan *i-vector***” dapat terselesaikan dengan baik.

Penulis telah banyak mendapatkan bantuan dari berbagai pihak dalam pengerjaan Tugas Akhir dan Laporan Tugas Akhir ini. Penulis mengucapkan terimakasih kepada:

1. Bapak Dr. Suyanto, S.T., M.T. selaku Kepala departemen Teknik Fisika ITS
2. Bapak Dr. Dhany Arifianto, S.T, M.Eng selaku dosen pembimbing tugas akhir ini atas waktu dan bimbingan konsultasi yang telah diberikan
3. Segenap Bapak/Ibu dosen pengajar di Departemen Teknik Fisika - ITS
4. Segenap keluarga terutama kedua almarhum orang tua beserta adik tercinta penulis yang telah menjadi motivasi dalam penyelesaian tugas akhir ini.
5. Teman - teman angkatan 2016 dan warga Teknik Fisika - ITS, yang senantiasa memberikan motivasi dan perhatian
6. Teman-teman asisten Laboratorium Vibrasi dan Akustik - Teknik Fisika yang senantiasa memberikan motivasi, perhatian dan dukungan penuh

Serta pihak-pihak lain yang tidak dapat disebutkan satu-persatu. Penulis menyadari bahwa penulisan laporan Tugas Akhir ini tidak sempurna, namun penulis berharap laporan ini dapat memberikan kontribusi dan wawasan yang bermanfaat bagi pembaca. Semoga laporan tugas akhir ini dapat dipergunakan dengan sebaik-baiknya.

Surabaya, 21 Agustus 2020

Penulis

*Halaman ini sengaja dikosongkan*



## DAFTAR ISI

HALAMAN JUDUL.....	i
COVER PAGE.....	iii
PERNYATAAN BEBAS PLAGIASI .....	v
LEMBAR PENGESAHAN .....	vii
LEMBAR PENGESAHAN .....	ix
ABSTRAK.....	xi
ABSTRACT.....	xiii
KATA PENGANTAR .....	xv
DAFTAR ISI.....	xvii
DAFTAR GAMBAR .....	xxi
DAFTAR TABEL.....	xxv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	3
1.5 Sistematika Laporan .....	3
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....	5
2.1 Forensik Suara Ucap .....	5
2.2 Karakterisasi Suara.....	5
2.3 Universal Background Model (UBM).....	8
2.4 Joint Factor Analysis .....	10
2.5 Total Variabilitas .....	12
2.6 Statistik Baum-Welch.....	13

2.7	Ekstraksi i-vector .....	14
2.8	Matriks Proyeksi .....	14
2.9	Cos Similarity Score (CSS) .....	15
2.10	Indikator Kinerja Sistem .....	16
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>19</b>
3.1	Studi Literatur dan Pengumpulan Data.....	19
3.2	Perancangan Program .....	19
3.2.1	Ekstraksi Fitur .....	20
3.2.2	Pelatihan UBM-GMM.....	22
3.2.3	Statistik Baum-Welch.....	25
3.2.4	Total Variabilitas Space.....	25
3.2.5	Ekstraksi i-vector.....	26
3.2.6	Matriks Proyeksi.....	26
3.2.7	Tahap Pendaftaran .....	29
3.2.8	Tahap Verifikasi .....	30
3.3	Simulasi Program dan Pengujian.....	34
3.4	Penarikan Kesimpulan .....	34
3.5	Pembuatan Laporan .....	35
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>		<b>37</b>
4.1	Data Uji Coba .....	37
4.2	Evaluasi Babble Noise .....	39
4.3	Hasil Uji Coba .....	41
4.3.1	Dataset Graz (Clean Data).....	41
4.3.2	Dataset Graz (Lar Data).....	45
4.3.3	Dataset Indonesia (Clean Data).....	49
4.3.4	Dataset Indonesia (Rain Data).....	52

4.3.5	Dataset Indonesia (Babble Data).....	55
4.4	Evaluasi Performansi.....	59
BAB V KESIMPULAN DAN SARAN.....		67
5.1	Kesimpulan.....	67
5.2	Saran.....	67
DAFTAR PUSTAKA .....		69
BIODATA PENULIS .....		73

*Halaman ini sengaja dikosongkan*

## DAFTAR GAMBAR

<b>Gambar 2. 1</b>	Alur ekstraksi fitur MFCC.....	6
<b>Gambar 2. 2</b>	Pembengkokan fitur berdasarkan (J. Pelecanos, 2001) .....	7
<b>Gambar 2. 3</b>	Alur Proses GMM-UBM (Dehak, et al., 2009) .....	8
<b>Gambar 2. 4</b>	Ilustrasi <i>Joint Factor Analysis</i> (P. Kenny, 2008) .....	10
<b>Gambar 2. 5</b>	Distribusi skor target dan non-target .....	16
<b>Gambar 3. 1</b>	Blok diagram proses perancangan program .....	19
<b>Gambar 3. 2</b>	(a) Ilustrasi <i>Speech Segment</i> Dari (Giannakopoulos, 2009), (b) Hasil Salah Satu Deteksi <i>Voice Segment</i> Pada Sinyal Audio.....	21
<b>Gambar 3. 3</b>	Hasil Ekstraksi Fitur Dari Sinyal Suara Audio.....	21
<b>Gambar 3. 4</b>	Eigenvalue LDA Projection Matriks LDA.....	28
<b>Gambar 3. 5</b>	Projection Matriks LDA .....	28
<b>Gambar 3. 6</b>	Hasil Faktorisasi Cholesky B .....	29
<b>Gambar 3. 7</b>	Projection Matriks Gabungan LDA dan WCNN.....	29
<b>Gambar 3. 8</b>	Salah Satu Model I-Vector Dataset <i>Enrollment</i> Dataset Indonesia ( <i>Clean Data</i> ) .....	30
<b>Gambar 3. 9</b>	(a) Salah Satu Model I-Vector Dataset Uji, (b) Salah Satu Model i-Vector Dataset Enrollment Yang Akan Saling Dibandingkan Dataset Indonesia ( <i>Clean Data</i> ) .....	31
<b>Gambar 3. 10</b>	(a) Hasil Scattering Css FRR, (b) Hasil Scattering Css FAR Dari Dataset Graz ( <i>Clean Data</i> ) .....	32
<b>Gambar 3. 11</b>	Hasil FRR, FAR, dan EER .....	34
<b>Gambar 4. 1</b>	Sinyal Suara Ucapan Dataset Graz Pada Clean Data Dan LAR Data	38
<b>Gambar 4. 2</b>	Sinyal Suara Ucapan Dataset Indonesia Pada Clean Data, Rain Data Dan Data Babble .....	38
<b>Gambar 4. 3</b>	Data Intrinsic Suara Clean Data Dataset Indonesia Dan Graz Pada (a) Waveform, (b) Fundamental Frequency, (c) Jumlah Komponen Aperiodik, (d) Ekstraksi Spectral.....	39
<b>Gambar 4. 4</b>	Distribusi Statistik Babble Noise.....	40
<b>Gambar 4. 5</b>	Plot Uji Normalitas Babble Noise .....	40

<b>Gambar 4. 6</b> (a) Ekstraksi FileSuara Menjadi Fitur Mfcc, Delta Mfcc, Dan Delta Delta Mfcc. (b) Ekstraksi File Suara Menjadi Fitur Mfcc Yang Telah Dinormalisasi Dari Dataset Graz (Clean Data).....	42
<b>Gambar 4. 7</b> (a) Model I-Vector Dataset Background, (b) Model I-Vector Dataset Enrollment Dari Dataset Graz (Clean Data).....	42
<b>Gambar 4. 8</b> (a) Salah Satu Model I-Vector Tes, (b) Salah Satu Model I-Vector Target Yang Akan Saling Dibandingkan Dari Dataset Graz (Clean Data).....	43
<b>Gambar 4. 9</b> i-vector Setiap Penutur Pada Data Enrollment Dataset Graz (Clean Data) .....	44
<b>Gambar 4. 10</b> (a) Hasil scattering css FRR, (b) Hasil scattering css FAR dari dataset Graz (clean data).....	44
<b>Gambar 4. 11</b> Hasil ERR dari dataset Graz (clean data) dari dataset Graz (clean data) .....	45
<b>Gambar 4. 12</b> (a) ekstraksi file suara menjadi fitur mfcc, delta mfcc, dan delta delta mfcc. (b) ekstraksi file suara menjadi fitur mfcc yang telah dinormalisasi dari dataset Graz (lar data).....	46
<b>Gambar 4. 13</b> (a) Scattering Model I-Vector Dataset Background, (b) Scattering Model I-Vector Dataset Enrollment Dari Dataset Graz (Lar Data).....	46
<b>Gambar 4. 14</b> (a) Salah Satu Model I-Vector Tes, (b) Salah Satu Model I-Vector Target Dari Dataset Graz (Lar Data) .....	47
<b>Gambar 4. 15</b> i-vector Setiap Speaker Data Enrollment Dataset Graz (Lar Data) .....	47
<b>Gambar 4. 16</b> (a) Hasil scattering css FRR, (b) Hasil scattering css FAR dari dataset Graz (lar data).....	48
<b>Gambar 4. 17</b> Hasil ERR dari dataset Graz (lar data).....	48
<b>Gambar 4. 18</b> Hasil Ekstraksi Fitur dan Normalisasi File Suara Clean Data .....	49
<b>Gambar 4. 19</b> (a) Model i-vector data training, (b) Model i-vector data enrollment Dataset Indonesia (Clean Data) .....	50
<b>Gambar 4. 20</b> i-vector Setiap Speaker Pada Data Enrollment .....	50
<b>Gambar 4. 21</b> (a) Model i-vector Tes, (b) Model i-vector Target Dataset Indonesia (Clean Data).....	51

<b>Gambar 4. 22</b> (a) Hasil css FAR, (b) Hasil css FRR Dataset Indonesia (Clean Data) .....	51
<b>Gambar 4. 23</b> Plot Grafik FAR, FRR dan EER Dataset Indonesia (Clean Data)	52
<b>Gambar 4. 24</b> Hasil Ekstraksi Fitur File Suara Ucap Data Rain .....	53
<b>Gambar 4. 25</b> Hasil i-vector Model Data Enrollment Pada Data Rain.....	53
<b>Gambar 4. 26</b> (a) Model i-vector Tes Dan (b) Model i-vector pada Target.....	54
<b>Gambar 4. 27</b> (a) Hasil CSS FAR dan (b) Hasil CSS FRR Pada Data Rain .....	54
<b>Gambar 4. 28</b> Hasil Plot Grafik FAR, FRR, dan EER pada data rain.....	55
<b>Gambar 4. 29</b> Hasil ekstraksi fitur file suara ucap (a) data babble – SNR 0, (b) data babble- SNR 10, (c) data babble - SNR 20. ....	55
<b>Gambar 4. 30</b> Hasil ivector model data enrollment pada (a) data babble – SNR 0, (b) data babble - SNR 10, (c) data babble - SNR 20.....	56
<b>Gambar 4. 31</b> Model ivector tes dan target pada (a) & (b) data babble - SNR 0, (c) & (d) data babble – SNR 10, (e) & (f) data babble – SNR 20 .....	57
<b>Gambar 4. 32</b> Hasil CSS FAR dan CSS FRR pada (a) & (b) data babble - SNR 0, (c) & (d) data babble – SNR 10, (e) & (f) data babble – SNR 20.....	58
<b>Gambar 4. 33</b> Hasil Plot Grafik FAR, FRR, dan EER pada (a) data babble - SNR 0, (b) data babble – SNR 10, (c) data babble – SNR 20 .....	59
<b>Gambar 4. 34</b> (a) waveform, (b) spectrogram dan (c) F0 Graz “all year”, (d) waveform, (e) spectrogram dan (f) F0 indonesia “saya suka” .....	61
<b>Gambar 4. 35</b> Struktur waveform, spectrogram dan F0 (a)-(c) file clean, (d)-(f) data dengan babble noise SNR 20, (g)-(i) data dengan babble noise SNR 10, dan (j)-(l) data dengan babble noise SNR 0.....	62
<b>Gambar 4. 36</b> Pengaruh Banyak Speaker Pada Data Training Untuk Performansi Sistem.....	63
<b>Gambar 4. 37</b> Hasil Pencampuran Database Indonesia dan Graz .....	64
<b>Gambar 4. 38</b> Pengaruh Gender Terhadap Performansi.....	65

*Halaman ini sengaja dikosongkan*



## DAFTAR TABEL

<b>Tabel 3. 1</b> Pengujian Speaker Sama Dataset Graz (Clean Data).....	32
<b>Tabel 3. 2</b> Pengujian Speaker Acak Dataset Graz (Clean Data) .....	33
<b>Tabel 4. 1</b> Hasil EER dan Akurasi Pada Dataset Noise .....	60

*Halaman ini sengaja dikosongkan*

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Salah satu barang bukti yang banyak digunakan dalam berbagai jenis kasus adalah berupa rekaman suara (audio). Umumnya rekaman akan berupa percakapan yang menjadi bukti petunjuk adanya perbuatan yang disangkakan. Sebagian besar rekaman berwujud audio digital, maka upaya untuk melakukan analisa terhadap rekaman audio dilakukan dengan menggunakan pendekatan audio Forensik. Forensik suara ucap adalah proses untuk menentukan kecocokan identitas antara suara seseorang (*known speaker*) dengan suara yang akan diselidiki (*suspect speaker*) (Rose, 2002). Jika hasil dari analisa forensik suara ucap dijadikan sebagai bukti utama untuk memutuskan suatu perkara di pengadilan, maka dibutuhkan sistem forensik suara ucap yang secara akurat dengan pembacaan error sekecil mungkin.

Tantangan utama komunitas riset pengenalan speaker selama beberapa tahun terakhir adalah bagaimana mengatasi masalah variabilitas sesi dan ketidaksesuaian saluran yang disebabkan oleh berbagai faktor seperti: keadaan emosi pembicara, kondisi lingkungan, perangkat rekaman, saluran transmisi yang berbeda, dsb (Boulkenafet, et al., 2013). Keragaman ini menurunkan secara drastis kinerja sistem klasik berdasarkan pada paradigma Gaussian Mixture Model-Universal Background Model (GMM-UBM) (D. Reynolds, 2000). Terlepas dari kenyataan bahwa kehandalan sistem GMM-UBM terhadap variabilitas sesi dapat ditingkatkan dengan berbagai fitur, model, dan teknik kompensasi skor, pencapaian paling sukses yang menghasilkan sistem verifikasi speaker berkinerja terbaik didasarkan pada pembelajaran variabilitas sesi ini. Dalam konteks ini, *Joint Factor Analysis* (JFA) (P.Kenny, 2005) dan pemodelan Total Variability i-vektor (N. Dehak, 2011) telah mendapatkan prevalensi di hampir semua sistem verifikasi penutur teks independen yang canggih. Baru-baru ini, Mandasari et al (M. Mandasari, 2011) mempelajari efek dari durasi bicara pada kalibrasi skor sistem i-

vektor untuk mengevaluasi kekuatan bukti menggunakan database NIST SRE-2010.

Dalam analisa forensik suara ucap, umumnya jumlah data suara yang ada terbatas dan dibutuhkan dalam waktu yang singkat. Secara umum, sistem pengenalan suara menggunakan UBM-GMM untuk mendapatkan karakteristik general dari *speaker*. Salah satu kesulitan utama sistem GMM-UBM yaitu melibatkan variabilitas intersesi. *Joint factor analysis* (JFA) diusulkan untuk mengkompensasi variabilitas ini dengan memodelkan variabilitas inter-speaker secara terpisah dan variabilitas saluran atau sesi dan mengabaikan faktor saluran. Namun ditemukan bahwa terdapat informasi tentang speaker pada faktor saluran dalam JFA (Dehak, 2009), sehingga total variabilitas ruang adalah pengembangan dari *Joint Factor Analysis* klasik yang digunakan untuk mengkompensasi informasi tentang pembicara di dalam faktor saluran dari JFA dengan mengetahui total variabilitas ruangnya. sedangkan *i-vector* digunakan untuk mengekstraksi faktor total atau variabel tersembunyi menggunakan statistic Baum-Welch untuk menemukan *unknown parameters* (P. Kenny, 2008). Oleh karena itu penelitian ini bertujuan untuk membuat sistem identifikasi forensik suara ucap dengan metode *Joint Factor Analysis* dan *i-vector*.

## 1.2 Rumusan Masalah

Permasalahan yang akan diselesaikan dalam penelitian tugas akhir ini adalah:

- a) Bagaimana membuat program identifikasi suara untuk analisa forensik dengan sampel data suara yang terbatas dan terdapat lebih dari 1 penutur dengan menggunakan metode JFA dan *i-vector*?
- b) Bagaimana mengukur performansi dari sistem identifikasi yang telah dibuat?

## 1.3 Tujuan

Tujuan dari penelitian tugas akhir ini adalah sebagai berikut:

- a) Dapat membuat identifikasi suara untuk analisa forensik dengan sampel data suara yang terbatas dan terdapat lebih dari 1 penutur dengan menggunakan metode JFA dan *i-vector*
- b) Dapat mengukur performansi dari sistem identifikasi yang telah dibuat

#### **1.4 Batasan Masalah**

Agar pembahasan tidak meluas dan menyimpang dari tujuan dan rumusan masalah, akan diberikan beberapa batasan permasalahan dari penelitian ini, yaitu sebagai berikut:

- a) Data yang digunakan adalah data sekunder berupa dataset Graz (20 penutur) sebagai data utuh dan dataset Bahasa Indonesia (8 Penutur).
- b) Simulasi forensik dilakukan dengan pendekatan penambahan noise / derau suara berupa babble noise dan suara hujan dengan SNR 0, SNR 10 dan SNR 20.
- c) Data Bahasa Indonesia diasumsikan sudah otentik untuk keperluan identifikasi dalam analisa forensik

#### **1.5 Sistematika Laporan**

Sistematika Laporan tugas akhir ini adalah sebagai berikut:

a) **BAB I PENDAHULUAN**

Bab ini menjelaskan dari latar belakang permasalahan, rumusan masalah, tujuan, batasan masalah dan sistematika laporan.

b) **BAB II TINJAUAN PUSTAKA DAN DASAR TEORI**

Bab membahas tentang studi pustaka materi yang berkaitan dengan tugas akhir yang dilakukan.

c) **BAB III METODOLOGI PENELITIAN**

Bab ini berisi tahapan-tahapan yang dilakukan untuk menjawab rumusan masalah yang diambil demi tercapainya tujuan dalam Tugas Akhir ini. Selain itu, dibahas pula mengenai proses perancangan dan implementasi sistem mulai dari proses pengolahan data masukan hingga mendapatkan data keluaran berupa hasil verifikasi suara.

d) **BAB IV HASIL DAN PEMBAHASAN**

Bab ini akan membahas tentang data uji coba, evaluasi noise yang digunakan dan hasil uji coba dari setiap simulasi verifikasi suara yang dilakukan hingga analisa performansi dari sistem identifikasi suara yang telah dibuat.

e) **BAB V KESIMPULAN DAN SARAN**

Pada bab V berisi mengenai kesimpulan tentang tugas akhir yang dilakukan berdasarkan data-data yang diperoleh pada bab IV untuk menjawab rumusa masalah yang diberikan serta saran untuk pengembangan tugas akhir selanjutnya.

## **BAB II**

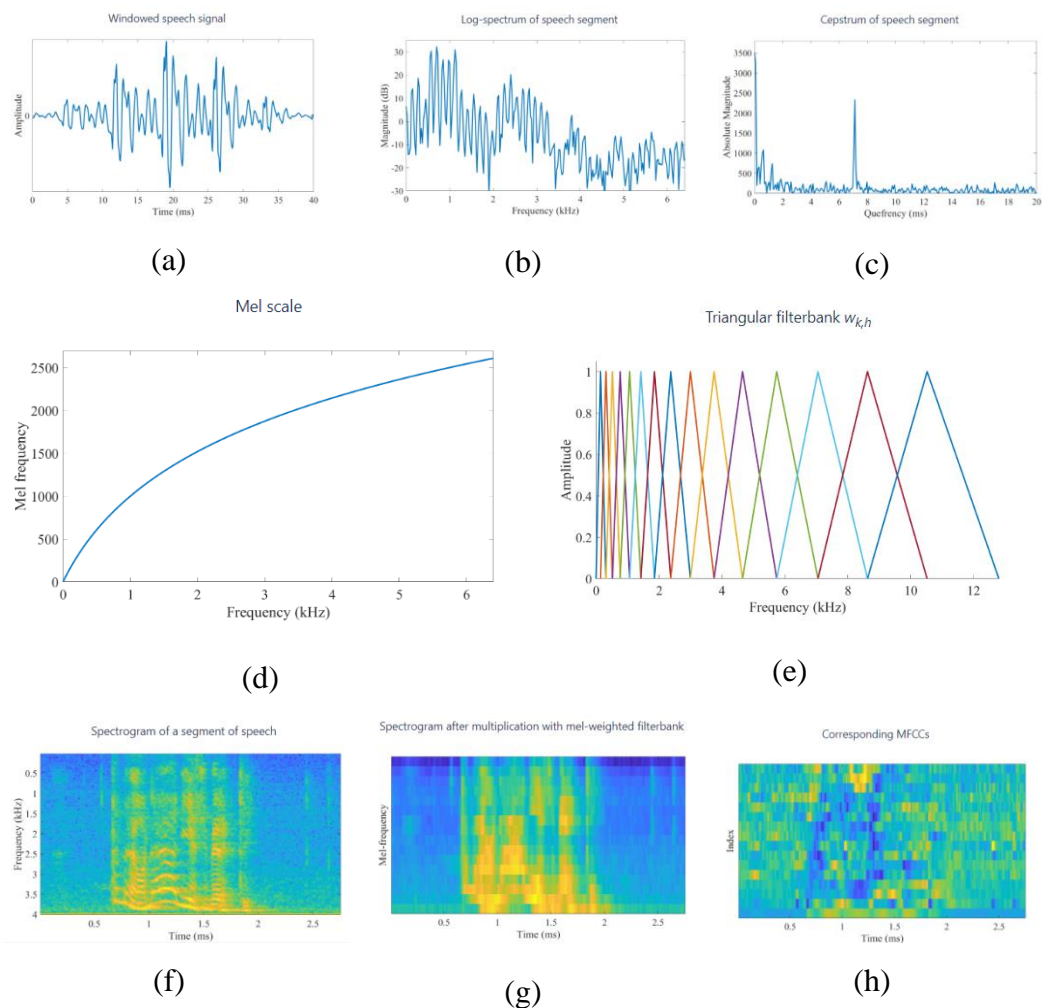
### **TINJAUAN PUSTAKA DAN DASAR TEORI**

#### **2.1 Forensik Suara Ucap**

Identifikasi pembicara dalam konteks forensik umumnya berupa proses tentang membandingkan suara yang melibatkan perbandingan satu atau lebih sampel suara pelaku dengan satu atau lebih sampel suara tersangka (Rose, 2002). Meskipun asumsi umum bahwa pembicara yang berbeda memiliki suara yang berbeda, penting untuk memahami bahwa suara dari pembicara yang sama akan selalu bervariasi juga. Ini merupakan kebenaran fonetis dimana tidak seorang pun pernah mengatakan hal yang sama persis dengan cara yang sama (Rose, 1996). Oleh karena itu perlu untuk memahami variasi dalam pembicara, istilah ini biasanya disebut variasi dalam-speaker (*within-speaker*) atau (*intra-speaker*). Istilah *intra-speaker* mirip dengan variasi antar pembicara. Namun, konsistensi yang paling penting dari variasi ini adalah bahwa akan selalu ada perbedaan antara sampel suara ucap, bahkan jika sampel suara ucap tersebut berasal dari pembicara yang sama. Perbedaan-perbedaan ini yang biasanya terdengar, dapat terukur dan dapat dikuantifikasi serta dievaluasi dengan identifikasi pembicara forensik. Dengan kata lain, identifikasi pembicara forensik melibatkan kemampuan untuk mengetahui apakah perbedaan yang tak terhindarkan antara sampel lebih cenderung menjadi perbedaan dalam penutur (*within speaker*) atau perbedaan antara penutur (Rose, 2002).

#### **2.2 Karakterisasi Suara**

Spektrum daya (*Power spectra*) sinyal wicara berisi informasi terpenting tentang fitur sinyal wicara seperti identitas vocal (*vowels*). Namun, rentang nilai sangat tidak seragam. Spektrum logaritmik merupakan representasi yang lebih mudah diakses. Spektrum logaritmik memvisualisasikan konten spektral nilai yang seragam di seluruh spektrum.



**Gambar 2. 1** Alur ekstraksi fitur MFCC

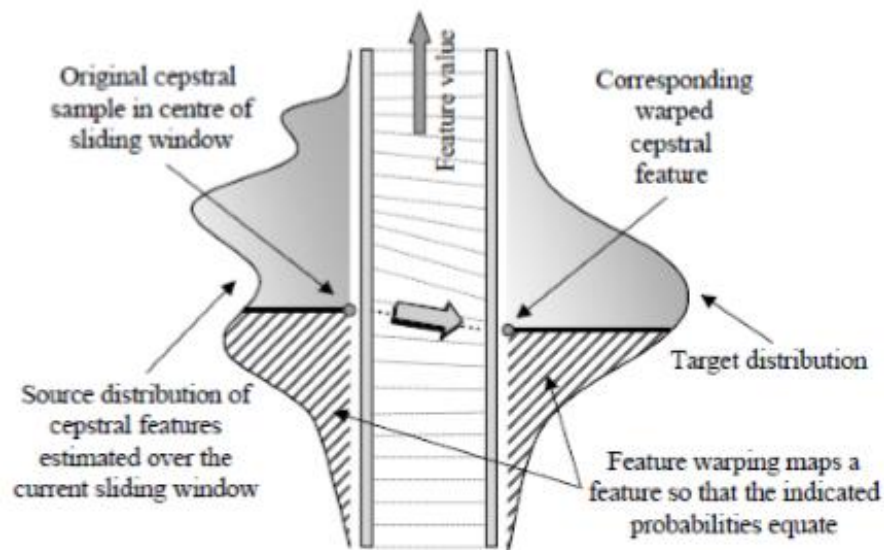
a. MFCC (Mel Frequency Cepstrum Cepstral)

MFCC adalah representasi jangka pendek dari spektrum suara. Bentuk gelombang sampel suara diasumsikan hampir stasioner dalam interval waktu singkat 20 hingga 30 msec. Prosedur ekstraksi fitur MFCC melibatkan jendela analisis geser di sepanjang sinyal ucapan. Untuk setiap penempatan jendela, pidatonya ditekankan dan spektrum daya dihitung. Sebuah bank filter dari filter pembobot segitiga kemudian digunakan untuk menghitung energi rata-rata di sekitar frekuensi tengah setiap segitiga. Filter didistribusikan pada skala Mel, yang mendekati perilaku sistem pendengaran manusia. Sehingga MFCC didefinisikan sebagai Discrete Cosine Transform (DCT) dari logaritma energi bank filter (Esfahani, 2018).



Salah satu langkah dasar dalam sistem pengenalan pola ekstraksi fitur dari suara ucap. Dalam konteks pengenalan pembicara, fitur yang diperoleh dari sinyal suara dapat mencerminkan informasi pembicara secara diskriminatif yang berasal dari spektrum bicara. Spektrum tersebut terkait erat dengan fisiologi saluran vokal manusia sebagai faktor pembeda yang penting. Tipe fitur yang paling populer yaitu MFCC (Furui, 2004).

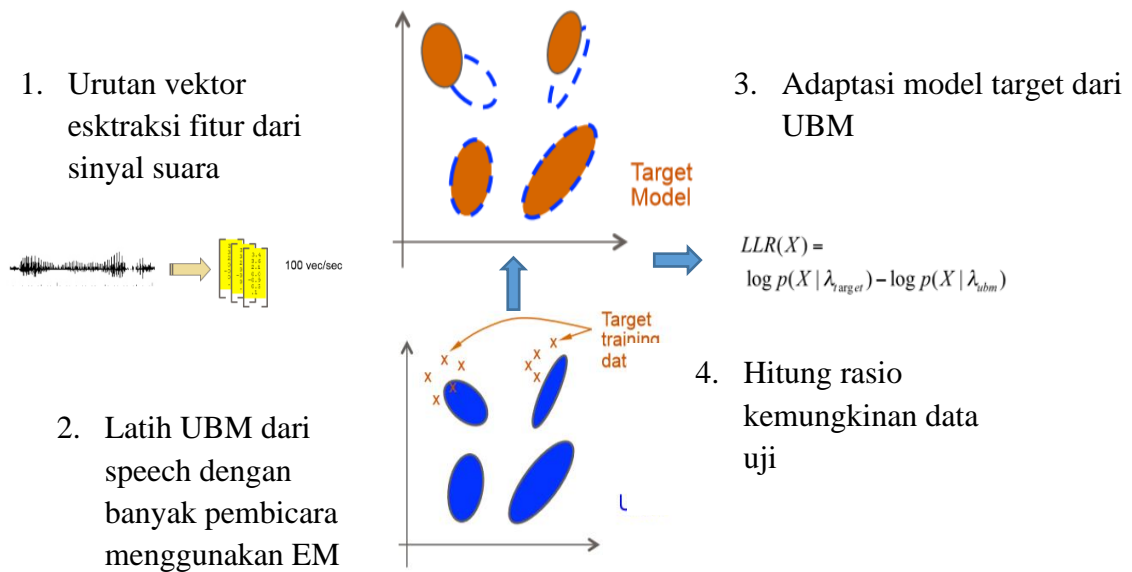
b. Normalisasi Fitur



**Gambar 2. 2** Pembengkokan fitur berdasarkan (J. Pelecanos, 2001)

Fitur-fitur perlu dinormalisasi untuk mengurangi varian antar-sampel (*between-sample*) atau *within-speaker*. Nilai-nilai MFCC yang diekstraksi dari sampel suara jangka menengah didistribusikan secara Gaussian, sehingga efek ketidakcocokan antar sampel (*between-sample*) dapat dikurangi dengan membengkokkan mereka ke dalam distribusi target sesuai dengan probabilitas sumber yang sama (*equal source probability*) (J. Pelecanos, 2001).

### 2.3 Universal Background Model (UBM)



**Gambar 2.3** Alur Proses GMM-UBM (Dehak, et al., 2009)

Pemodelan statistik digunakan untuk menemukan perkiraan distribusi suara. Dalam verifikasi pembicara, banyak model telah digunakan dan diusulkan. Untuk penelitian ini metode yang digunakan adalah Gaussian Mixture Model (GMM). GMM adalah model probabilistik yang mengasumsikan semua titik data dihasilkan dari bobot campuran  $C$  multivariat distribusi Gaussian dengan parameter yang tidak diketahui. Dalam pemodelan ini berlaku algoritma Expectation Maximization (EM) (Reynolds & Rose, 1995) untuk memperkirakan parameter model kemungkinan maksimum. Implementasi menggunakan Universal Background Model (UBM) (Reynolds, et al., 2000) dapat mewakili distribusi fitur secara normal dan tidak tergantung pada speaker dan kemudian melakukan adaptasi untuk melatih model target. Penilaian dilakukan dengan menghitung uji rasio log-likelihood.

Langkah pertama identik dengan langkah ekspektasi dari algoritma EM, di mana perkiraan statistik yang memadai dari data pelatihan pembicara dihitung untuk setiap campuran dalam UBM.

$$p(x|\lambda) = \sum_{i=1}^M w_i p_i(x) \quad (2.2)$$

Persamaan 2.2 digunakan untuk mengitung *mixture density* untuk D-dimensional fitur vector ( $x$ ). Sedangkan densitas Gaussian M unimodal,  $p_i(x)$ , masing-masing parameter dengan vektor  $D \times 1$  rata-rata ( $\mu_i$ ) dan matriks kovarians  $D \times D$  ( $\Sigma_i$ ) dapat menggunakan persamaan 2.3 (Reynolds, et al., 2000).

$$p_i(x) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_i)' (\Sigma_i)^{-1} (x - \mu_i) \right\} \quad (2.3)$$

Menghitung probabilitas normalisasi posterior:

$$\log p(X|\lambda) = \sum_{t=1}^T \log p(x_t|\lambda) \quad (2.4)$$

Umumnya, vektor fitur  $X$  diasumsikan independen, jadi loglikelihood model  $\lambda$  untuk urutan vektor fitur,  $X = \{x_1, \dots, x_T\}$ , dihitung menggunakan persamaan 2.4 (Reynolds, et al., 2000). Tahap ini nilai loglikelihood model  $\lambda$  seperti pada persamaan diatas, dimana di dalam program ini diinisiasi menggunakan variabel  $L$

Sedangkan langkah kedua dari algoritma EM yaitu *maximization* digunakan untuk adaptasi perkiraan *sufficient statistic* baru kemudian digabungkan dengan *old sufficient statistics* dari *UBM mixture parameters* menggunakan *data-dependent mixing coefficient*. Dengan diberi UBM dan vektor pelatihan dari pembicara yang dihipotesiskan,  $X = \{x_1, \dots, x_T\}$ , pertama-tama menentukan penyalarsan probabilistik vektor pelatihan ke dalam komponen campuran yaitu untuk campuran  $i$  di UBM, dapat dihitung dengan persamaan 2.5

$$\Pr(i|xt) = \frac{w_i p_i(x_t)}{\sum_{j=1}^M w_j p_j(x_t)} \quad (2.5)$$

Koefisien adaptasi yang mengendalikan keseimbangan antara estimasi lama dan baru adalah  $(\alpha_i^w, \alpha_i^m, \alpha_i^v)$  untuk masing-masing bobot (weight), rata-rata (mean) dan varians.

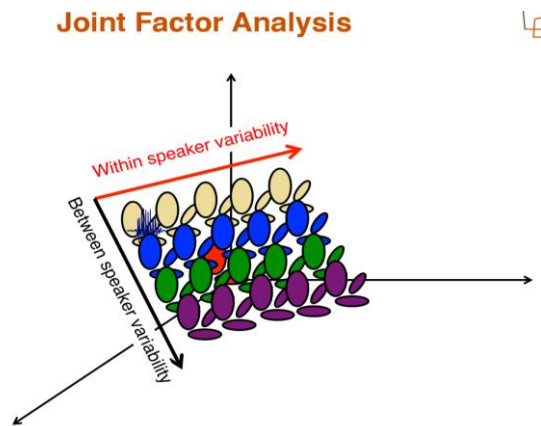
$$\hat{w}_i = \left[ \frac{\alpha_i^w n_i}{T} + (1 - \alpha_i^w) w_i \right] \gamma \quad (2.6)$$

$$\hat{\mu}_i = \alpha_i^m E_i(x) + (1 - \alpha_i^m) \mu_i \quad (2.7)$$

$$\hat{\sigma}_i^2 = \alpha_i^v E_i(x^2) + (1 - \alpha_i^v)(\sigma_i^2 + \mu_i^2) - \hat{\mu}_i^2 \quad (2.8)$$

Faktor skala ( $\gamma$ ) dihitung pada semua *mixture weight* yang disesuaikan untuk memastikan parameter tersebut berjumlah menjadi satu.

## 2.4 Joint Factor Analysis



**Gambar 2. 4** Ilustrasi *Joint Factor Analysis* (P. Kenny, 2008)

*Joint Factor Analysis* adalah model yang digunakan untuk mengkompensasi permasalahan terkait pembicara dan variabilitas sesi dalam GMM. Dalam model ini, setiap pembicara direpresentasikan melalui rata-rata, kovarian, dan bobot campuran  $C$  multivariat. Gaussian density didefinisikan dalam beberapa fitur kontinu ruang dimensi  $F$ . GMM untuk pembicara target diperoleh dengan mengadaptasi parameter rata rata Universal Background Model (UBM). Dalam *Joint Factor Analysis* (P. Kenny, 2008) ucapan pembicara diwakili oleh supervector ( $M$ ) yang terdiri dari komponen tambahan dari speaker dan subruang saluran/sesi. Secara khusus, supervektor yang bergantung pada speaker didefinisikan dalam persamaan 2.9.

$$M = m + Vy + Ux + Dz \quad (2.9)$$

di mana  $m$  adalah super-speaker dan sesi-independen (umumnya dari Model Latar Belakang Universal (UBM)),  $V$  dan  $D$  mendefinisikan subruang speaker (eigenvoice matrix dan diagonal residual, berturut-turut), dan  $U$  mendefinisikan subruang sesi (matriks eigenchannel). Vektor  $y$ ,  $z$  dan  $x$  adalah speaker dan faktor

yang tergantung pada sesi di masing-masing subruang dan masing-masing diasumsikan sebagai variabel acak dengan Normal distribusi  $N(0, I)$ . Untuk menerapkan JFA ke sistem pengenalan pembicara terdiri dari pertama memperkirakan subruang (yaitu,  $V, D, U$ ) dari korpora pengembangan tepat diberi label dan kemudian memperkirakan faktor pembicara dan sesi (yaitu,  $x, y, z$ ) untuk diberi ucapan target baru (N. Dehak, 2011).

Asumsi dasarnya adalah bahwa supervektor yang bergantung pada speaker dan saluran bergantung pada  $M$  yang dapat di dekomposisi menjadi jumlah dari dua supervektor: supervektor speaker  $s$  dan supervektor saluran  $c$ ,

$$M = s + c \quad (2.10)$$

di mana  $s$  dan  $c$  terdistribusi secara normal. Dalam (P. Kenny, 2008) Kenny et al. dijelaskan bagaimana *speaker dependent supervector* dan *channel dependent supervector* dapat direpresentasikan dalam *low dimensional space*. Istilah  $s$  dimodelkan dengan mengasumsikan supervektor pembicara untuk pembicara yang dipilih secara acak

$$s = m + Vy + Dz \quad (2.11)$$

di mana  $m$  adalah pembicara dan saluran supervektor independent (UBM),  $D$  adalah matriks diagonal,  $V$  adalah matriks persegi panjang pangkat rendah, serta  $y$  dan  $z$  adalah vektor acak independen yang memiliki standar distribusi normal. Dengan kata lain,  $s$  dianggap distribusi normal dengan mean  $m$  dan matriks kovarians  $VV^* + DD^*$ . Komponen  $y$  dan  $z$  masing-masing adalah speaker dan factor umum. Supervektor yang bergantung pada saluran  $c$ , yang mewakili efek saluran dalam ucapan, Dapat direpresentasikan dalam persamaan 2.12 berikut:

$$c = Ux \quad (2.12)$$

di mana  $U$  adalah matriks segi empat dengan peringkat rendah (dikenal sebagai *eigenchannel matrix*),  $x$  adalah vektor yang didistribusikan dengan distribusi normal standar. Ini sama dengan mengatakan bahwa  $c$  biasanya terdistribusi dengan rata-rata nol dan kovarian  $UU^*$ . Komponen  $x$  adalah saluran faktor dalam pemodelan analisis faktor.

Tugas mendasar di JFA adalah untuk melatih hyperparameters  $U, V$ , dan  $D$  pada set *training* besar. Dalam kerangka Bayesian, distribusi posterior faktor dapat dihitung menggunakan data pendaftaran (*enrollment*). Kemungkinan (*likelihood*)

tes ujaran  $X$  dihitung dengan mengintegrasikan lebih dari distribusi posterior  $y$  dan  $z$ , dan distribusi  $x$  sebelumnya (N. Dehak, 2011). Penilaian dilakukan dengan komputasi kemungkinan vektor fitur ujaran terhadap model speaker kompensasi sesi ( $M - Ux$ ).

## 2.5 Total Variabilitas

Pemodelan JFA klasik berdasarkan speaker dan saluran faktor terdiri dalam mendefinisikan dua ruang yang berbeda, pembicara ruang yang ditentukan oleh matriks eigenvoice  $V$  dan saluran ruang diwakili oleh eigenchannel matriks  $U$ . Pendekatan yang usulkan didasarkan pada pendefinisian hanya satu ruang, bukannya dua ruang terpisah. Ruang baru ini, yang kemudian disebut sebagai "ruang total variabilitas", berisi speaker dan variabilitas saluran secara bersamaan. Hal ini didefinisikan oleh total variability matrix yang berisi vektor eigen dengan nilai eigen terbesar dari total variabilitas matriks kovarians. Pada model baru, tidak terdapat perbedaan antara efek pembicara dan efek saluran di ruang supervector GMM. Pendekatan dilakukan karena hasil dari eksperimen (Dehak, 2009) yang menunjukkan bahwa faktor saluran JFA yang biasanya memodelkan hanya efek saluran juga mengandung informasi tentang pembicara. Sehingga saat diberikan ucapan, pembicara baru dan supervector GMM tergantung-saluran persamaan dapat didefinisikan sebagai berikut:

$$M = m + Tw \quad (2.13)$$

dimana  $m$  adalah speaker- dan channel-independent supervector (yang dapat dianggap sebagai supervector UBM),  $T$  adalah matriks segi empat dengan peringkat rendah dan  $w$  adalah acak vektor memiliki distribusi normal standar  $N(0, I)$ . komponen vektor  $w$  adalah faktor total. Vektor baru ini sebagai vektor identitas atau vektor-  $i$ .

Dalam pemodelan ini,  $M$  diasumsikan berdistribusi normal dengan mean vektor  $m$  dan matriks kovarians  $TT^t$ . Proses pelatihan total variabilitas matriks  $T$  persis sama seperti mempelajari matriks eigenvoice  $V$ , kecuali satu perbedaan penting yaitu dalam pelatihan eigenvoice, semua rekaman dari pembicara yang diberikan dianggap milik pembicara yang sama (P.Kenny, 2005). Dalam total variabilitas matriks, seluruh rangkaian ucapan yang diberikan

pembicara dianggap sebagai hasil produksi dari pembicara yang berbeda (data direpresentasikan seakan akan setiap ucapan dari pembicara tertentu diproduksi oleh yang berbeda speaker). Model baru yang diusulkan dapat dilihat sebagai analisis faktor sederhana yang memungkinkan kita memproyeksikan pidato ucapan ke ruang variabilitas total dimensi rendah.

Faktor total  $w$  adalah variabel tersembunyi, yang dapat didefinisikan oleh distribusi posteriornya yang dikondisikan oleh Baum-Welch statistik untuk ucapan yang diberikan. Distribusi posterior ini adalah sebuah Distribusi Gaussian dan rata-rata dari distribusi ini sesuai persis dengan vektor- $i$  kami. Statistik Baum-Welch diekstraksi menggunakan UBM (P. Kenny, 2008).

## 2.6 Statistik Baum-Welch

Menurut pendekatan dari hasil dari eksperimen (Dehak, 2009) menunjukkan bahwa faktor saluran JFA juga mengandung informasi tentang pembicara, informasi ini tersimpan kedalam variabel tersembunyi (*hidden variabel*) atau dalam istilah lain adalah faktor total, sehingga Statistik Baum-Welch digunakan untuk mengkompensasi variabel tersembunyi tersebut. Misalkan terdapat urutan frame  $L \{y_1, y_2, \dots, y_L\}$  dan UBM  $\Omega$  terdiri dari komponen campuran  $C$  yang didefinisikan di beberapa ruang fitur dimensi  $F$ . Baum-Welch statistik diperlukan untuk memperkirakan vektor- $i$  untuk penutur dalam ujaran  $u$  dapat diperoleh dengan persamaan sebagai berikut:

$$N_c = \sum_{t=1}^L P(c|y_t, \Omega) \quad (2.14)$$

$$F_c = \sum_{t=1}^L P(c|y_t, \Omega) y_t \quad (2.15)$$

dengan  $c = 1, \dots, C$  adalah indeks Gaussian dan  $P(c|y_t, \Omega)$  sesuai dengan probabilitas posterior komponen campuran  $c$  menghasilkan vektor  $\gamma_t$ . Statistik Baum-Welch adalah statistik  $N$  (orde ke 0) dan  $F$  (orde pertama) yang digunakan pada algoritma EM. Untuk memperkirakan vektor- $i$ , juga diperlukan menghitung Baum-Welch orde pertama yang tersentralisasi statistik berdasarkan UBM berarti komponen campuran.

$$\hat{F}c = \sum_{t=1}^L P(c|y_t, \Omega) y_t (y_t - m_c) \quad (2.16)$$

dimana  $m_c$  adalah rata-rata komponen campuran UBM  $c$ . Setelah orde 0 dan orde pertama statistic baum-welch berdasarkan set *training* dengan persamaan tersebut, kemudian statistic diperluas kedalam matriks  $N(s)$  dan center  $F(s)$ .  $N(s)$  adalah sebuah  $CF \times CF$  matrik diagonal yang bloknya adalah  $N(s)I(c = 1, \dots, C)$ .  $F(s)$  adalah sebuah supervector  $CF \times 1$  didapatkan dengan menggabungkan  $F(s)(c = 1, \dots, C)$ . Dengan  $C$  adalah jumlah komponen dari UBM dan  $F$  adalah jumlah fitur dalam vector fitur.

## 2.7 Ekstraksi i-vector

i-vektor untuk ucapan tertentu dapat diperoleh dengan menggunakan persamaan 2.17 berikut:

$$w = (I + T^t \Sigma^{-1} N(u) T)^{-1} \cdot T^t \Sigma^{-1} \hat{F}(u) \quad (2.17)$$

$N(u)$  didefinisikan sebagai matriks diagonal dimensi  $CF \times CF$  yang blok diagonalnya adalah  $N_c I(c = 1, \dots, C)$ .  $\hat{F}(u)$  adalah supervector dimensi  $CF \times 1$  diperoleh dengan menggabungkan semua orde pertama Baum-Welch statistik  $\hat{F}c$  untuk ucapan yang diberikan  $u$ .  $\Sigma$  adalah matriks kovarian diagonal dimensi  $CF \times CF$  yang diestimasi selama pelatihan analisis faktor (P.Kenny, 2005) dan model tersebut variabilitas residual yang tidak ditangkap oleh variabilitas total matriks T (N. Dehak, 2011).

## 2.8 Matriks Proyeksi

*Backend* telah diusulkan untuk i-vector, namun yang paling mudah dan masih berkinerja baik adalah kombinasi *linear discriminant analysis (LDA)* and *within-class covariance normalization (WCCN)*. LDA digunakan untuk meminimalkan varians intra-kelas dan memaksimalkan varians antara speaker. Itu dapat dihitung seperti dijelaskan dalam (Dehak, et al., 2009), dengan persamaan berikut:

$$S_b = \sum_{s=1}^S (\bar{w}_s - \bar{w})(\bar{w}_s - \bar{w})' \quad (2.18)$$



$$S_w = \sum_{s=1}^s \frac{1}{n_s} \sum_{i=1}^{n_s} (w_i^s - \bar{w}_s)(w_i^s - \bar{w}_s)' \quad (2.19)$$

Dengan  $\bar{w}_s = \left(\frac{1}{n_s}\right) \sum_{i=1}^{n_s} w_i^s$  adalah rata-rata i-vector untuk setiap speaker,  $\bar{w} = \frac{1}{N} \sum_{s=1}^s \sum_{i=1}^{n_s} w_i^s$  adalah rata-rata i-vector pada seluar speaker,  $n_s$  adalah jumlah suara ucap untuk setiap speaker. Sehingga persamaan eigenvalue untuk eigenvectors adalah sebagai berikut:

$$S_b v = \lambda S_w v \quad (2.20)$$

Sedangkan WCCN digunakan untuk menskala ruang i-vector yang berbanding terbalik dengan *in-class covariance*, sehingga arah variabilitas *intra-speaker* yang tinggi tidak ditekankan pada i-vector (N. Dehak, 2011), dapat dinyatakan dalam persamaan berikut:

$$W = \frac{1}{S} \sum_{s=1}^s \frac{1}{n_s} \sum_{i=1}^{n_s} (w_i^s - \bar{w}_s)(w_i^s - \bar{w}_s)' \quad (2.21)$$

Dengan  $\bar{w}_s = \left(\frac{1}{n_s}\right) \sum_{i=1}^{n_s} w_i^s$  adalah rata-rata i-vector untuk setiap speaker,  $n_s$  adalah jumlah suara ucap untuk setiap speaker. Sehingga untuk B menggunakan dekomposisi Cholesky sebagai berikut:

$$W^{-1} = BB' \quad (2.22)$$

## 2.9 Cos Similarity Score (CSS)

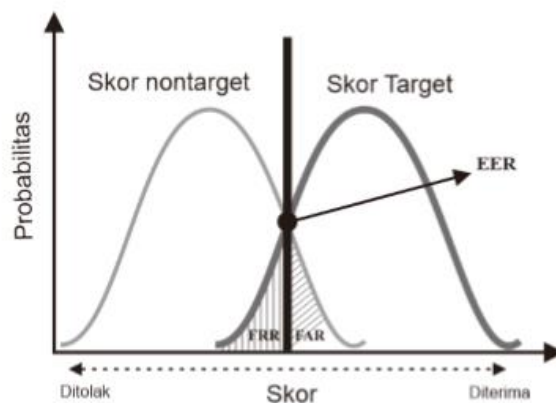
Teknik penilaian yang secara langsung menggunakan nilai kernel cosinus antara speaker target i-vector  $w_{target}$  dan tes i-vector  $w_{test}$  sebagai skor keputusan.

$$score_{(w_{target}, w_{test})} = \frac{\langle w_{target}, w_{test} \rangle}{\|w_{target}\| \|w_{test}\|} \stackrel{\geq}{\leq} \theta \quad (2.23)$$

Nilai kernel ini kemudian dibandingkan dengan ambang batas  $\theta$  untuk mengambil keputusan akhir. Keuntungan dari penilaian ini adalah tidak diperlukan pendaftaran pembicara target, tidak seperti SVM (*support vector machine*) dan analisis JFA klasik, di mana supervektor yang bergantung pada pembicara target perlu

diperkirakan dalam langkah pendaftaran (P. Kenny, 2008). Dalam pemodelan baru ini, analisis faktor memainkan peran ekstraktor fitur dari pada pemodelan efek speaker dan saluran (P. Kenny, 2008) dalam (N. Dehak, 2011). Penggunaan kernel cosinus sebagai skor keputusan untuk verifikasi pembicara membuat proses lebih cepat dan lebih kurang kompleks dari pada metode penilaian JFA lainnya (O. Glembek, 2009).

## 2.10 Indikator Kinerja Sistem



**Gambar 2. 5** Distribusi skor target dan non-target

Indikator kinerja sistem rekognisi pengucap dapat dilihat dari bentuk distribusi skor target saat pengucap *Known* dan *Unknown* sama, dan distribusi skor non-target saat pengucap *Known* dan *Unknown* berbeda. Gambar 2.5 menunjukkan distribusi tersebut. Terdapat satu ukuran performa sistem rekognisi pengucap yang banyak digunakan, yakni nilai Equal Error Rate (EER). Sistem rekognisi pengucap merupakan system binary-classifier atau pemisah kelas biner, dimana hanya terdapat dua macam keluaran atau kelas. Kelas tersebut adalah kelas skor target dan non-target. Seperti pada sistem pemisah kelas biner lainnya, terdapat dua macam kesalahan yang mungkin, yakni kesalahan False Rejection Rate (FRR) dan False Alarm Rate (FAR). FRR menunjukkan probabilitas kesalahan klasifikasi skor target menjadi non-target, sedangkan FAR menunjukkan probabilitas kesalahan klasifikasi skor non-target menjadi target. Nilai EER didapatkan ketika nilai FRR dan FAR sama. Nilai EER merupakan indikator kinerja sistem rekognisi pengucap otomatis yang umum digunakan. Nilai EER menunjukkan performa diskriminasi

antara skor target dan non-target. Nilai EER yang semakin kecil menunjukkan semakin baiknya performa diskriminasi suatu system (Mandasari, et al., 2009).

a. False Acceptance Rate (FAR)

Tingkat penerimaan palsu, atau FAR, adalah ukuran kemungkinan bahwa sistem keamanan biometrik akan secara keliru menerima upaya akses oleh pengguna yang tidak berwenang. Sistem FAR biasanya dinyatakan sebagai rasio jumlah penerimaan palsu dibagi dengan jumlah upaya identifikasi. FAR didefinisikan sebagai persentase dari penipu yang diterima oleh sistem biometrik. Oleh karena itu perlu bahwa persentase ini sekecil mungkin agar orang yang tidak terdaftar dalam sistem tidak boleh diterima oleh sistem. Jadi penerimaan Salah harus diminimalkan.

b. FRR (False Rejection Rate)

Tingkat penolakan palsu adalah ukuran kemungkinan bahwa sistem keamanan biometrik secara keliru akan menolak upaya akses oleh pengguna yang berwenang. FRR sistem biasanya dinyatakan sebagai rasio jumlah penolakan palsu dibagi dengan jumlah upaya identifikasi. FRR didefinisikan sebagai persentase dari pengguna asli ditolak oleh sistem biometrik. Dalam verifikasi biometric sistem pengguna yang berwenang akan membuat klaim identitas mereka dan karenanya sistem harus tidak menolak pengguna terdaftar dan jumlah Penolakan Salah harus dijaga sekecil mungkin. Dengan demikian Penolakan Salah harus diminimalkan.

c. EER (Error Equal Rate)

Equal error rate (EER) adalah algoritma sistem keamanan biometrik yang digunakan untuk menentukan nilai ambang batas untuk tingkat penerimaan palsu dan tingkat penolakan salahnya. Didefinisikan sebagai titik persimpangan pada grafik di mana kurva FAR dan FRR diplot. Nilai menunjukkan bahwa proporsi penerimaan palsu sama dengan proporsi penolakan palsu. Semakin rendah nilai EER, semakin tinggi akurasi sistem biometrik.

*Halaman ini sengaja dikosongkan*

## BAB III

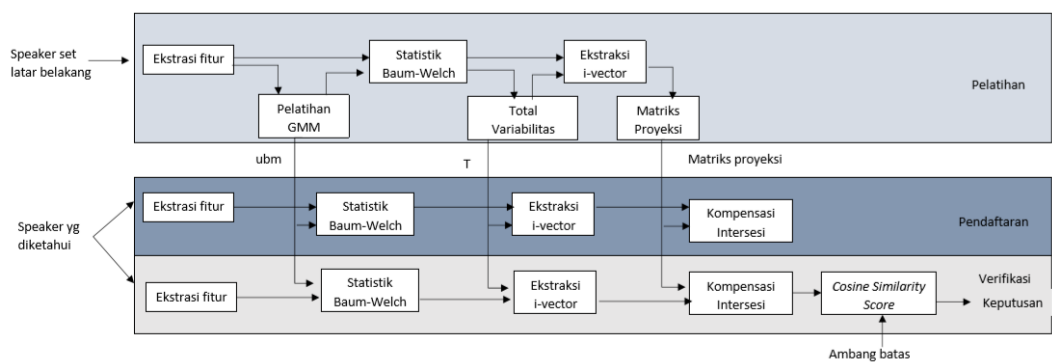
### METODOLOGI PENELITIAN

#### 3.1 Studi Literatur dan Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data yang diperlukan untuk pengerjaan tugas akhir ini. Data yang digunakan merupakan data sekunder berupa data suara dengan format .wav yang berasal dari dataset Graz (Anon., n.d.) dan dataset Bahasa Indonesia (Anggraini, 2013). Dalam dataset graz terdapat 2 jenis data yaitu dataset *clean* dan dataset LAR (dataset dengan penambahan noise *reverberant*) dalam 20 penutur (10 perempuan dan 10 laki-laki). Sedangkan dataset Indonesia berisi data *clean* dengan 8 penutur (4 perempuan dan 4 laki-laki). Kemudian dilakukan identifikasi permasalahan dan pemahaman teori dengan mencari dan mengumpulkan referensi penunjang mengenai forensik suara ucap, fitur penciri dari suara dan metode yang akan digunakan yaitu *joint factor analysis* dan *i-vector*, serta performansi sistem biometric untuk sistem forensik. Referensi yang digunakan berupa buku-buku literatur, jurnal ilmiah, serta artikel internet yang relevan

#### 3.2 Perancangan Program

Pada tahap ini dilakukan perancangan program yang dibuat menggunakan platform pemrograman pada Matlab R2020A. Proses perancangan program dapat disajikan menggunakan blok diagram pada gambar 3.1 serta akan diberikan penjelasan pada setiap tahapan:



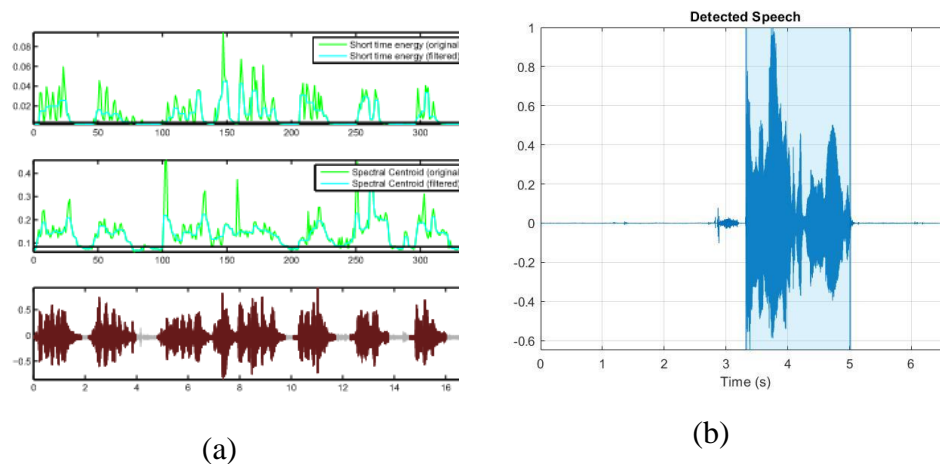
**Gambar 3. 1** Blok diagram proses perancangan program

Dalam simulasi forensik dataset yang berisi file suara ucap akan dibagi menjadi dataset latar belakang (*background dataset*) dan dataset uji. Dataset uji digunakan sebagai data target speaker dan data tes. Data target speaker didefinisikan dari data pembicara yang telah dikenali / *known speaker* yang telah diberi id / label sebagai *suspect speaker*. Sedangkan data tes terdefinisi sebagai *unknown speaker*, yang kemudian akan dibandingkan dengan data target speaker untuk mencari keidentikan kedua pembicara tersebut. Proses dalam identifikasi suara dibagi menjadi 3 tahapan utama yakni *training*, *enrollment* dan verifikasi. Seperti yang terdapat pada blok diagram proses yang ditunjukkan oleh gambar 3.1, setiap tahapan utama dalam sistem identifikasi dikenakan proses yang sama yaitu pengambilan fitur melalui ekstraksi fitur, pelatihan UBM-GMM, perhitungan statistik baum-welch, perentuan total variabilitas ruang, ekstraksi i-vector, perhitungan matriks proyeksi, kompensasi intersesi dan verifikasi menggunakan *cosine similarity score*.

### 3.2.1 Ekstraksi Fitur

Proses awal berupa ekstraksi fitur yaitu pengambilan fitur MFCC, delta MFCC, dan delta-delta MFCC yang memuat informasi berupa fitur penciri dari sinyal suara-ucap yang akan dianalisa. Proses ekstraksi fitur dilakukan dengan Langkah sebagai berikut:

- a. Inisiasi data audio  
Data audio merupakan kolom vektor dari audio data yang berasal dari file suara yang akan diekstraksi.
- b. Mengganti nilai NaN dari data audio menjadi 0  
Jika terdapat nilai NaN (tidak terdefinisi) pada audioData maka proses ekstraksi fitur tidak dapat dijalankan
- c. Mengambil daerah voice (*speech segment*)  
*Function detectSpeech* pada matlab digunakan untuk menentukan *speech segment* dari sinyal suara audio. Tahap ini melibatkan beberapa proses yaitu penentuan *signal energy* dan *spectral centroid* menggunakan STFT (*short time foriuer transform*) dengan hanning window dan Panjang overlap 50 ms (Giannakopoulos, 2009).

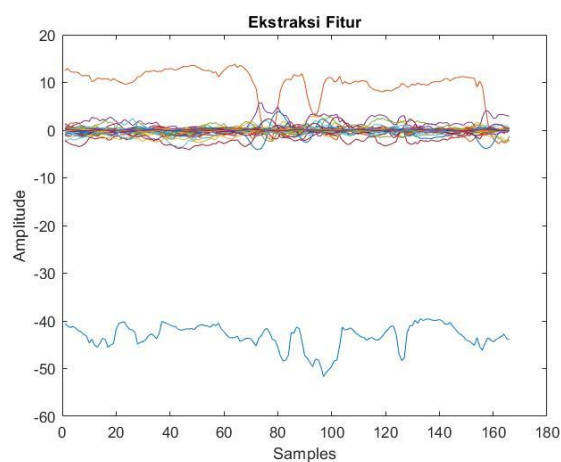


**Gambar 3. 2** (a) Ilustrasi *Speech Segment* Dari (Giannakopoulos, 2009), (b) Hasil Salah Satu Deteksi *Voice Segment* Pada Sinyal Audio.

Gambar 3.2 (a) Menunjukkan Sub-Gambar Pertama Adalah *Sequence Of Signal Energy*, Sub-Gambar Kedua Adalah *Ceptral Centroid Sequence* Beserta Thresholdnya, Dan Sub-Gambar Ketiga Adalah Sinyal Audio Dengan Garis Coklat Mengindikasikan *Voice Segment* Yang Terdeteksi.

#### d. Ekstraksi Fitur

*Function audioFeatureExtractor* pada matlab digunakan untuk mengekstraksi fitur dari sinyal suara audio. Tahap ekstraksi fitur dilakukan dengan ekstrak fitur menggunakan 25 ms analisa *Hanning Windows* dengan 10 ms overlap dan *window length* 9. Ekstraksi fitur dilakukan untuk mengambil 20 *dimension feature* mfcc pada setiap sinyal audio suara. Ditambah dengan koefisien delta dan delta-delta mfcc, maka terbentuk 60 *dimension feature*.



**Gambar 3. 3** Hasil Ekstraksi Fitur Dari Sinyal Suara Audio

Gambar 3.3 menunjukkan hasil ekstraksi fitur dari sinyal audio dari gambar 3.2 (b) yang telah diambil voice segmentnya saja.

e. Normalisasi Fitur

Terdapat 2 parameter dalam normalisasi fitur yaitu factor normalisasi rata-rata (*mean-normFactor*) dan factor normalisasi standar deviasi (*STD-normfactor*) yang didapatkan dari perhitungan pada seluruh hasil pengolahan fitur. Kemudian kedua parameter tersebut dikenakan kepada fitur yang telah diekstraksi untuk menormalisasi fitur yang telah didapatkan.

### 3.2.2 Pelatihan UBM-GMM

Sistem yang di *training* dengan kumpulan data TIMIT biasanya berisi sekitar 2048 komponen ubm. Namun dalam hal ini hanya memakai 256 komponen. Pada tahap ini langkah pertama identik dengan langkah ekspektasi dari algoritma EM, di mana perkiraan statistik yang memadai dari data pelatihan pembicara dihitung untuk setiap campuran dalam UBM. dimana  $p(x_t | \lambda)$  adalah fungsi posterior likelihood yang dapat dihitung menggunakan persamaan 2.2 (Reynolds, et al., 2000). Pada bagian *Expectation* terdapat sub-proses dalam sebagai berikut:

a. Menghitung posterior log likelihood

Persamaan 2.2 digunakan untuk menghitung *mixture density* untuk  $D$ -dimensional fitur vector ( $x$ ). Sedangkan densitas Gaussian  $M$  unimodal,  $\pi_i(x)$ , masing-masing parameter dengan rata-rata vektor  $D \times 1$  ( $\mu_i$ ) dan matriks kovarians  $D \times D$  ( $\Sigma_i$ ) dapat menggunakan persamaan 2.3 (Reynolds, et al., 2000). *Fuction helperGMMLogLikelihood* dibuat untuk menghitung posterior loglikelihood seperti yang ditunjukkan pada persamaan tersebut. Terdapat 2 masukan pada proses ini yakni  $Y$  yaitu fitur suara yang telah diekstraksi dan dinormalisasi dan ubm. Formula tersebut diinisiasi kemudian dipakai untuk mengolah masukan menjadi keluaran berupa posterior loglikelihood.

b. Menghitung probabilitas normalisasi posterior

Umumnya, vektor fitur  $X$  diasumsikan independen, jadi loglikelihood model  $\lambda$  untuk urutan vektor fitur,  $X = \{x_1, \dots, x_T\}$ , dihitung menggunakan persamaan 2.4 (Reynolds, et al., 2000). Tahap ini nilai loglikelihood model  $\lambda$



seperti pada persamaan diatas, dimana di dalam program ini diinisiasi menggunakan variabel  $L$ . Untuk menggambarkan proses perolehan nilai  $L$  maka akan diberikan permisalan dengan menggunakan perhitungan sederhana. Variabel *loglikelihood* berupa matriks dari posterior log likelihood berukuran  $256 \times 166$  yang didapatkan dari proses sebelumnya. Contoh penyederhanaan proses perhitungan probabilitas normalisasi posterior:

- Permisalan matriks

misalkan matriks *loglikelihood* berupa matriks ukuran  $2 \times 3$ :

$$\begin{bmatrix} 1 & 0 & 2 \\ -1 & 5 & 0 \end{bmatrix}$$

- Kemudian untuk memperoleh *logLikelihoodSum*

Mencari nilai maksimum dari setiap kolom pada matriks *loglikelihood* menggunakan *function max* pada matlab pada setiap nilai kolom.

$$\begin{bmatrix} 1 & 0 & 2 \\ -1 & 5 & 0 \end{bmatrix} \rightarrow [1 \quad 5 \quad 2]$$

Sehingga menghasilkan matrik baru berupa matriks  $[1 \quad 5 \quad 2]$ .

- Mencari hasil dari logaritmik

Yaitu logaritmik jumlah dari  $\exp$  (matriks *loglikelihood* – matriks *matrikBaru*).

$$\begin{aligned} \log \sum \exp\left(\begin{bmatrix} 1 & 0 & 2 \\ -1 & 5 & 0 \end{bmatrix} - [1 \quad 5 \quad 2]\right) \\ = \log \sum \exp\left(\begin{bmatrix} 0 & -5 & 0 \\ -2 & 0 & -2 \end{bmatrix}\right) \\ = [0.1269 \quad 0.0067 \quad 0.1269] \end{aligned}$$

Sehingga menghasilkan matriks  $[0.1269 \quad 0.0067 \quad 0.1269]$

Kemudian matriks tersebut ditambahkan dengan matriks baru

$[1 \quad 5 \quad 2]$  hasil

$$[0.1269 \quad 0.0067 \quad 0.1269] + [1 \quad 5 \quad 2] =$$

$$[1.1269 \quad 5.0067 \quad 2.1269]$$

- $L$  merupakan hasil dari penjumlahan semua nilai pada matriks *logLikelihoodSum*, sehingga nilai  $L$  diperoleh sebesar 8.2606

Sedangkan pada simulasi sebenarnya dengan ukuran matriks *loglikelihood* lebih besar dan nilainya mengikuti hasil pengolahan sebelumnya, didapatkan nilai L sebesar  $-1.0999e+07$ .

Sedangkan langkah kedua dari algoritma EM yaitu *maximization* digunakan untuk adaptasi perkiraan *sufficient statistic* baru kemudian digabungkan dengan *old sufficient statistics* dari *UBM mixture parameters* menggunakan *data-dependent mixing coefficient*. Dengan diberi UBM dan vektor pelatihan dari pembicara yang dihipotesiskan,  $X = \{x_1, \dots, x_T\}$ , Menentukan penyesuaian probabilitas vektor pelatihan ke dalam komponen campuran yaitu untuk campuran  $i$  di UBM, dapat dihitung dengan persamaan 2.5.

Dalam  $w_i p_i(x_t)$  didapatkan dari nilai maksimum hasil dari pembagian matriks hasil dari orde 0 (N) pada statistik baum-welch dengan jumlahnya. Matriks N pada program memuat 560 cell (banyaknya data training) setiap cell berupa matriks berukuran  $1 \times 15360$ .

Contoh penyederhanaan proses perhitungan  $w_i p_i(x_t)$ :

$$N = [1 \quad 2 \quad 3]$$

Jumlah dari matriks N tersebut adalah 6. Hasil pembagiannya matriks N terhadap jumlah matriksnya adalah :

$$[1/6 \quad 2/6 \quad 3/6] = [0.1667 \quad 0.3333 \quad 0.5000]$$

Sehingga  $w_i p_i(x_t)$  adalah 0.5000, perhitungan yang sama dilakukan untuk semua matrix cell matriks N.  $\Pr(i|x_t)$  juga dapat ditentukan dengan membagi matriks  $w_i p_i(x_t)$  terhadap jumlahnya. Parameter lainnya seperti *ubm.mu* (*ubm komponen rata-rata / component means*) didapatkan dari hasil pembagian matriks F (orde 1 statistik baum-welch) dan N (orde 0), sehingga hasil kalkulasinya berupa *ubm.mu* dengan matriks berukuran  $60 \times 256$ , sedangkan *ubm.sigma* (*component covariance matrices*) didapatkan dari  $\frac{S}{N} - \text{ubm.mu}^2$ , S merupakan orde ke-2 dari statistika baum-welch. Oleh karena itu *ubm.sigma* yang dihasilkan berupa matriks berukuran ( $60 \times 256$ ).

### 3.2.3 Statistik Baum-Welch

Statistik Baum-Welch digunakan untuk mengkompensasi variabel tersembunyi tersebut. Statistik Baum-Welch adalah statistik  $N$  (orde ke 0) dan  $F$  (orde pertama) yang digunakan pada algoritma EM, dapat dihitung menggunakan persamaan 2.14 dan 2.15 sebagai berikut:

$$Nc = \sum_{t=1}^L P(c|y_t, \Omega) = \sum_t \gamma_t(c)$$

$$Fc = \sum_{t=1}^L P(c|y_t, \Omega) y_t = \sum_t \gamma_t(c) y_t$$

Dengan  $c = 1, \dots, C$  adalah indeks Gaussian dan  $P(c|y_t, \Omega)$  sesuai dengan probabilitas posterior komponen campuran  $c$  menghasilkan vektor  $y_t$ .

Gamma  $\gamma_t(c)$  dihasilkan dari  $\exp(\text{loglikelihood} - \text{loglikelihoodSum})$ , loglikelihood dan loglikelihoodSum merupakan variabel hasil dari perhitungan posterior log likelihood pada Langkah ekspektasi ubm.  $Nc$  adalah jumlah dari gamma, sedangkan  $Fc$  merupakan hasil dari perkalian gamma dan  $Y$  (hasil ekstrak fitur dari file audio)

### 3.2.4 Total Variabilitas Space

*Total variability space* ( $T$ ) yang terdapat dalam rumusan persamaan 2.13. Ekstraksi  $i$ -vector dicirikan oleh  $m$  sebagai rata-rata supervector (ubm),  $T$  adalah *low rank* matriks total variability,  $\Sigma$  adalah diagonal matriks kovarian. Berikut tahapan untuk mencari *Total variability space* :

- Menghitung distribusi posterior  $l(u)$  dari variable tersembunyi  $w(u)$  dengan persamaan

$$l(u) = I + T^t \Sigma^{-1} N(u) T$$

Dimana,  $N(u)$  adalah orde 0 *baum-welch statistic*,  $I$  adalah matriks identitas,  $T$  dan  $\Sigma$  total variability dan matriks kovarian atau ubm.sigma.

- Mengakumulasi statistik pada seluruh speaker, diekspresikan dengan persamaan

$$E[w(u)] = l^{-1}(u) T^t \Sigma^{-1} \tilde{F}(u)$$

$$C = \sum_u \tilde{F}(u) E[w'(u)]$$

$$A_c = \sum_u N_c(u) l_T^{-1}(u)$$

$\tilde{F}(u)$  adalah *centralized first order statistic*,  $l^{-1}(u)$  adalah matriks kovarian yang dihasilkan dari invers matriks  $l(u)$ , T dan  $\Sigma$  total variability dan matriks kovarian,  $A_c$  adalah akumulasi statistic untuk seluruh speaker.

c. Total variabilitas ruang, dapat diekspresikan melalui persamaan berikut:

$$T_c = A_c^{-1} C$$

$$T = \begin{bmatrix} T_1 \\ \vdots \\ T_c \end{bmatrix}$$

$A_c^{-1}$  adalah invers matriks dari  $A_c$ . Input dari proses total variabilitas ruang berupa ubm.  $\Sigma$  dari UBM variance sesuai dengan persamaan berikut:

$$\Sigma = N^{-1} \left( \sum_u \tilde{S}(u) - \text{diag}(CT^t) \right)$$

$\tilde{S}(u)$  adalah *centralized second order statistic baum-welch*,  $c = 1, \dots, c$  adalah jumlah dari komponen ubm, umumnya pada data TIMIT digunakan sebanyak 1000 namun pada program ini digunakan sebanyak 256 komponen. Kemudian melalui Langkah-langkah diatas hingga diperoleh matriks T sebagai matriks total variabilitas ruang dari data *training* yang juga digunakan pada tahap selanjutnya yaitu tahap pendaftaran (*enrolment*) dan verifikasi.

### 3.2.5 Ekstraksi i-vector

Setelah mendapatkan hasil dari matriks total variabilitas ruang (T), maka secara matematis i-vector dapat dihitung menggunakan persamaan 2.17. Untuk menginverskan matriks hasil dari perhitungan  $(I + T^t \Sigma^{-1} N(u) T)$  pada persamaan tersebut digunakan *function pinv* sebagai pseudo-invers agar dapat menginverskan matriks non-square. Sedangkan untuk operasi matriks *transpose* pada matlab dapat dioperasikan menggunakan *function (')*. Sehingga didapatkan model i-vector untuk data training. Tahap ini melibatkan operasi matematika sederhana berupa penambahan dan perkalian matriks.

### 3.2.6 Matriks Proyeksi

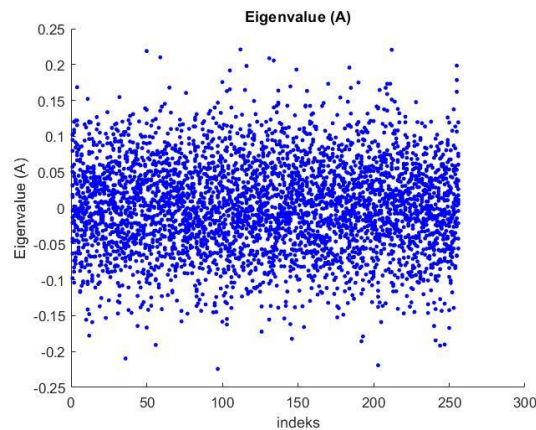
Tahap ini digunakan untuk mengkompensasi intersesi dalam proses identifikasi suara. Data masukan untuk proses ini adalah  $w_{bar}$  sebagai rata-rata i-

vector untuk setiap speaker. Dalam hal ini digunakan LDA dan WCNN sebagai *projection matrix* pada simulasi ini.  $\bar{w}$  sebagai rata-rata i-vector pada seluruh speaker dan jumlah ujaran untuk setiap speaker. Matriks LDA bisa didapatkan dari persamaan 2.18 dan 2.19.

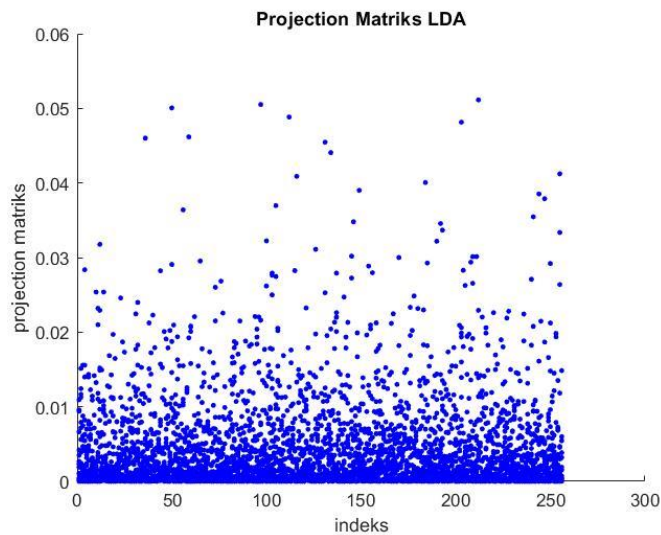
Tahap ini melibatkan operasi matematika berupa sigma ( $\Sigma$ ), di dalam operasi matlab dapat menggunakan *function sum*, matriks transpose dalam matlab bisa langsung memakai ( $'$ ), selebihnya berupa pertambahan dan perkalian matriks. Persamaan 2.20 digunakan untuk mencari eigenvalue untuk eigenvectors. Jumlah eigen vector yang digunakan adalah 16. Untuk mendapatkan eigenvalue tersebut dapat digunakan *function eigs* pada matlab dengan memasukkan hasil kalkulasi matriks  $S_b$ ,  $S_w$ , dan jumlah eigen value menghasilkan sebuah matriks A berukuran  $256 \times 16$ . Sebelum dikalikan dengan projection matriks, matriks eigen A dibagi oleh vector wise-norm matriks tersebut lalu ditranpos. Vector wise-norm pada matlab dapat menggunakan *function vecnorm*, vector wise-norm menghitung Euclidian-norm dari matriks tersebut, secara matematis mempunyai formula:

$$\|v\| = \sqrt{\sum_{k=1}^N |v|^2}$$

misal terdapat matriks  $x = [2 \ 2 \ 2]$  maka *vecnorm* dari  $x$  adalah  $\sqrt{12} = 3.4641$ , *vecnorm* dari matriks eigen A ( $256 \times 16$ ) didapatkan dengan menghitung Euclidian-norm pada setiap kolomnya menghasilkan matriks ukuran  $1 \times 16$ . Sehingga matriks eigen yang baru berupa matriks A ( $16 \times 256$ ) seperti pada gambar 3.4. Kemudian didapatkan matriks projection yang merupakan hasil perkalian eigenvalue A ( $16 \times 256$ ) dengan inisiasi matriks projection awal yang berupa matriks identitas yang memiliki ukuran sama seperti matriks  $w$  (i-vector) yaitu  $16 \times 16$  sehingga menghasilkan matriks projection dari LDA dengan ukuran  $16 \times 256$  seperti pada gambar 3.5.



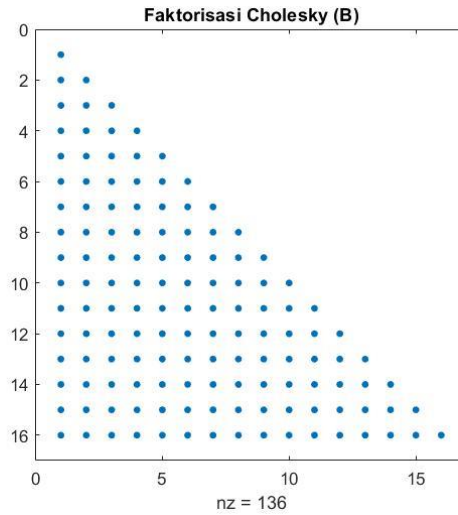
**Gambar 3. 4** Eigenvalue LDA Projection Matriks LDA



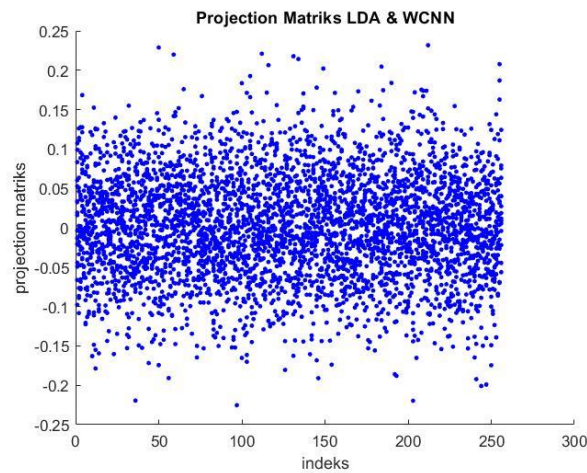
**Gambar 3. 5** Projection Matriks LDA

Sedangkan WCNN digunakan untuk menskala ruang i-vector yang berbanding terbalik dengan *in-class covariance*, sehingga arah variabilitas *intra-speaker* yang tinggi tidak ditekankan pada i-vector. Untuk mencari B yang merupakan eigenvalue dari Matriks WCNN bisa didapatkan dengan persamaan 2.22 dekomposisi Cholesky.  $W^{-1}$  adalah invers dari matriks matriks WCNN (W), untuk mendapatkan matriks WCNN (W) diinisiasi berupa matriks *square* bernilai 0 (zero) yang memiliki ukuran matriks sama seperti banyak kolom pada projection matriks hasil dari LDA (256x16) yaitu W(16x16) kemudian dihitung menggunakan persamaan 2.21. B merupakan hasil dari faktorisasi Cholesky, dalam matlab untuk mengkalkulasi faktorisasi tersebut dapat menggunakan *function chol* menghasilkan matriks B (16x16) seperti pada gambar 3.6. Sehingga projection matriks LDA dan

WCNN didapatkan dari perkalian matriks  $B(16 \times 16)$  dan matriks projection LDA( $16 \times 256$ ) menjadi projection LDA dan WCNN ( $16 \times 256$ ) seperti pada gambar 3.7.



**Gambar 3. 6** Hasil Faktorisasi Cholesky B



**Gambar 3. 7** Projection Matriks Gabungan LDA dan WCNN

Dari proses ini didapatkan matriks proyeksi dari tahap training yang juga digunakan pada tahap selanjutnya yaitu tahap pendaftaran (*enrolment*) dan verifikasi.

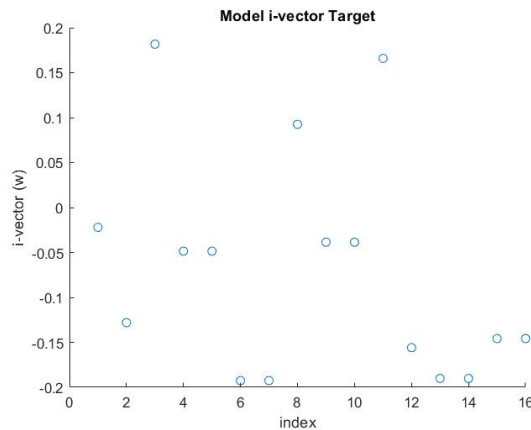
### 3.2.7 Tahap Pendaftaran

Tahap ini adalah tahapan untuk mendaftarkan (*enroll*) *speaker* baru yang belum ada pada *training dataset* dan memberikan label setiap *speaker* sesuai dengan ID *speaker* masing masing. Pertama, Pisahkan objek data audio pada variable

*adsEnrollAndVerify* menjadi data pendaftaran dan data untuk verifikasi. Pada Langkah ini 20 ucapan setiap speaker digunakan untuk pendaftaran (*enrollment*). Membuat i-vektor pada setiap file untuk masing-masing pembicara dalam set pendaftaran (*enroll*) menggunakan urutan langkah-langkah ini:

- a. Ekstraksi fitur
- b. Statistik Baum-Welch: Menentukan statistik nol dan statistik orde pertama
- c. Ekstraksi i-vektor
- d. Kompensasi intersersi / *matriks projection LDA* dan *WCNN*

Model i-vector untuk setiap speaker pada data enrollment atau juga disebut sebagai model i-vector target berupa matriks ukuran 16x1 seperti yang ditunjukkan pada gambar 3.8.

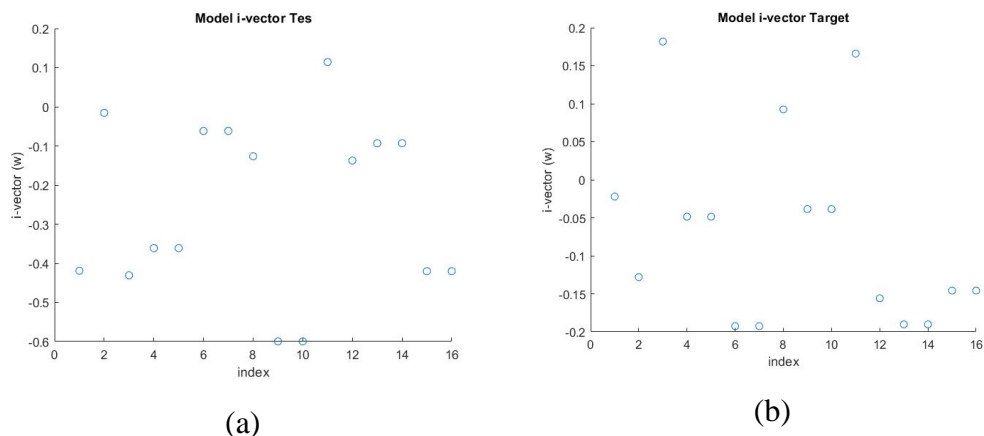


**Gambar 3. 8** Salah Satu Model I-Vector Dataset *Enrollment* Dataset Indonesia (*Clean Data*)

### 3.2.8 Tahap Verifikasi

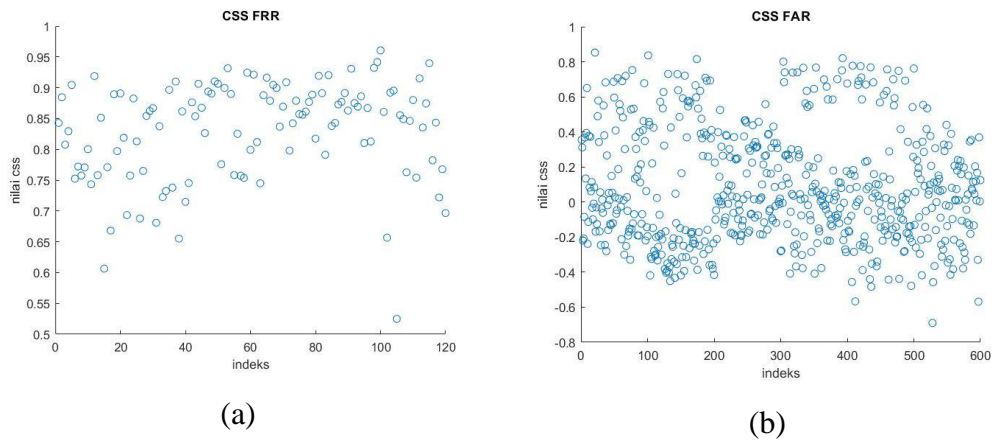
Langkah klasifikasi dilakukan dengan mengukur kesamaan model i-vector untuk data tes atau disebut model i-vector tes. Kemudian model i-vector ini akan dibandingkan dengan model i-vector target dari model i-vector *enrollment*. Untuk mengukur kedua model *vector* tersebut digunakan CSS (*cosine similarity score*). CSS menggunakan persamaan 2.23, nilai CSS memiliki rentang nilai -1 hingga 1.





**Gambar 3. 9** (a) Salah Satu Model I-Vector Dataset Uji, (b) Salah Satu Model i-Vector Dataset Enrollment Yang Akan Saling Dibandingkan Dataset Indonesia (Clean Data)

Tahap verifikasi dilakukan dengan 2 jenis uji yakni menguji *speaker* yang sama antara data tes terhadap target dan menguji *speaker* berbeda antara data tes terhadap target. Data target speaker terinisiasi kedalam tahap verifikasi berupa data yang telah di urutkan berdasarkan id penutur (6 penutur), pada matlab dapat menggunakan *function unique*. Sedangkan data tes yang diambil bergantung pada jenis uji. Pada pengujian penutur (*speaker*) sama, data tes yang diujikan berasal dari id penutur yang sama untuk setiap 20 file suara. CSS FRR merupakan hasil dari tahap verifikasi untuk menguji *speaker* yang sama sehingga banyaknya jumlah pengujian memiliki total 120 untuk semua penutur. Pada pengujian penutur (*speaker*) berbeda, data tes diambil secara acak diluar data target *speaker* namun masih didalam lingkup id penutur pada data target speaker sebanyak 100 file suara. CSS FAR merupakan hasil dari tahap verifikasi untuk menguji *speaker* secara acak sehingga banyaknya jumlah pengujian memiliki total 600 untuk semua penutur. Pada gambar 3.10 (a) CSS FRR menunjukkan hasil pengujian *speaker* yang sama berada pada sebaran data css diatas 0.6, pada tabel 3.1 ditampilkan sebagian dari hasil verifikasi pengujian *speaker* yang sama. Sedangkan untuk gambar 3.10 (b) CSS FAR menunjukkan hasil pengujian *speaker* secara acak memiliki sebaran data css diatas merata pada rentang nilai css, pada tabel 3.2 ditampilkan sebagian dari hasil verifikasi pengujian *speaker* secara acak.



**Gambar 3. 10** (a) Hasil Scattering Coss FRR, (b) Hasil Scattering Coss FAR Dari Dataset Graz (Clean Data)

**Tabel 3. 1** Pengujian Speaker Sama Dataset Graz (Clean Data)

Target	Test	Speakeridx	Target	Test	Speakeridx
'F01'	'F01'	0,8429	'M01'	'M01'	0,9212
'F01'	'F01'	0,8846	'M01'	'M01'	0,8118
'F01'	'F01'	0,8076	'M01'	'M01'	0,7450
'F01'	'F01'	0,8294	'M01'	'M01'	0,8879
'F01'	'F01'	0,9045	'M01'	'M01'	0,9163
'F01'	'F01'	0,7523	'M01'	'M01'	0,8788
'F01'	'F01'	0,7721	'M01'	'M01'	0,9047
'F01'	'F01'	0,7574	'M01'	'M01'	0,8995
'F01'	'F01'	0,7707	'M01'	'M01'	0,8368
'F01'	'F01'	0,8003	'M01'	'M01'	0,8693
..	..	...	..	..	...
'F02'	'F02'	0,8189	'M02'	'M02'	0,9193
'F02'	'F02'	0,6933	'M02'	'M02'	0,8911
'F02'	'F02'	0,7573	'M02'	'M02'	0,7911
'F02'	'F02'	0,8825	'M02'	'M02'	0,9203
'F02'	'F02'	0,8129	'M02'	'M02'	0,8378
'F02'	'F02'	0,6877	'M02'	'M02'	0,8429
'F02'	'F02'	0,7649	'M02'	'M02'	0,8728
'F02'	'F02'	0,8542	'M02'	'M02'	0,8770
'F02'	'F02'	0,8624	'M02'	'M02'	0,8914
'F02'	'F02'	0,8669	'M02'	'M02'	0,8629
..	..	...	..	..	...
'F05'	'F05'	0,7453	'M05'	'M05'	0,8607
'F05'	'F05'	0,8763	'M05'	'M05'	0,6567
'F05'	'F05'	0,8537	'M05'	'M05'	0,8920
'F05'	'F05'	0,9063	'M05'	'M05'	0,8953
'F05'	'F05'	0,8674	'M05'	'M05'	0,5251

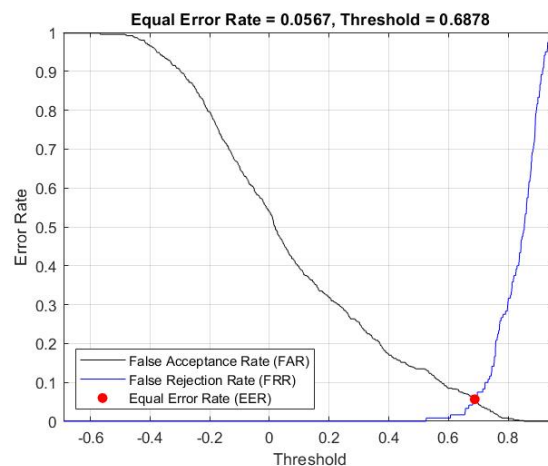
'F05'	'F05'	0,8263	'M05'	'M05'	0,8554
'F05'	'F05'	0,8939	'M05'	'M05'	0,8491
'F05'	'F05'	0,8904	'M05'	'M05'	0,7629
'F05'	'F05'	0,9102	'M05'	'M05'	0,8463
'F05'	'F05'	0,9063	'M05'	'M05'	0,8801
..	..	...	..	..	...

**Tabel 3. 2** Pengujian Speaker Acak Dataset Graz (Clean Data)

Target	Test	Speakeridx	Target	Test	Speakeridx
'F01'	'F05'	0,3548	'M01'	'F01'	0,2707
'F01'	'F05'	0,3113	'M01'	'F05'	0,1057
'F01'	'M02'	-0,2193	'M01'	'F05'	0,2464
'F01'	'M02'	-0,2082	'M01'	'M02'	-0,4081
'F01'	'M02'	-0,0847	'M01'	'M02'	-0,0555
'F01'	'F05'	0,3680	'M01'	'M02'	0,3015
'F01'	'M05'	0,1341	'M01'	'F05'	0,0453
'F01'	'F05'	0,3941	'M01'	'M05'	0,7386
'F01'	'F02'	0,6504	'M01'	'F05'	0,0356
'F01'	'M01'	-0,2385	'M01'	'F02'	-0,2509
..	..	...	..	..	...
'F02'	'F01'	0,8368	'M02'	'F01'	-0,1071
'F02'	'F05'	0,4576	'M02'	'F05'	-0,1633
'F02'	'F05'	0,4398	'M02'	'F05'	-0,1696
'F02'	'M02'	-0,4187	'M02'	'F05'	0,0746
'F02'	'M02'	-0,3127	'M02'	'M05'	0,1590
'F02'	'M02'	-0,3249	'M02'	'F05'	-0,0791
'F02'	'F05'	0,2203	'M02'	'F02'	-0,4563
'F02'	'M05'	-0,1482	'M02'	'M01'	0,7780
'F02'	'F05'	0,4245	'M02'	'M05'	0,1830
'F02'	'M01'	-0,2788	'M02'	'F02'	-0,2844
..	..	...	..	..	...
'F05'	'F01'	0,2249	'M05'	'F01'	-0,1303
'F05'	'M02'	-0,1000	'M05'	'F05'	0,0573
'F05'	'M02'	-0,0476	'M05'	'F05'	-0,2591
'F05'	'M02'	-0,2110	'M05'	'M02'	0,3265
'F05'	'M05'	0,0263	'M05'	'M02'	0,2374
'F05'	'F02'	0,3988	'M05'	'M02'	0,1634
'F05'	'M01'	0,0042	'M05'	'F05'	0,1944
'F05'	'M05'	0,1441	'M05'	'F05'	-0,0869
'F05'	'F02'	0,5251	'M05'	'F02'	-0,2955
'F05'	'M05'	0,0033	'M05'	'M01'	0,1301
..	..	...	..	..	...

### 3.3 Simulasi Program dan Pengujian

Simulasi program dilakukan dengan menggunakan data suara sekunder sesuai dengan program yang telah dirancang. Pada dataset Graz dilakukan simulasi dataset utuh yakni tanpa mengubah atau menambah noise lainnya. Karena didalam dataset Graz terdapat dataset *clean* dan dataset LAR yaitu dataset dengan pendekatan simulasi forensik dengan ditambahkan *noise reverberant* pada file suara dataset tersebut. Sedangkan untuk dataset Indonesia akan disimulasikan dengan pendekatan forensik seperti gangguan oleh suara pembicara lain seakan akan suara yang diuji berada ditengah keramaian orang dilakukan dengan penambahan *babble noise* atau disebut sebagai babble data. Untuk melihat performansi keakuratan program yang telah dirancang. Performansi program dapat dilihat dari nilai EER (Error Equal Rate) dari setiap pengujian. EER terbentuk dari hasil perpotongan FRR dan FAR. FRR didapatkan dari css FRR.



**Gambar 3. 11** Hasil FRR, FAR, dan EER

Performansi ditunjukkan dari  $100\% - \text{EER} (\%)$ , sehingga pada salah satu hasil pengujian pada gambar menunjukkan EER sebesar 5.6%, sehingga performansinya sebesar 94.4%.

### 3.4 Penarikan Kesimpulan

Pada tahap ini akan dilakukan penarikan kesimpulan dari hasil pembahasan sebelumnya dan juga akan diberikan saran mengenai hal-hal yang dapat dikembangkan untuk penelitian selanjutnya

### **3.5 Pembuatan Laporan**

Bagian terakhir dalam tugas akhir ini adalah pembuatan laporan seluruh tahapan atau proses yang sudah dilakukan.

*Halaman ini sengaja dikosongkan*

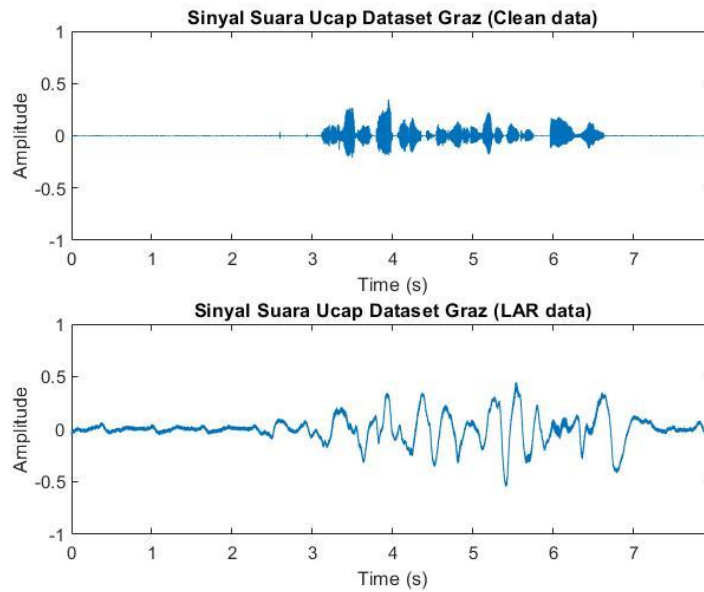
## **BAB IV**

### **HASIL DAN PEMBAHASAN**

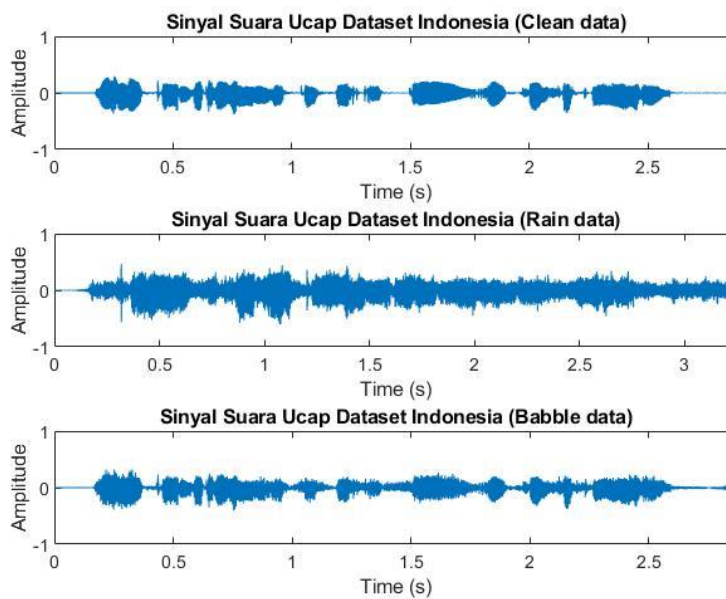
Berdasarkan penelitian yang telah dilakukan diperoleh hasil sebagai berikut:

#### **4.1 Data Uji Coba**

Uji coba program dalam tugas akhir ini dilakukan terhadap file audio wav. Audio uji coba berupa data sekunder yang diambil dari dataset Graz dan dataset Indonesia. Dataset Graz memiliki 2 jenis file uji yakni dataset file suara bersih (*clean dataset*) dan dataset suara dengan campuran *noise reverberant (LAR dataset)*. Dalam dataset Graz terdapat 20 penutur yang terbagi atas 10 penutur perempuan dan 10 penutur laki-laki. Tersedia file audio sebanyak 236 file setiap penutur dalam setiap jenis file. Sedangkan untuk dataset Indonesia berupa dataset *clean* tanpa adanya campuran *noise*. Dataset Indonesia memiliki 8 penutur yang terbagi atas 4 penutur perempuan dan 4 penutur laki-laki. Pelabelan file dilakukan dengan formasi (jenis file)-(gender Female/Male dan urutannya)-(kode penutur berupa nama singkatan)-(nomer file suara dengan kalimat urutan dalam database) misalnya CLEAN-F01-fala-001. Dalam simulasi, pengujian untuk dataset Graz akan dilakukan utuh tanpa adanya penambahan noise apapun. Namun, untuk dataset Indonesia akan dilakukan penambahan gangguan berupa noise babble dan noise berupa suara hujan sebagai pendekatan situasi dalam proses analisa forensik suara ucap. Noise suara hujan didapatkan dari hasil rekaman suara dengan durasi sepanjang 30 detik. Sedangkan noise babble merupakan gangguan suara yang dibuat dengan menggabungkan beberapa sinyal suara dari beberapa penutur yang berbeda. Dalam hal ini noise babble dibuat dengan menggabungkan 3 sinyal suara dari penutur yang berbeda.



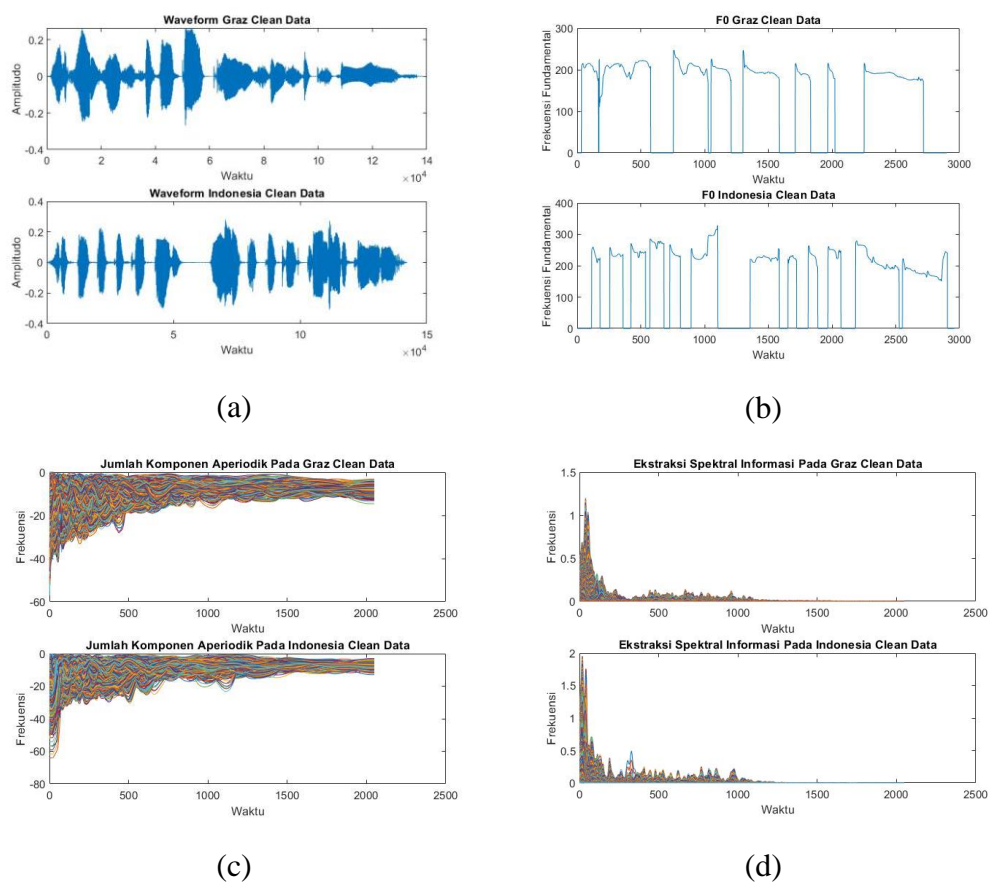
**Gambar 4. 1** Sinyal Suara Ucap Dataset Graz Pada Clean Data Dan LAR Data



**Gambar 4. 2** Sinyal Suara Ucap Dataset Indonesia Pada Clean Data, Rain Data Dan Data Babble

Dari gambar 4.1 dan 4.2 diatas dapat dilihat jenis dataset suara dan perbedaannya melalui plot sinyal suara waveform dari masing masing jenis dataset suara. Penggunaan jenis dataset tersebut diperuntukkan untuk menguji performansi dari program yang dibuat, sekaligus mengetahui keandalan terhadap beberapa jenis data dalam banyak kondisi.





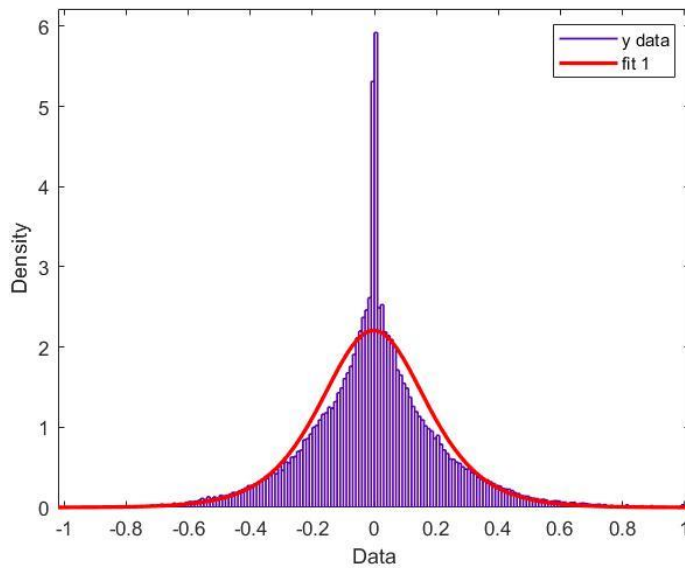
**Gambar 4.3** Data Intrinsic Suara Clean Data Dataset Indonesia Dan Graz Pada (a) Waveform, (b) Fundamental Frequency, (c) Jumlah Komponen Aperiodik, (d) Ekstraksi Spectral

Jika ditelisik dari data intrinsic kedua dataset melalui perbandingan data clean didapatkan hasil seperti yang ditunjukkan pada gambar 4.3. Dapat dilihat dari *spectral* dan jumlah komponen aperiodic kedua dataset memiliki distribusi yang mirip. Sehingga secara general kedua dataset tersebut memiliki parameter data instrinsik yang mirip.

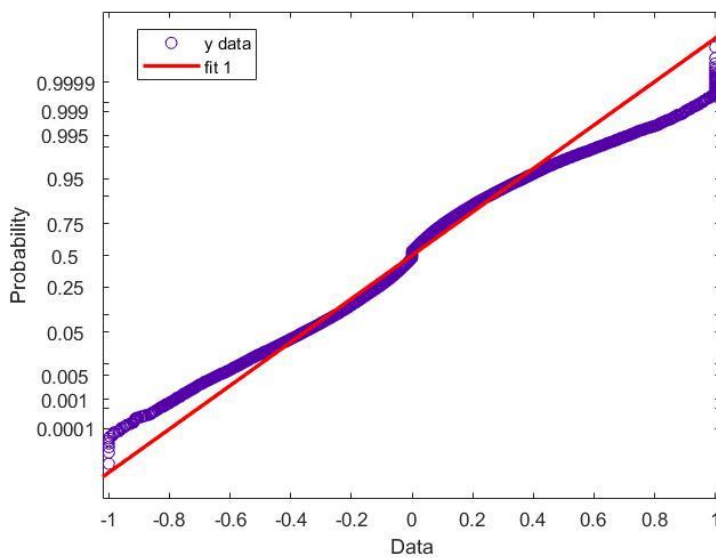
## 4.2 Evaluasi Babble Noise

Suara yang digunakan sebagai *masker* sebagai *babble noise* pada percobaan ini berupa kalimat Bahasa Indonesia sesuai dengan kaidah *International Phonetic Alphabet* (IPA) yang diucapkan oleh narasua laki-laki dan narasua perempuan. Pembuatan babble noise dilakukan dengan menggabungkan 3 file suara dari 3 penutur yang berbeda. Analisis distribusi statistik bising latar belakang berupa *babble noise* dilakukan untuk mengetahui karakteristik pemodelan untuk noise

yang dibuat sudah mendekati kondisi audial yang sebenarnya. Gambar 4.4 menunjukkan bentuk distribusi statistika dari babble noise. Sinyal tersebut dapat dikategorikan sebagai *non-Gaussian* atau tidak dapat diketahui dari uji normalitas probabilitas (Nadiroh, 2019) yang ditunjukkan melalui gambar 4.5. Proses *fitting* dilakukan dengan menggunakan *dfittool* perangkat lunak MATLAB dan uji normalitas dilakukan dengan menggunakan menu uji normalitas (*probability plot*) pada perangkat lunak Minitab.



**Gambar 4. 4** Distribusi Statistik Babble Noise



**Gambar 4. 5** Plot Uji Normalitas Babble Noise

Berdasarkan hasil *fitting* dan uji normalitas yang ditunjukkan pada Gambar 4.4 hingga 4.5 diperoleh informasi bahwa bising latar belakang sebagai *babble noise* merupakan sinyal non - *Gaussian*. Hal tersebut tampak pada gambar 4.4 yang menunjukkan adanya distribusi diluar garis normal. Oleh karena itu *babble noise* yang dibuat dapat dikatakan memiliki *Speech Intelligibility Index* yang tinggi berupa tingkat kejelasan suara yang didengar pendengar baik dengan kata lain semua informasi percakapan dapat didengar baik (*audible*) (Hornsby, 2004).

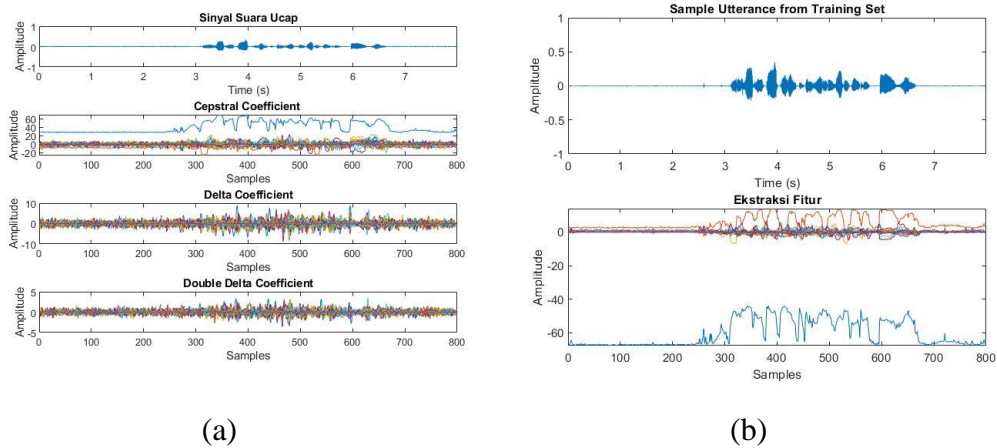
### 4.3 Hasil Uji Coba

Berikut adalah hasil uji coba yang dilakukan pada simulasi uji forensik. Pada masing masing jenis dataset dilakukan pengenalan identitas suara melalui hasil estimasi EER (error equal rate) setiap pengujian.

#### 4.3.1 Dataset Graz (Clean Data)

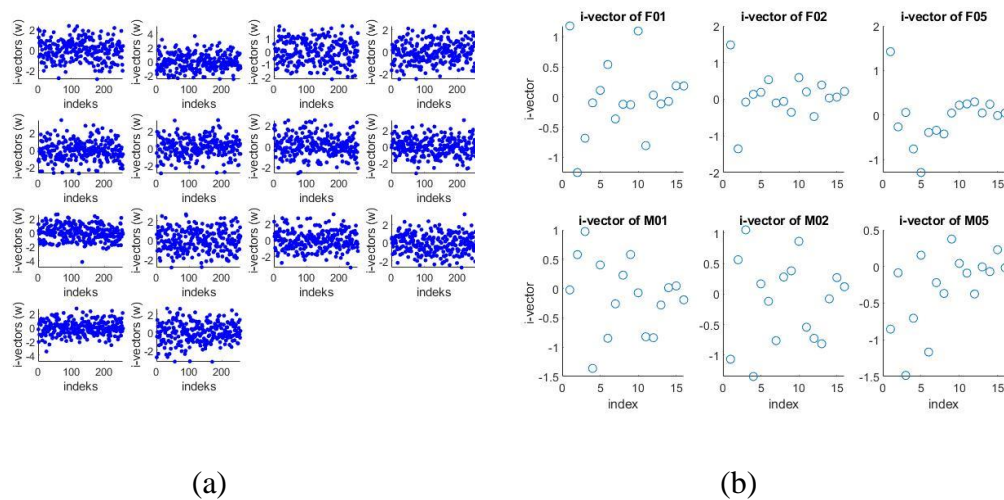
##### a. Pemrosesan Data Dataset Graz (Clean Data)

Sesuai dengan simulasi pada sistem forensik, dilakukan pembagian dataset menjadi bagian yaitu pelatihan (*training*) dan tes. 20 data penutur dibagi menjadi 2 yaitu 14 data penutur sebagai data *training* atau sebagai suara latar belakang (*background set*) dan 6 data sebagai data tes. Data tes yang terdiri atas 6 penutur (F01, F02, F05, M01, M02, M05) akan diregistrasikan sebagai data *enrollment* dan sebagai data verifikasi. Dari dataset asli setiap penutur terdiri dari 236 file, suara akan menjadi identitas penanda berdasarkan fitur penciri suara yang dimiliki berbeda disetiap penutur. File suara yang akan dianalisa dalam sistem identifikasi harus diekstraksi. Gambar 4.6 (a) menunjukkan hasil ekstraksi fitur berupa fitur mfcc, delta mfcc dan delta-delta mfcc dari suara pada *clean data*. Sedangkan untuk gambar 4.6 (b) menunjukkan hasil ekstraksi salah satu file suara pada *clean data* yang telah dinormalisasi.



**Gambar 4. 6** (a) Ekstraksi FileSuara Menjadi Fitur Mfcc, Delta Mfcc, Dan Delta Delta Mfcc. (b) Ekstraksi File Suara Menjadi Fitur Mfcc Yang Telah Dinormalisasi Dari Dataset Graz (Clean Data)

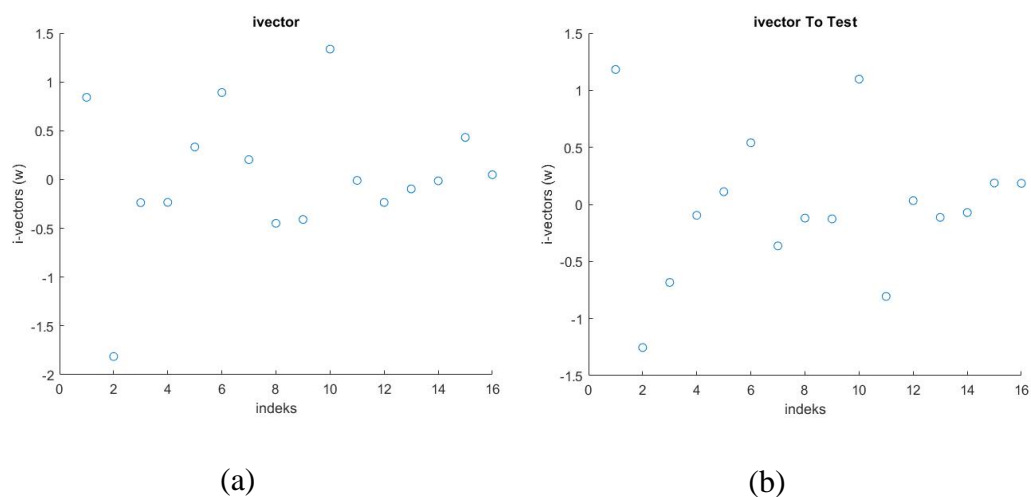
Data suara yang telah diekstraksi sebagai data training maupun data tes kemudian akan diolah untuk diperoleh model ivector masing-masing. Model i-vector yang didapatkan merupakan hasil model ivector dengan LDA dan WCNN *projection*.



**Gambar 4. 7** (a) Model I-Vector Dataset Background, (b) Model I-Vector Dataset Enrollment Dari Dataset Graz (Clean Data)

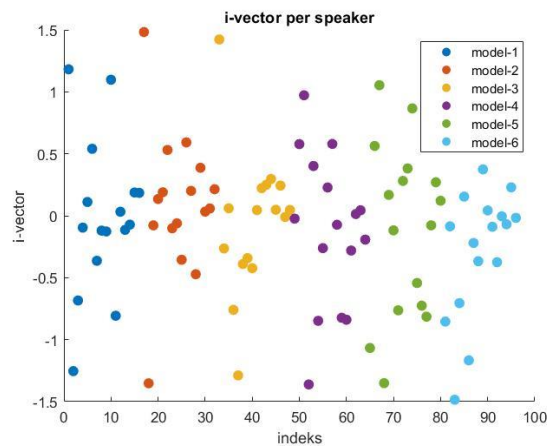
Dataset background adalah bagian dari analisa dalam identifikasi suara pada sistem forensik sehingga i-vector model dataset tersebut juga diperlukan. Namun dataset background tidak diberikan label atau id dalam proses uji melainkan id/label dari pembicara yang bersangkutan telah terekam dalam database, i-vector model dari *background* model hanya dibutuhkan sebagai bahan perbandingan yang termasuk

kedalam data *training* jika suara uji tidak memiliki kecocokan dengan dataset *enrollment*. Maka dataset uji akan dibandingkan dengan model i-vector dari dataset background. Hasil ivector model untuk setiap penutur pada dataset background berupa matriks dengan panjang 256x40. Gambar 4.7 (a) menunjukkan model ivector untuk setiap penutur di dalam dataset background. Sedangkan gambar 4.7 (b) menunjukkan model i-vector untuk masing masing speaker dalam dataset enrollment sesuai dengan id/label masing masing penutur. Masing masing penutur dalam dataset enrollment memiliki model i-vector dengan panjang 16x1. Perbedaan panjang data disebabkan oleh banyaknya data file suara yang diambil untuk proses pengolahan model i-vector pada masing-masing penutur yang disesuaikan dengan kebutuhan pada proses pengidentifikasian.



**Gambar 4.8** (a) Salah Satu Model I-Vector Tes, (b) Salah Satu Model I-Vector Target Yang Akan Saling Dibandingkan Dari Dataset Graz (Clean Data)

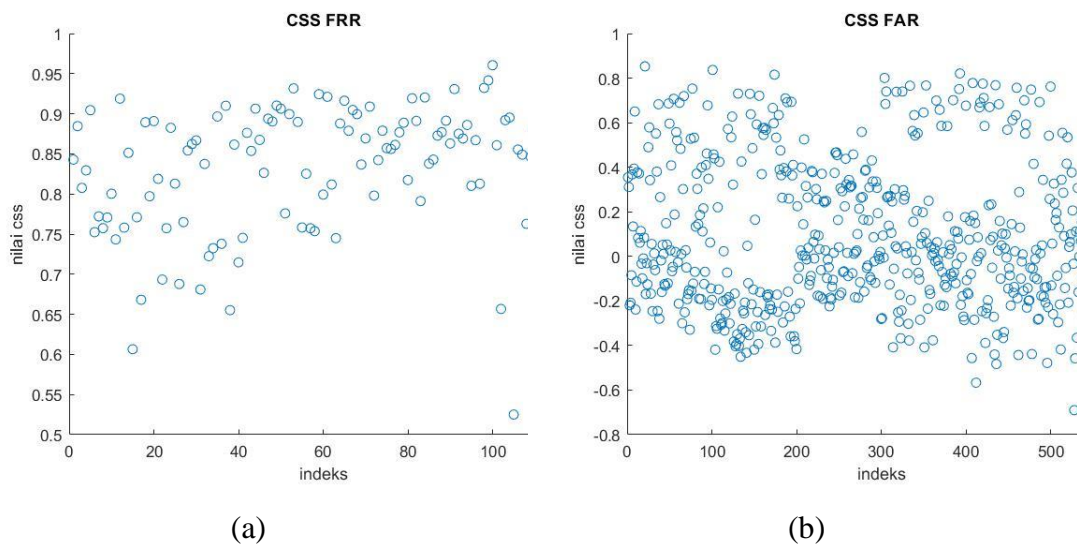
Gambar 4.8 (a) menunjukkan hasil dari model i-vector pada salah satu penutur pada dataset uji. Sedangkan gambar 4.8 (b) menampilkan salah satu model i-vector dari salah satu penutur di dalam dataset *enrollment*. Sedangkan gambar 4.9 menunjukkan hasil ivector dalam data enrollment yang telah terkelompokkan berdasarkan model i-vector setiap penutur.



**Gambar 4. 9** i-vector Setiap Penutur Pada Data Enrollment Dataset Graz (Clean Data)

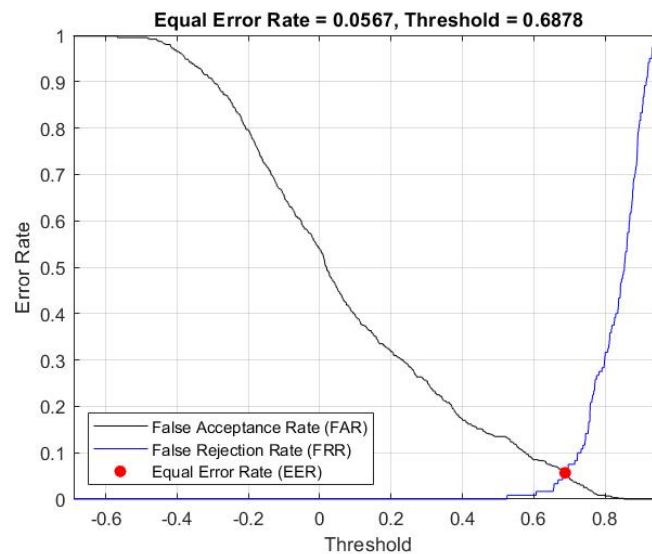
b. Verifikasi Dataset Graz (Clean Data)

Tahap verifikasi dilakukan dengan 2 jenis uji yakni menguji *speaker* yang sama antara data tes terhadap target dan menguji *speaker* campuran data tes terhadap target. Untuk mengukur kecocokan antar penutur digunakan css (*cosine similarity score*). Masing masing model i-vector pada dataset enrollment dan tes memiliki panjang yang sama yaitu matrik berukuran 16x1 untuk setiap penuturnya.



**Gambar 4. 10** (a) Hasil scattering css FRR, (b) Hasil scattering css FAR dari dataset Graz (clean data)

Sebaran data yang ditunjukkan dari data hasil verifikasi yang terdapat pada gambar 4.10 (a) dan (b). Gambar 4.10 menunjukkan bahwa verifikasi atau pengujian dengan speaker yang sama menghasilkan nilai css diatas 0.5 sampai 1. Sedangkan pengujian speaker acak pada gambar 4.10 (b) memiliki sebaran data acak yang sepadan dengan tujuan verifikasi model ke-2.



**Gambar 4. 11** Hasil ERR dari dataset Graz (clean data) dari dataset Graz (clean data)

Gambar 4.11 menunjukkan hasil grafik plot dari grafik FAR, grafik FRR dan persinggungan kedua grafik tersebut sebagai nilai EER. Nilai EER yang didapatkan dari hasil pengidentifikasian untuk dataset graz pada clean data menghasilkan nilai eer sebesar 5.6 %. Sehingga akurasi pada identifikasi untuk dataset ini sebesar 94.4 %, dengan nilai keakuratan tersebut dapat diambil kesimpulan bahwa data tes yang dijalankan memiliki kecocokan dengan data enrollment yang telah ditentukan id/label penutur yang dimaksud.

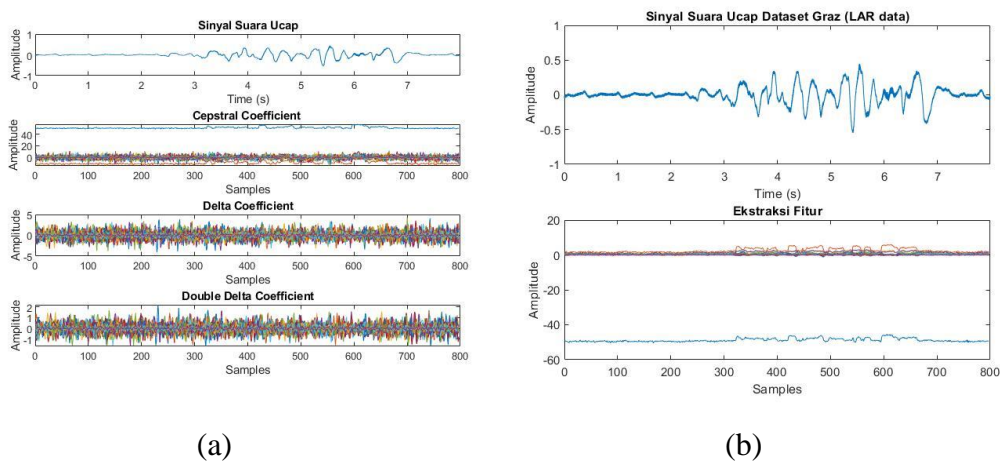
#### 4.3.2 Dataset Graz (Lar Data)

Data Lar merupakan hasil penggabungan sinyal suara pada clean data dengan noise *reverberant* sehingga didapatkan hasil sinyal suara seperti seolah olah terdapat gangguan telepon. Ketidakjelasan suara akibat tertutup (*mask*) oleh noise yang

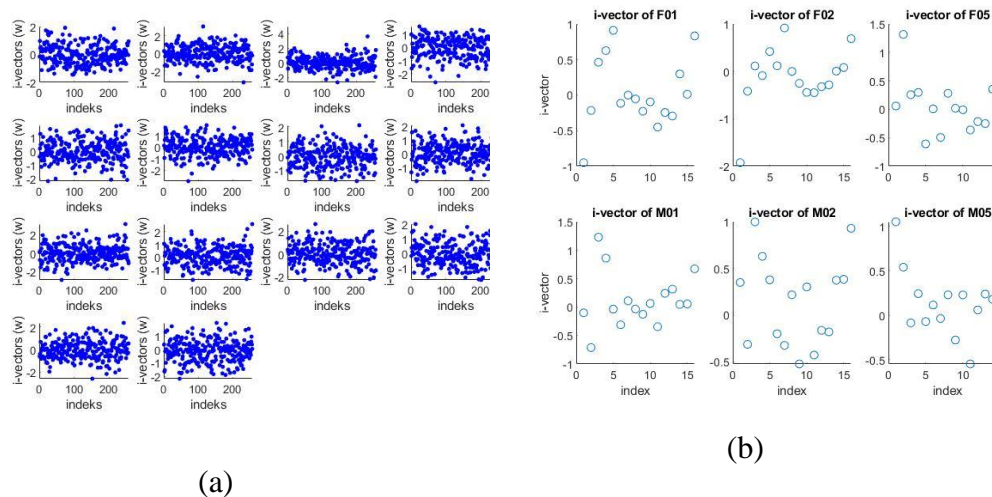
diberikan akan digunakan untuk menguji akurasi sistem identifikasi yang telah dibuat.

a. Pemrosesan Data Dataset Graz (Lar Data)

Gambar 4.12 menunjukkan hasil ekstraksi fitur pada salah satu file suara ucap pada dataset LAR.



**Gambar 4. 12** (a) ekstraksi file suara menjadi fitur mfcc, delta mfcc, dan delta delta mfcc. (b) ekstraksi file suara menjadi fitur mfcc yang telah dinormalisasi dari dataset Graz (lar data)

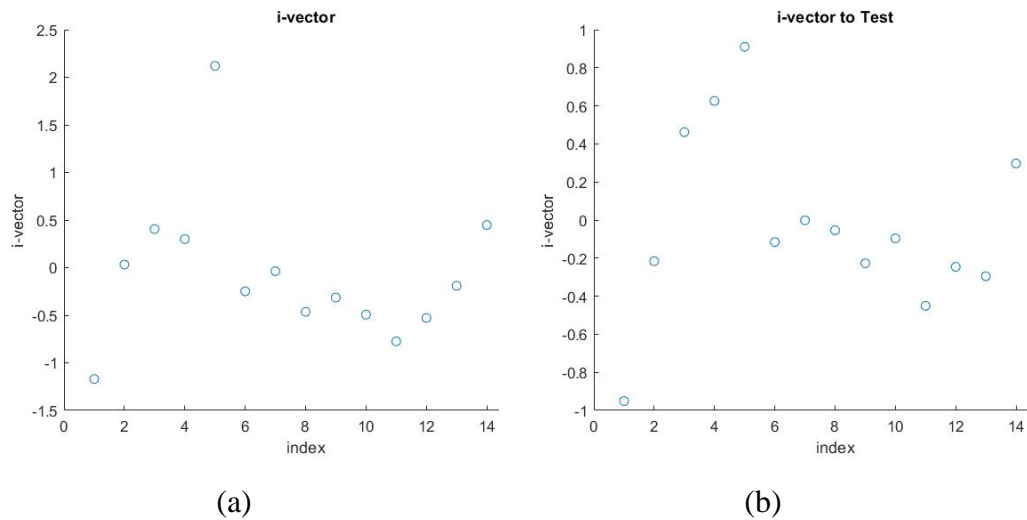


**Gambar 4. 13** (a) Scattering Model I-Vector Dataset Background, (b) Scattering Model I-Vector Dataset Enrollment Dari Dataset Graz (Lar Data)

Tidak ada perbedaan antara pengambilan model i-vector pada dataset Lar maupun clean data. Model i-vector setiap data yang diambil menggunakan projection

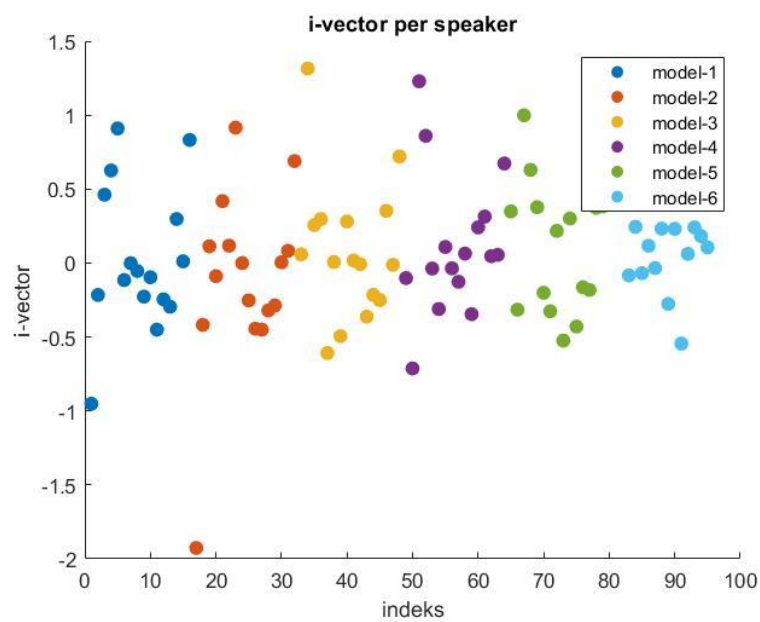


matriks menggunakan LDA dan WCNN. Gambar 4.13 (a) merupakan model i-vector untuk masing masing data pada dataset training. Sedangkan gambar 4.13 (b) merupakan hasil model i-vector dari data yang diregistrasikan atau *enrollment data*.



**Gambar 4. 14** (a) Salah Satu Model I-Vector Tes, (b) Salah Satu Model I-Vector Target Dari Dataset Graz (Lar Data)

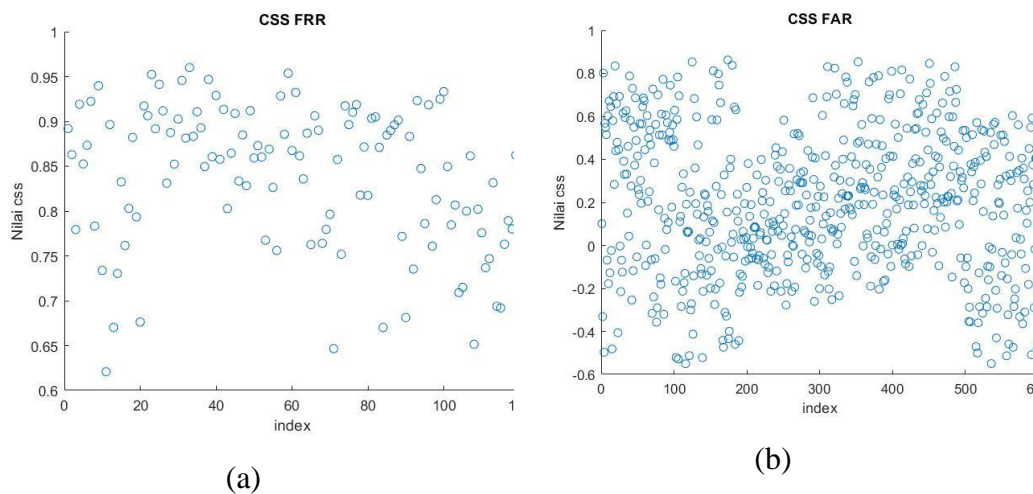
Gambar 4.14 menunjukkan salah satu model i-vector untuk data tes dan target. Model tersebut yang digunakan pada tahap verifikasi. Gambar 4.15 menunjukkan model i-vector setiap penutur pada data *enrollment*.



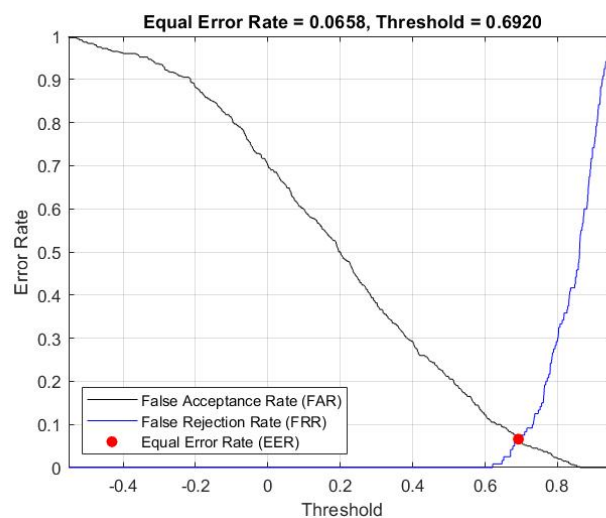
**Gambar 4. 15** i-vector Setiap Speaker Data Enrollment Dataset Graz (Lar Data)

### b. Verifikasi Dataset Graz (Lar Data)

Tahap verifikasi dengan 2 jenis uji yakni menguji *speaker* yang sama antara data tes terhadap target dan menguji *speaker* campuran data tes terhadap target pada dataset Graz (Lar Data) menghasilkan data css. Dalam pengklasifikasian, model ivector tes dan model ivector target akan dibandingkan menggunakan css (cosine similarity score). Sebagian hasil dari tahap verifikasi ditunjukkan pada tabel 4.4 dan 4.5. Gambar 4.16 (a) dan (b) menunjukkan seluruh data hasil perhitungan verifikasi baik pengujian dengan *speaker* sama (css FRR) atau berbeda (css FAR).



**Gambar 4. 16** (a) Hasil scattering css FRR, (b) Hasil scattering css FAR dari dataset Graz (lar data)



**Gambar 4. 17** Hasil ERR dari dataset Graz (lar data)

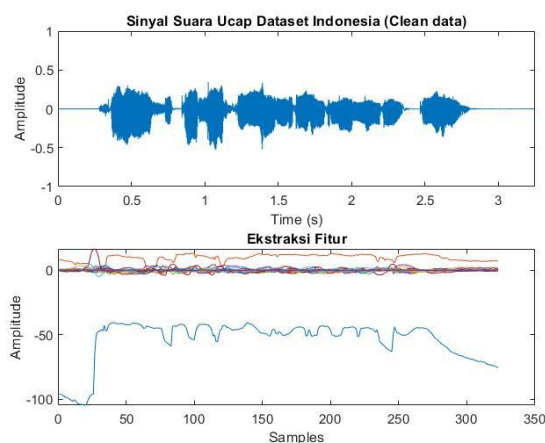
Hasil  $c_{ss}$  FAR dan FRR akan digunakan untuk mendefinisikan nilai FAR dan FRR. Sehingga EER untuk percobaan ini dapat ditentukan sebagai penentuan keakuratan *suspect* pada *target speaker*. Hasil EER pada percobaan dengan menggunakan dataset Lar Graz menunjukkan nilai sebesar 6.5%. Oleh karena itu dapat dikatakan keakuratan *suspect* dari *target speaker* pada dataset tersebut sebesar 93,5%.

### 4.3.3 Dataset Indonesia (Clean Data)

Dataset Indonesia terdiri atas 8 penutur dengan 4 penutur perempuan dan 4 penutur laki-laki. Dalam hal ini 4 penutur (F02, F03, M01, dan M04) dijadikan sebagai data tes dan target. Sedangkan sisa dataset tersebut akan dijadikan sebagai data training. Untuk menunjang simulasi forensik, analisa dilakukan tidak hanya pada dataset Indonesia (*clean data*). Terdapat simulasi tambahan berupa dataset rain, dan dataset babble. Dataset baru ini dibuat dengan menambahkan sinyal noise berupa suara hujan dan babble noise kedalam dataset Indonesia (*clean data*). Pengolahan sinyal suara ucap untuk dataset Indonesia melalui Langkah yang sama seperti dataset graz.

#### a. Pemrosesan Data Dataset Indonesia (Clean Data)

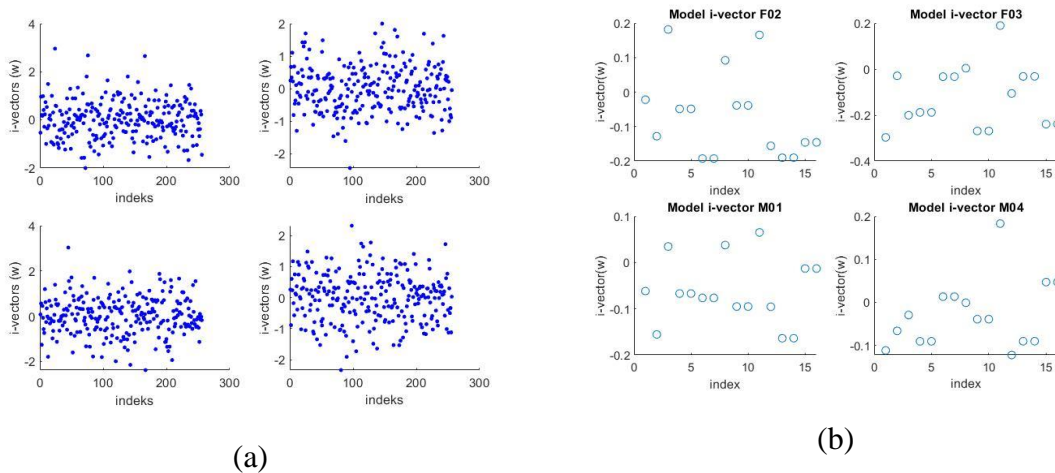
File suara ucap akan di ekstraksi untuk pengambilan fitur suara dari setiap file yang akan diproses. Pada gambar 4.16 ditunjukkan hasil ekstraksi fitur suara dari salah satu file suara dalam dataset Indonesia (Clean data) yang telah dinormalisasi.



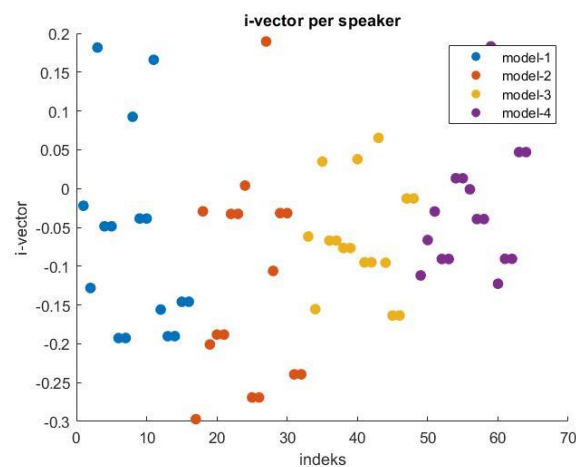
**Gambar 4. 18** Hasil Ekstraksi Fitur dan Normalisasi File Suara Clean Data

Dengan Langkah yang sama dataset akan terbagi menjadi data training, data enrollment, dan data tes. Setiap data tersebut akan diproses untuk diambil model i-vector masing masing data. Gambar 4.19 (a) menunjukkan model i-vector untuk

masing-masing penutur pada data training sedangkan gambar 4.19 (b) menunjukkan model i-vector untuk masing-masing penutur dalam data *enrollment*. Sedangkan gambar 4.20 menunjukkan hasil ivector untuk setiap *speaker* pada data enrollment dataset Indonesia (clean data)



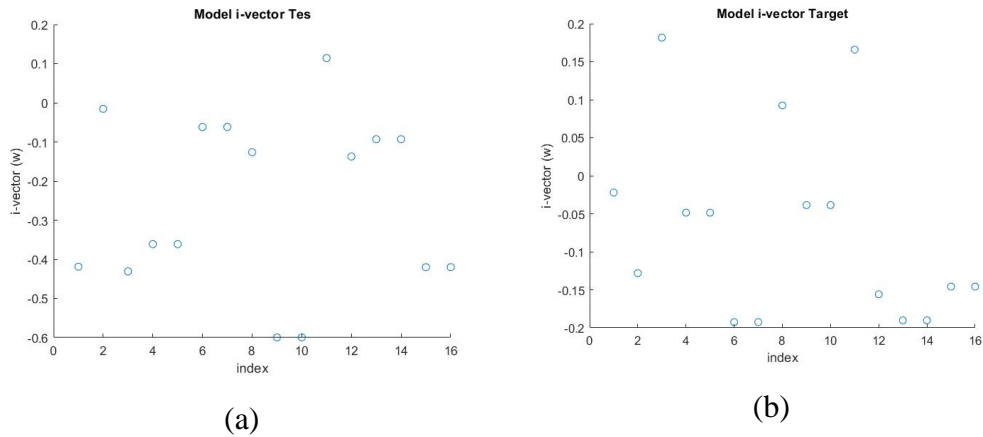
**Gambar 4. 19** (a) Model i-vector data training, (b) Model i-vector data enrollment Dataset Indonesia (Clean Data)



**Gambar 4. 20** i-vector Setiap Speaker Pada Data Enrollment

Model i-vector tes diambil dari data tes dan model i-vector target diambil dari data enrollment. Kemudian dalam tahap klasifikasi kedua model i-vector tersebut akan dibandingkan. Setiap model i-vector untuk masing-masing penutur memiliki pola model masing masing. Untuk memperoleh kecocokan antar kedua model tersebut, maka pola dalam kedua model yang akan dibandingkan akan dihitung menggunakan *css* (*cosine similarity score*). Gambar 3.21 (a) dan (b) merupakan

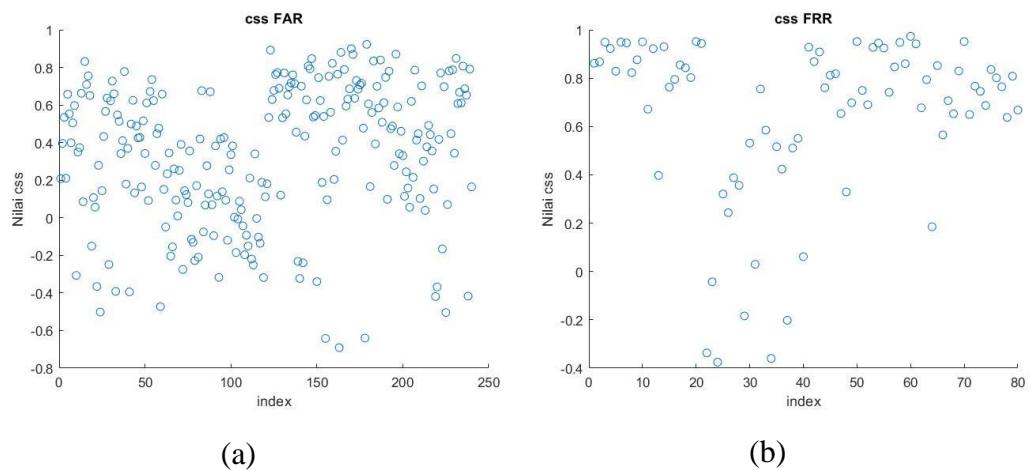
salah satu model i-vector tes dan target yang ditampilkan. Kedua model ivector ini berupa matriks berukuran  $16 \times 1$ .



**Gambar 4. 21** (a) Model i-vector Tes, (b) Model i-vector Target Dataset Indonesia (Clean Data)

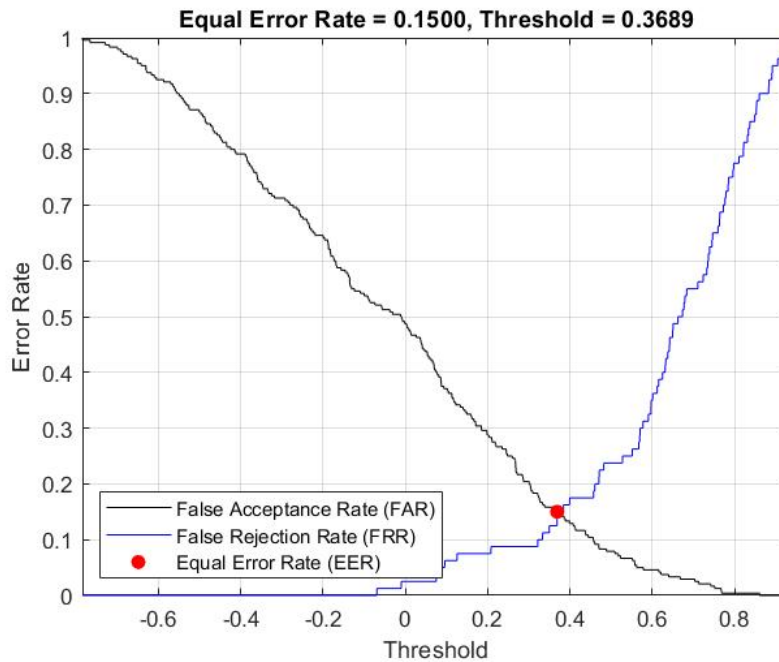
#### b. Verifikasi Dataset Indonesia (Clean Data)

Tahap verifikasi dengan 2 jenis uji yakni menguji *speaker* yang sama antara data tes terhadap target dan menguji *speaker* campuran data tes terhadap target pada dataset Indonesia (Clean Data) menghasilkan data css.



**Gambar 4. 22** (a) Hasil css FAR, (b) Hasil css FRR Dataset Indonesia (Clean Data)

Dalam hal ini css digunakan untuk mengukur seberapa dekat kemiripan pola model ivector tes dan target yang akan dibandingkan. Rentang nilai yang dimiliki css berada pada -1 sampai 1 seperti yang ditunjukkan pada gambar 4.22.



**Gambar 4. 23** Plot Grafik FAR, FRR dan EER Dataset Indonesia (Clean Data)

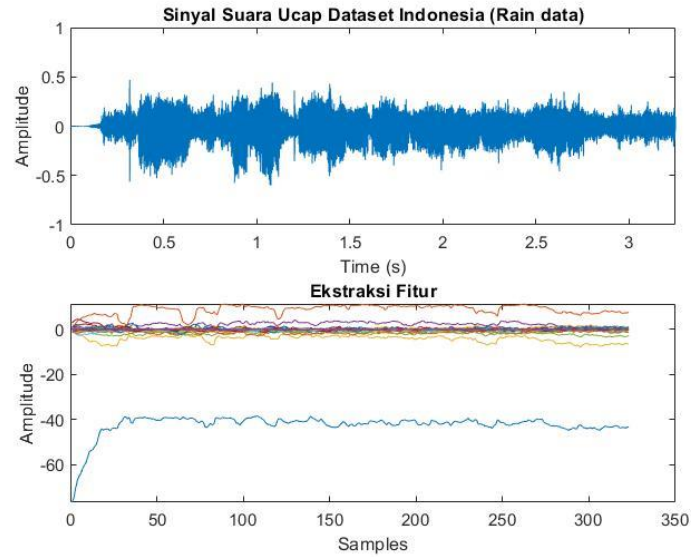
Dari *css* FAR dan *css* FRR akan diolah untuk mendapatkan grafik FAR dan FRR seperti pada gambar 4.23. ERR yang merupakan perpotongan dari kedua grafik tersebut merupakan nilai yang berisi informasi tentang keakuratan sistem terhadap identifikasi data tes dan target. Untuk dataset Indonesia (Clean data) mendapatkan nilai EER sebesar 15%, sehingga dapat diartikan bahwa keakuratan pada proses data ini memiliki nilai sebesar 85%. Nilai ini jauh lebih rendah dibawah angka keakuratan yang dihasilkan dengan proses identifikasi pada dataset Graz baik untuk *clean* data maupun *lar* data.

#### 4.3.4 Dataset Indonesia (Rain Data)

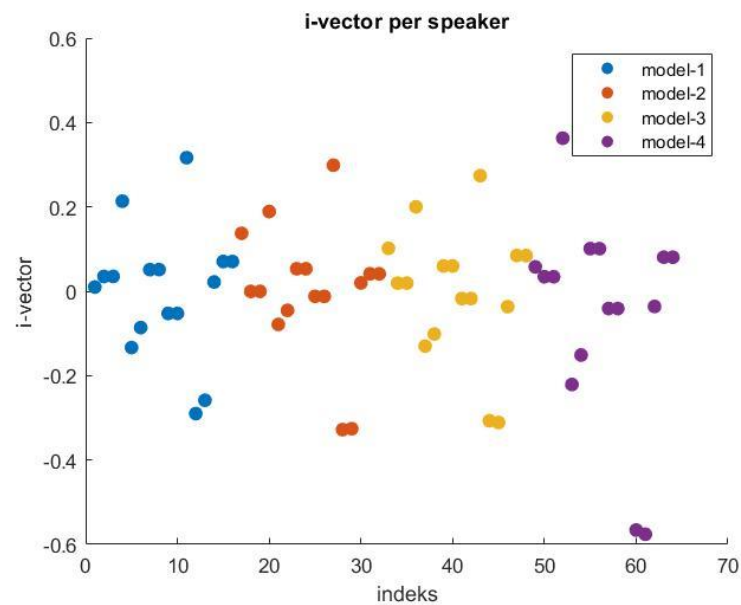
##### a. Pemrosesan Data Dataset Indonesia (Rain Data)

Tampak pada gambar 4.24 file suara ucap yang digunakan untuk simulasi berupa data dengan noise babble (data rain) dengan variasi SNR sebesar 0, 10 dan 20. Pengolahan untuk dataset tersebut diolah seperti dataset sebelumnya tanpa. Hal ini dikarenakan untuk menguji kemampuan sistem identifikasi terhadap kondisi sebenarnya dalam analisa forensik. Bable noise digunakan sebagai simulasi kondisi dimana terdapat pembicara lain yang lebih dari 1 orang dalam rekaman speaker

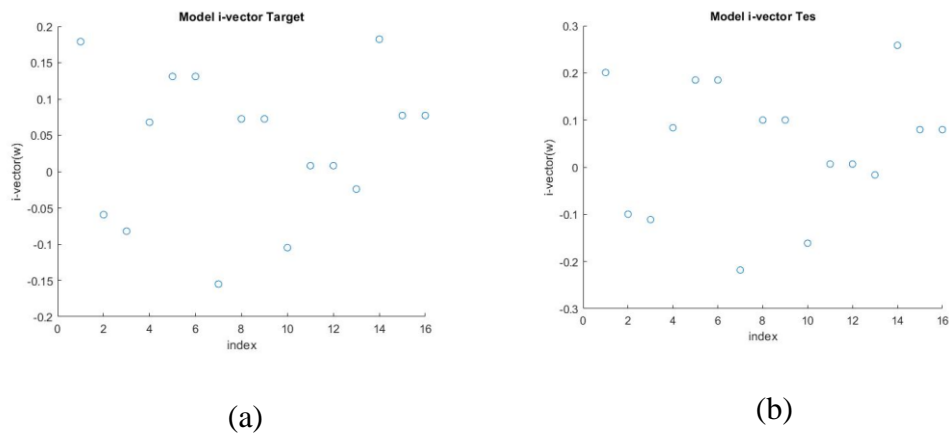
target. Gambar 4.25 hasil dari ekstraksi i-vector berupa model ivector untuk setiap penutur pada setiap penutur pada data *enrollment*.



**Gambar 4. 24** Hasil Ekstraksi Fitur File Suara Ucapan Data Rain



**Gambar 4. 25** Hasil i-vector Model Data Enrollment Pada Data Rain

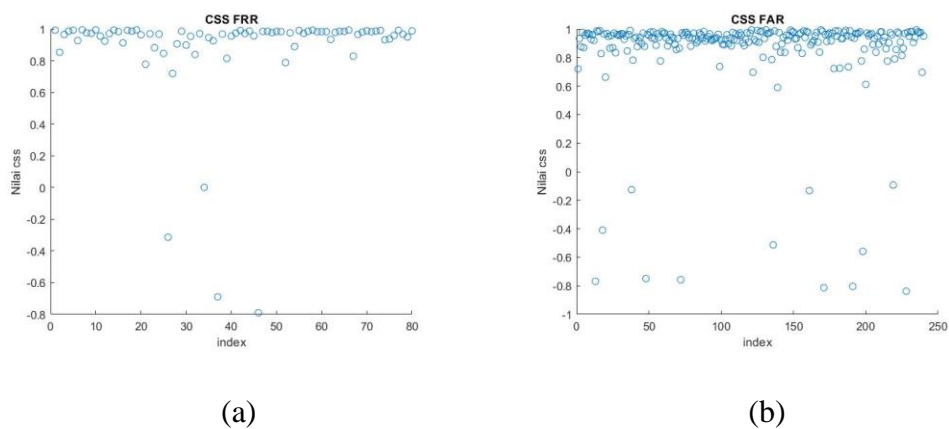


**Gambar 4. 26** (a) Model i-vector Tes Dan (b) Model i-vector pada Target

Gambar 4.26 menunjukkan salah satu pasang model i-vector tes dan target untuk masing masing jenis dataset. Dalam proses klasifikasi kedua model tersebut akan saling dibandingkan menurut pola model i-vector yang telah dimiliki masing masing penutur. Proses klasifikasi menghasilkan nilai perbandingan kedua model tersebut menggunakan *css (cosine similarity score)*.

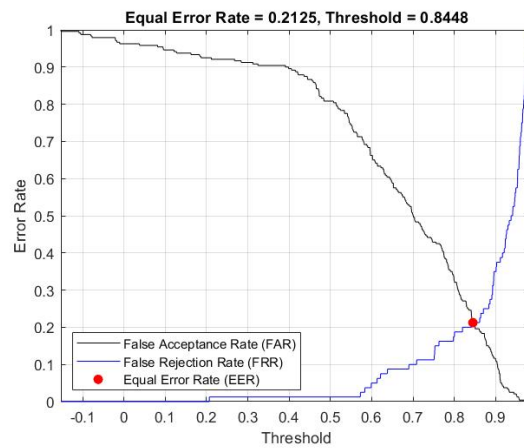
#### b. Verifikasi Dataset Indonesia (Data Rain)

Tahap verifikasi dengan 2 jenis uji yakni menguji *speaker* yang sama antara data tes terhadap target dan menguji *speaker* campuran data tes terhadap target pada dataset Indonesia (Clean Data) menghasilkan data *css* seperti pada gambar 4.27. Evaluasi dari sistem verifikasi pada dataset Indonesia (data rain) ditunjukkan pada gambar 4.28.



**Gambar 4. 27** (a) Hasil CSS FAR dan (b) Hasil CSS FRR Pada Data Rain

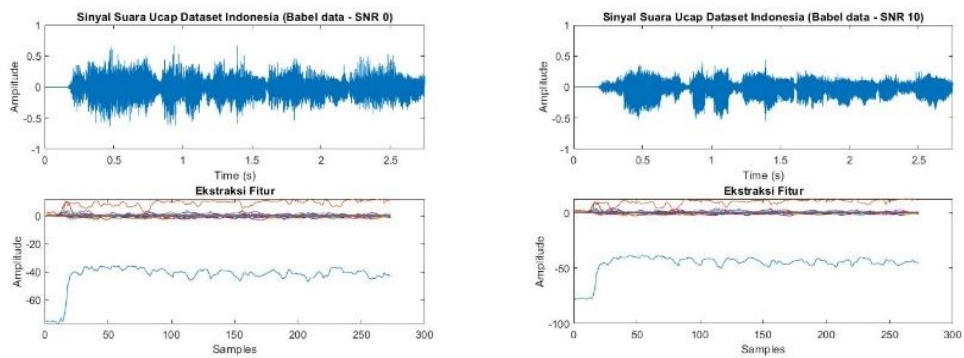




**Gambar 4. 28** Hasil Plot Grafik FAR, FRR, dan EER pada data rain

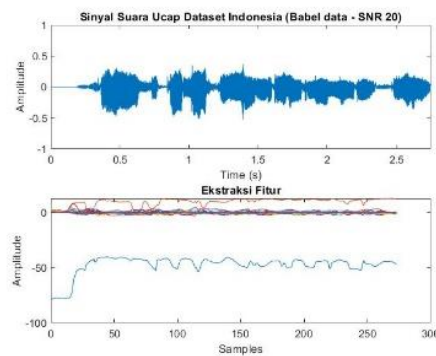
### 4.3.5 Dataset Indonesia (Babble Data)

#### a. Pemrosesan Data Dataset Indonesia (Babble Data)



(a)

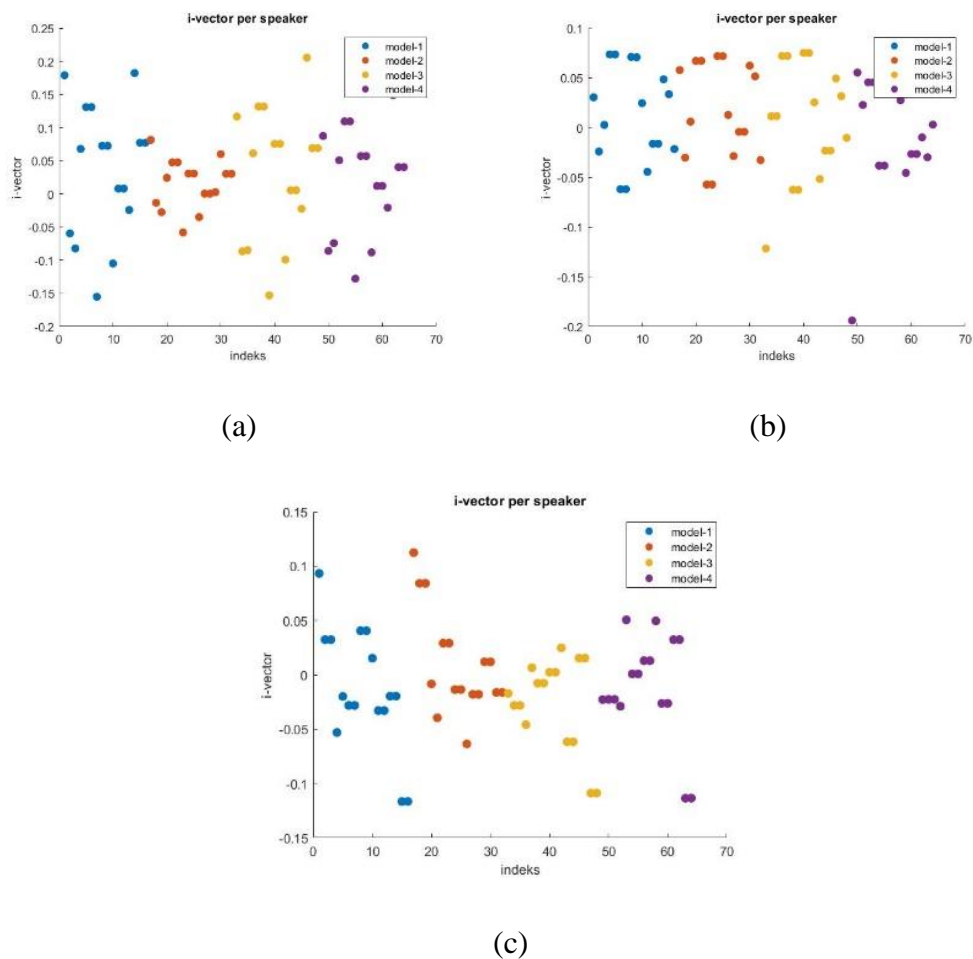
(b)



(c)

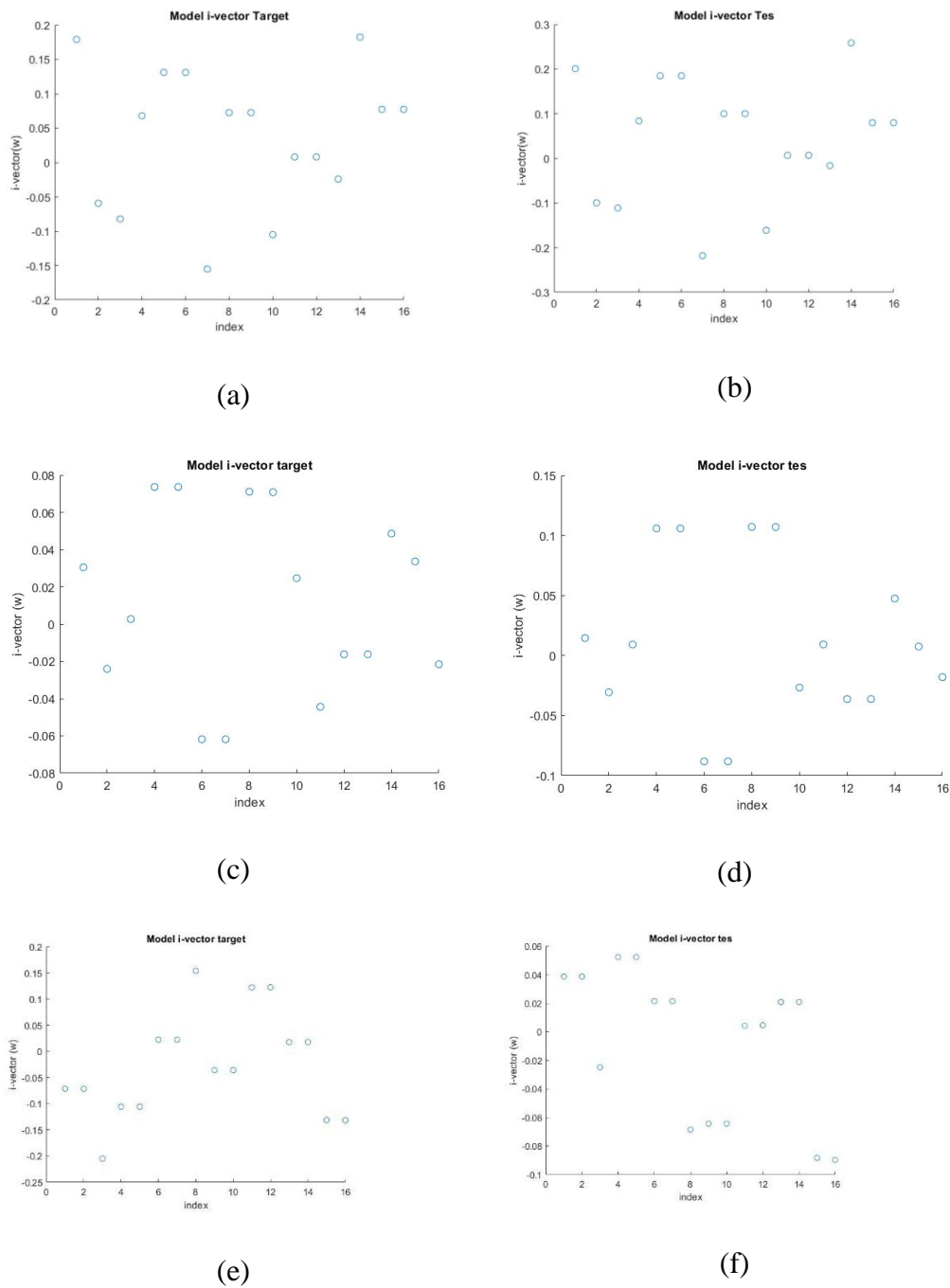
**Gambar 4. 29** Hasil ekstraksi fitur file suara ucap (a) data babble – SNR 0, (b) data babble- SNR 10, (c) data babble - SNR 20.

Tampak pada gambar 4.29, file suara ucap yang digunakan untuk simulasi berupa data dengan noise babble (data babble) dengan variasi SNR sebesar 0, 10 dan 20. Pengolahan untuk dataset tersebut diolah seperti dataset sebelumnya tanpa. Hal ini dikarenakan untuk menguji kemampuan sistem identifikasi terhadap kondisi sebenarnya dalam analisa forensik. Bable noise digunakan sebagai simulasi kondisi dimana terdapat pembicara lain yang lebih dari 1 orang dalam rekaman speaker target. Gambar 4.30 hasil dari ekstraksi i-vector berupa model ivector untuk setiap penutur pada setiap penutur pada data *enrollment*.



**Gambar 4. 30** Hasil ivector model data enrollment pada (a) data babble – SNR 0, (b) data babble - SNR 10, (c) data babble - SNR 20.

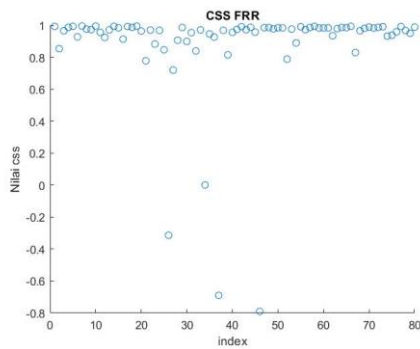
Setiap jenis dataset yang akan dianalisa akan berujung pada perolehan model i-vector baik untuk tes maupun target. Gambar 4.31 menunjukkan salah satu pasang model i-vector tes dan target untuk masing masing jenis dataset pada data babble.



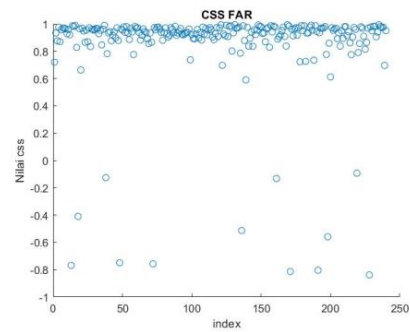
**Gambar 4. 31** Model ivector tes dan target pada (a) & (b) data babble - SNR 0, (c) & (d) data babble – SNR 10, (e) & (f) data babble – SNR 20

Dalam proses klasifikasi kedua model tersebut akan saling dibandingkan menurut pola model i-vector yang telah dimiliki masing-masing penutur. Proses klasifikasi menghasilkan nilai perbandingan kedua model tersebut menggunakan *css* (*cosine similarity score*).

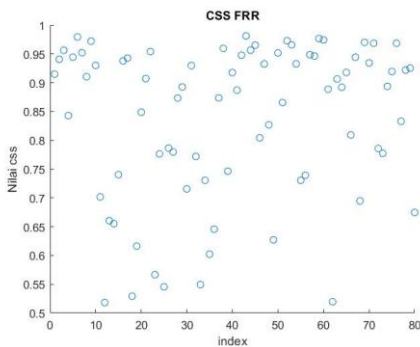
## b. Verifikasi Dataset Indonesia (Data Babble)



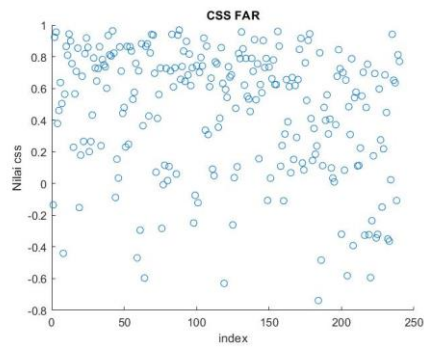
(a)



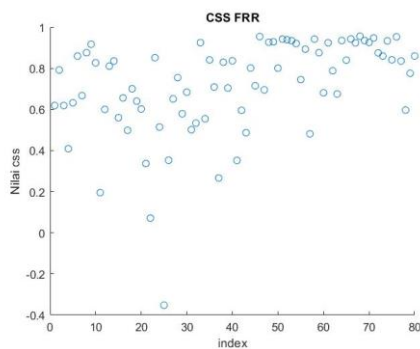
(b)



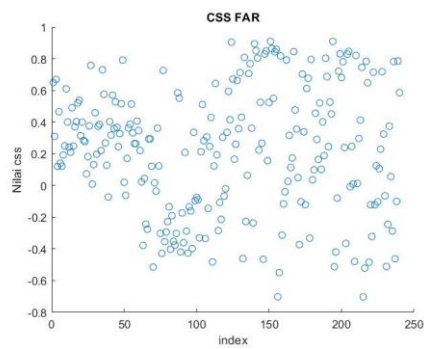
(c)



(d)



(e)

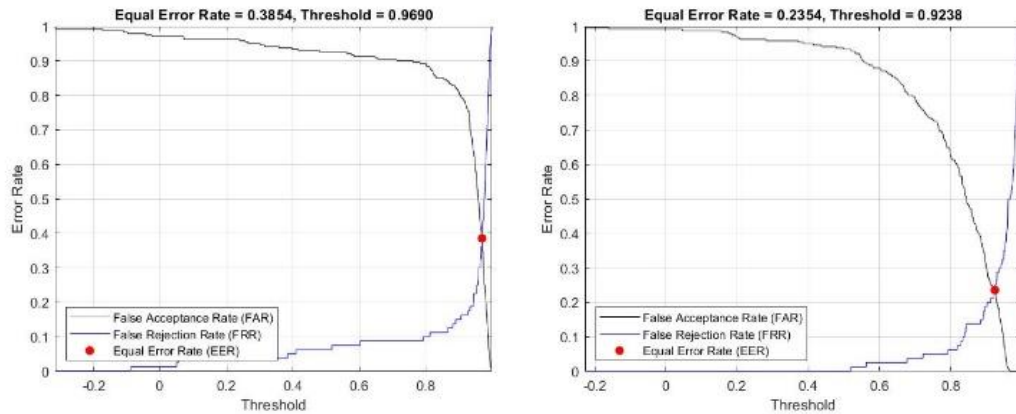


(f)

**Gambar 4. 32** Hasil CSS FAR dan CSS FRR pada (a) & (b) data babble - SNR 0, (c) & (d) data babble – SNR 10, (e) & (f) data babble – SNR 20.

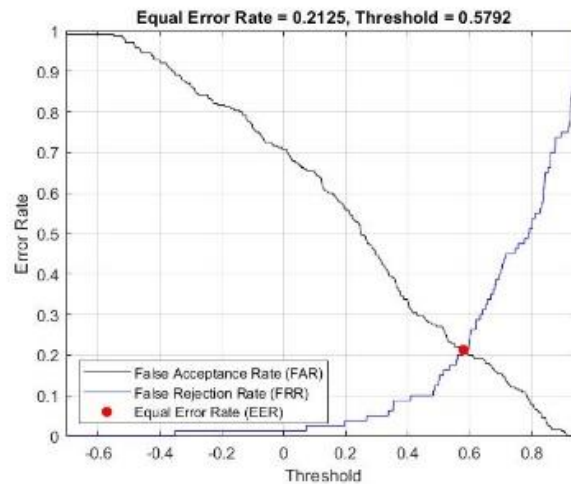
Tahap verifikasi dengan 2 jenis uji yakni menguji *speaker* yang sama antara data tes terhadap target dan menguji *speaker* campuran data tes terhadap target pada dataset Indonesia (Rain Data) menghasilkan data css seperti pada gambar 4.32.

Evaluasi dari sistem verifikasi pada dataset Indonesia (data babble) ditunjukkan pada gambar 4.33. Dataset yang memiliki rasio bising atau noise paling besar dengan ditunjukkan oleh nilai SNR paling kecil memiliki error performansi lebih paling besar, begitupun sebaliknya.



(a)

(b)



(c)

**Gambar 4. 33** Hasil Plot Grafik FAR, FRR, dan EER pada (a) data babble - SNR 0, (b) data babble – SNR 10, (c) data babble – SNR 20

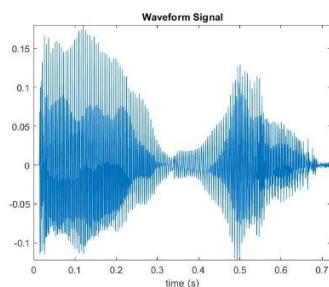
#### 4.4 Evaluasi Performansi

Evaluasi dari performansi pada masing masing percobaan untuk setiap dataset dapat ditunjukkan pada tabel 4.8. EER didapatkan dari perpotongan nilai FAR dan FRR. Sedangkan akurasi merupakan pengurangan 100 % dengan EER (%).

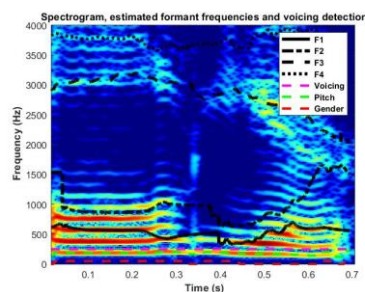
**Tabel 4. 1** Hasil EER dan Akurasi Pada Dataset Noise

Jenis Dataset	EER (%) / Performansi	Akurasi (%) / Confidence Level	Treshold (%) / sensitivitas
Data Graz (Data <i>Clean</i> )	5.67	94.33	68.78
Data Graz (Data <i>LAR</i> )	6.58	93.42	69.20
Data Indonesia (Data <i>Clean</i> )	15.00	85.00	36.89
Data Indonesia (Data <i>Rain</i> )	21.25	71.75	84.48
Data Indonesia (Data <i>Babble</i> - SNR 0)	38.54	61.46	96.90
Data Indonesia (Data <i>Babble</i> - SNR 10)	23.54	76.46	92.38
Data Indonesia (Data <i>Babble</i> - SNR 20)	17.50	82.46	57.92

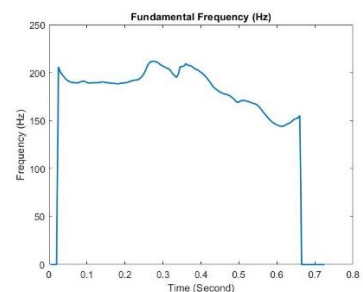
Tabel 4.1 menunjukkan nilai EER, akurasi dan sensitivitas untuk setiap jenis dataset yang telah diolah. Nilai EER menunjukkan performa diskriminasi antara skor target dan non-target. Nilai EER yang semakin kecil menunjukkan semakin baiknya performa diskriminasi suatu sistem. Sedangkan nilai threshold menunjukkan sensitivitas sistem sehingga semakin besar sensitivitas maka sistem semakin aman ditunjukkan dengan semakin bergesernya titik EER sebelah kanan, namun sistem akan lebih sulit diakses karena nilai FAR semakin tinggi atau dengan kata lain salah mengenali penutur.



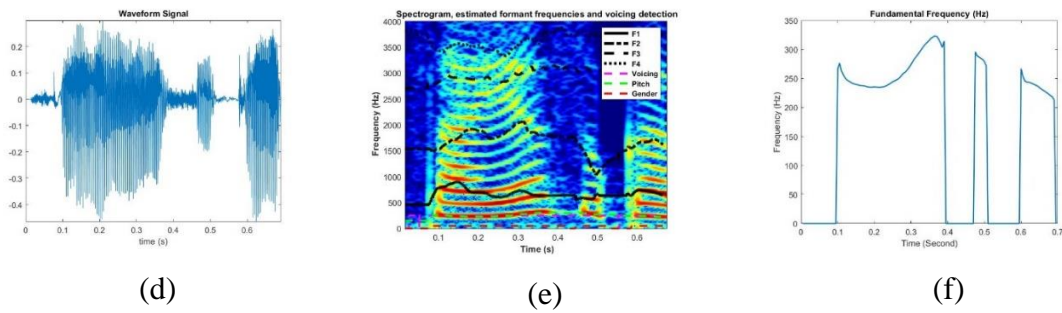
(a)



(b)

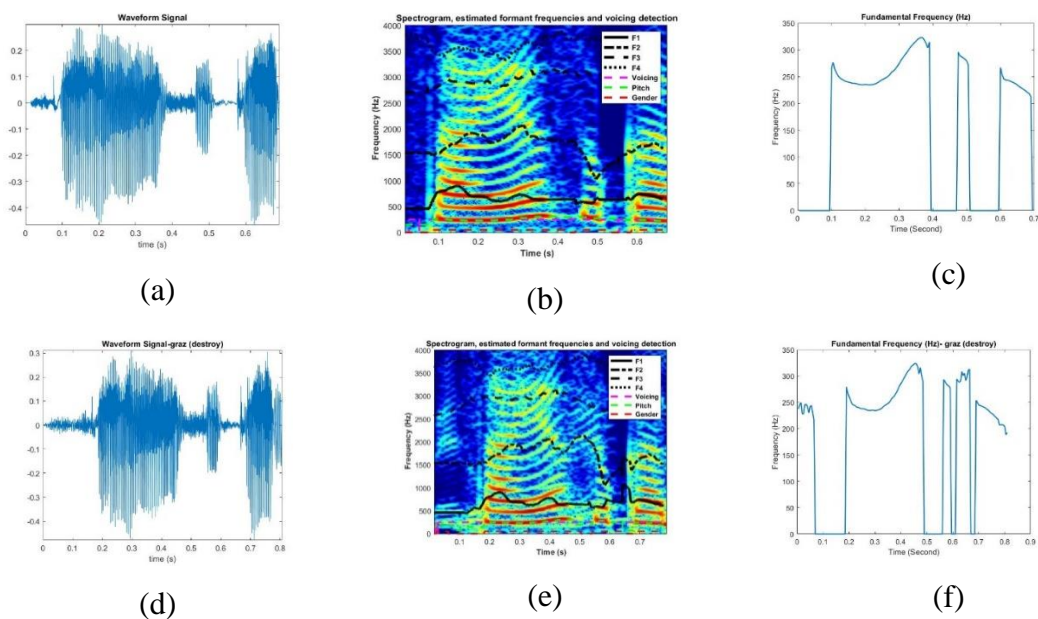


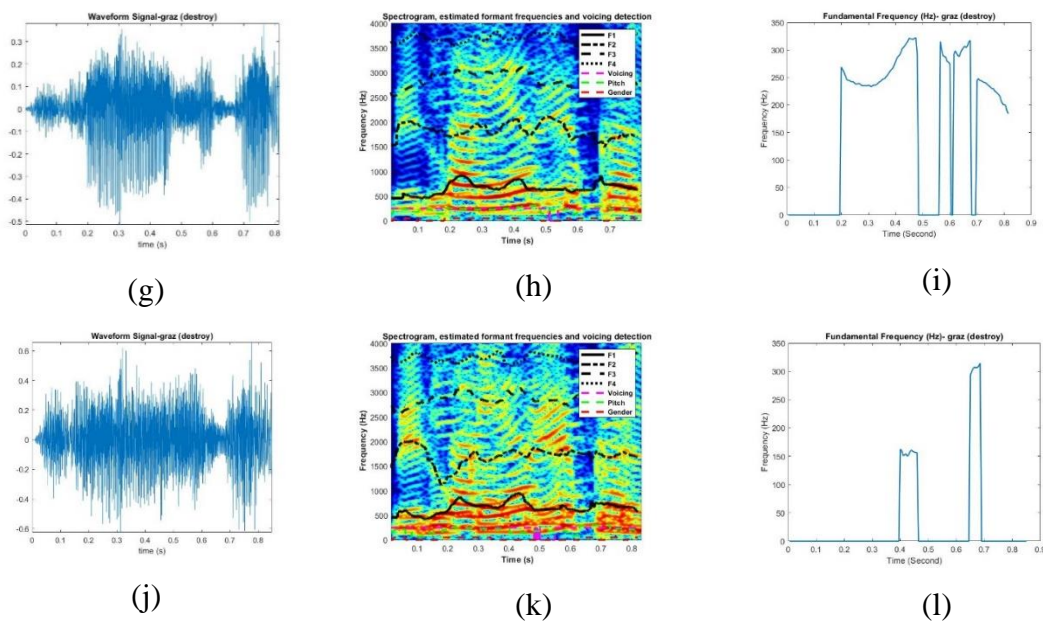
(c)



**Gambar 4. 34** (a) waveform, (b) spectrogram dan (c) F0 Graz “all year”, (d) waveform, (e) spectrogram dan (f) F0 indonesia “saya suka”

Jika dibandingkan antara data suara *clean*, nilai EER data graz sebesar 5.67% sedangkan nilai EER data Indonesia sebesar 15%. Perbedaan nilai eer yang cukup signifikan dari hasil perolehan uji verifikasi terhadap dataset Graz dan dataset Indonesia dapat dianalisa karena perbedaan pada struktur formant dan F0 (*Fundamental Frequency*). Seperti yang tampak pada gambar 4.34 menunjukkan salah satu hasil plot formant dan F0 dari kedua dataset *clean* yang dipotong daerah voice dengan mengandung vowels. Dari hasil yang didapatkan menunjukkan bahwa formant dari dataset Indonesia lebih jelas / *clear*, sedangkan F0 dataset Indonesia lebih *glide*. Perbedaan structural tersebut menjadi salah satu aspek internal terkait perbedaan nilai yang didapatkan pada tabel 4.1 Faktor lain yang berpeluang menjadikan kualitas kedua dataset berbeda yakni kualitas perekaman berupa kondisi proses perekaman serta alat yang digunakan.



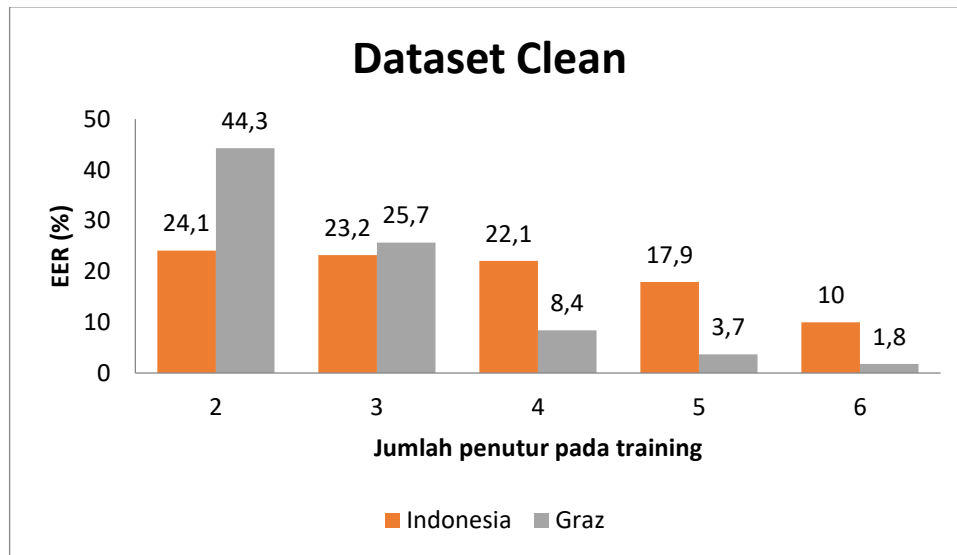


**Gambar 4.35** Struktur waveform, spectrogram dan F0 (a)-(c) file clean, (d)-(f) data dengan babble noise SNR 20, (g)-(i) data dengan babble noise SNR 10, dan (j)-(l) data dengan babble noise SNR 0

Hasil pada percobaan pada data suara dengan *babble noise* menunjukkan semakin tinggi nilai SNR maka semakin kecil error percobaan yang didapatkan. Hal tersebut terjadi karena saat nilai SNR semakin besar maka sinyal suara mendekati keadaan seperti sinyal suara *clean*. Oleh karena itu perolehan nilai EER untuk data suara dengan *babble noise* semakin kecil seiring bertambahnya nilai SNR yang digunakan. Perbedaan dari sampel dataset dengan penambahan *babble noise* dapat dianalisa dari perbedaan struktur Formant dan F0 (*Fundamental Frequency*). Signal to noise ratio (SNR) menunjukkan ratio perbandingan power antara sinyal dan noise yang diberikan. Semakin besar nilai SNR maka power noise semakin kecil dibandingkan dengan sinyalnya, begitupun sebaliknya. Variasi SNR yang diberikan memberikan data hasil yang tampak pada tabel 4.. Gambar 4.35 (a)-(c) untuk data suara file clean, (d)-(f) data suara dengan babble noise SNR 20, (g)-(i) data suara dengan babble noise SNR 10, dan (j)-(l) data suara dengan babble noise SNR 0. Jika dibandingkan dengan data suara bersih, struktur formant dan F0 dari data suara yang diberikan babble noise tampak berberda. Dari gambar 4.35 tersebut tampak semakin kecil SNR pada data suara maka struktur formantnya semakin bertumpuk. Sedangkan nilai F0 yang dihasilkan berbeda, mulai dari adanya F0 tambahan hingga menghasilkan F0 baru. Hal inilah yang menyebabkan pembacaan error yang



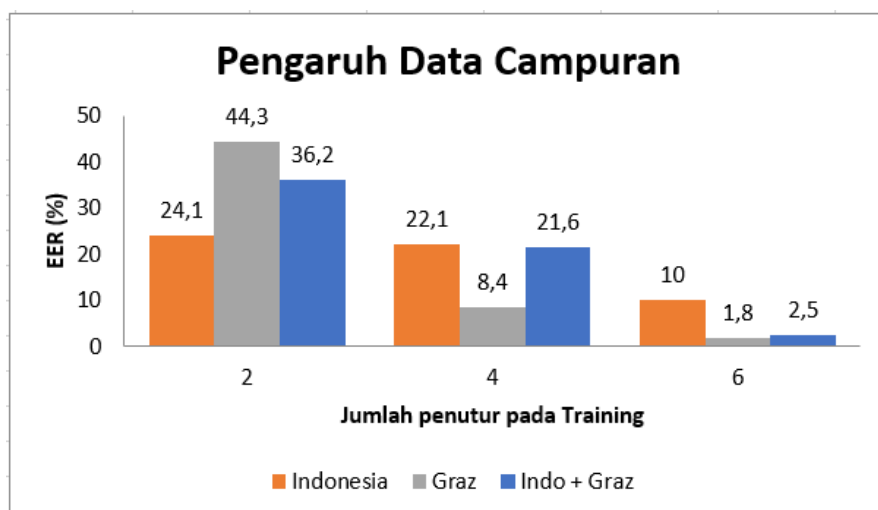
dihasilkan oleh data suara dengan babble noise SNR 0 lebih tinggi. Nilai EER dari dataset Indonesia yang dikenai derau misalnya pada noise babble yang meningkat 2x. Hal ini menunjukkan bahwa terdapat peluang pada noise babble dan noise hujan tersebut terhadap Teknik yang kuat untuk memberikan pengaruh (Garcia-Romero, et al., 2019).



**Gambar 4. 36** Pengaruh Banyak Speaker Pada Data Training Untuk Performansi Sistem

Pada gambar 4.36 merupakan hasil dari perbandingan antara dataset Indonesia dan dataset Graz. Pengujian dilakukan dengan menyamakan banyak dataset yang digunakan menjadi 8 penutur (jumlah maksimal penutur yang dimiliki dataset Indonesia) yang terdiri atas 4 penutur perempuan dan 4 penutur laki-laki. Pada pengujian tersebut dikenakan variasi jumlah penutur pada tahap training sebagai dataset background. Karena pada pengujian dataset dibagi menjadi 2 yakni training dari background set dan verifikasi sehingga jika terdapat 2 penutur sebagai pada training maka sisanya yaitu 6 penutur sebagai data untuk verifikasi. Gambar tersebut menunjukkan bahwa semakin banyak penutur pada training mempengaruhi performansi sistem. Hal tersebut dapat dilihat dari semakin menurunnya nilai EER yang didapatkan saat jumlah penutur pada training diperbanyak yang terjadi pada kedua dataset yakni dataset Indonesia dan dataset Graz. Dapat dilihat pada subbab 3.2 dari gambar 3.1 yang menunjukkan blok diagram sistem dimana tahap training menjadi bagian yang penting. Pada tahap dilakukan pelatihan gmm untuk

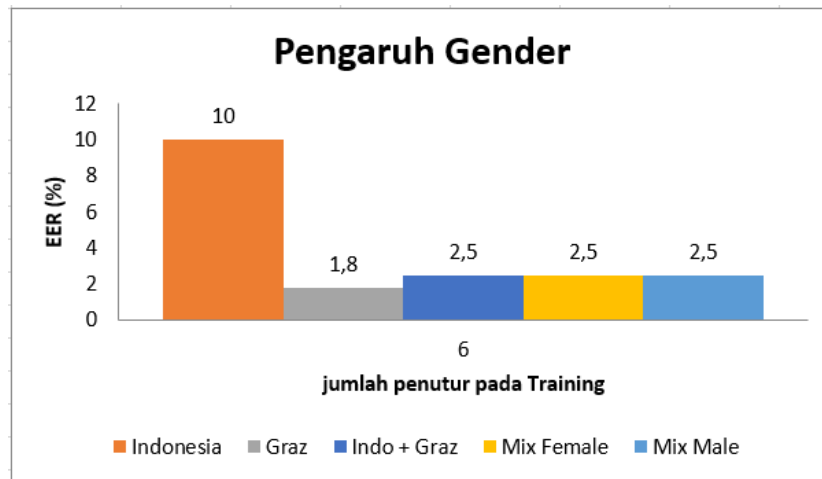
mendapatkan ubm (universal background model). Kemudian penginisiasian total variabilitas untuk mengkompensasi adanya informasi tentang pembicara di dalam faktor saluran dari JFA dengan mengetahui total variabilitas ruangnya yang kemudian dikenakan pada bagian verifikasi untuk dapat mengektaksi atau memodelkan i-vector setiap penuturnya. Projection matriks yang dihasilkan dari proses training digunakan untuk mengkompensasi intersesi pada proses verifikasi. Oleh karena itu banyaknya penutur pada background set untuk training dapat mempengaruhi performansi sistem. Gambar 4.36 juga menunjukkan bahwa kedua dataset memiliki sifat yang sama yakni performansinya semakin meningkat saat dikenakan penambahan penutur pada training. Performansi terbaik untuk dataset Indonesia didapatkan EER sebesar 10% sedangkan dataset Graz sebesar 1,8%.



**Gambar 4. 37** Hasil Pencampuran Database Indonesia dan Graz

Saat dilakukan pencampuran dataset Indonesia dan Graz menjadi satu dataset maka dihasilkan seperti pada gambar 4.37 yang ditunjukkan oleh diagram batang berwarna biru. Skema pengujiannya sama yakni menggunakan 8 penutur dimana setiap dataset masing masing diambil 4 penutur (2 perempuan dan 2 laki-laki). Dari gambar 4.37 dikenakan variasi banyak penutur untuk training minimum, medium dan maksimum. Hasil yang didapatkan saat database dicampurkan seperti yang ditunjukkan pada gambar tersebut menghasilkan EER diantara hasil EER dari pada pengujian Indonesia dan Graz. Pada gambar 4.37 menunjukkan salah satu hasil uji dataset campuran EER sebesar 2,5% pada jumlah penutur training maksimum. Nilai EER tersebut merepresentasikan performansi sistem gabungan dari dataset

Indonesia dan Graz. Nilai EER yang dihasilkan dari dataset campuran tersebut cenderung mendekati nilai EER dataset Indonesia yang memiliki error lebih tinggi dari pada EER dataset Graz. Oleh karena itu dapat dikatakan dataset Indonesia lebih dominan untuk mempengaruhi performansi dataset gabungan.



**Gambar 4. 38** Pengaruh Gender Terhadap Performansi

Dengan skema yang sama pula dilakukan uji untuk mengetahui adanya pengaruh gender terhadap performansi dari sistem yang telah dibuat. Dataset yang digunakan adalah dataset campuran dari Indonesia dan Graz. Dataset campuran perempuan disiapkan dengan mengambil 4 penutur perempuan dari masing masing dataset, begitupun berlaku sebaliknya dengan dataset campuran laki-laki. Gambar 4.38 merepresentasikan hasil uji untuk mengetahui pengaruh gender terhadap performansi, dimana dari gambar tersebut didapatkan untuk dataset campuran perempuan dan dataset campuran laki-laki menghasilkan EER sebesar 2,5%. Sehingga dapat dikatakan bahwa gender tidak mempengaruhi hasil performansi dari sistem identifikasi suara.

*Halaman ini sengaja dikosongkan*

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

1. Sistem berhasil memverifikasi dari campuran sekian pembicara dengan 20 file suara. Verifikasi dilakukan secara 2 arah yakni pengujian dengan penutur yang sama untuk mengukur indikator FRR dan penutur yang berbeda untuk mengukur indikator FAR, sehingga performasi EER dapat ditentukan saat nilai FRR sama dengan FAR.
2. Nilai EER menunjukkan performa diskriminasi dari program verifikasi. EER data graz sebesar 1,8% sedangkan nilai EER data Indonesia sebesar 10%. Perbedaan nilai EER yang cukup signifikan tersebut terjadi karena adanya perbedaan pada struktur formant dan FO (*Fundamental Frequency*) serta kualitas kedua dataset berbeda. Performansi dipengaruhi oleh jumlah speaker yang digunakan pada tahap training dan tidak dipengaruhi oleh gender.

#### **5.2 Saran**

- Gunakan babble noise yang sudah berbentuk sinyal gaussian sehingga sistem tidak salah mengenali penutur target sebagai salah satu penutur dalam babble noise.

*Halaman ini sengaja dikosongkan*

## DAFTAR PUSTAKA

- Anggraini, E., 2013. ON DEVELOPMENT OF INDONESIAN NATURAL SPEECH SYNTHESIS BASED ON HIDDEN MARKOV MODEL HMM. *Undergraduate Thesis of Physics Engineering*.
- Anon., n.d. *Signal Processing and Speech Communication Laboratory*. [Online] Available at: <https://www.spsc.tugraz.at/databases-and-tools/ptdb-tug-pitch-tracking-database-from-graz-university-of-technology.html>. [Accessed 12 December 2019].
- Boulkenafet, Z. M. et al., 2013. Forensic evidence reporting using GMM-UBM, JFA and I-vector methods : Application to Algerian Arabic dialect. *International Symposium on Image and Signal Processing and Analysis (ISPA 2013)*, pp. 397-402.
- D. Reynolds, T. Q. a. R. D., 2000. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10(1-3), pp. 19-41.
- Dehak, N., 2009. *Discriminative and Generative Approches for Long- and Short-Term Speaker Characteristics Modeling: Application to Speaker Verification*,. Montreal, Ph.D. thesis, 'Ecole de Technologie Sup'erieure.
- Dehak, N. et al., 2009. *Support Vector Machines versus Fast Scoring in the Low-Dimensional Total Variability Space for Speaker Verification*. Brighton, UK, s.n.
- Esfahani, O. G., 2018. *Deep Learning for i-Vector Speaker and Language Recognition*, Barcelona: Technical University of Catalonia, UPC.
- Furui, S., 2004. Fifty years of progress in speech and speaker recognition. *The Journal of the Acoustical Society of America*, pp. 116(4), 2497–2498.
- Garcia-Romero, D. et al., 2019. SpeakerRecognitionBenchmarkusingtheCHiME-5Corpus. *INTERSPEECH*, pp. 1506-1510.
- Giannakopoulos, T., 2009. A method for silence removal and segmentation of speech signals, implemented in Matlab. *University of Athens, Athens*, Volume 2.

- Hansen, J. H. L. & Hasan, T., 2015. Speaker Recognition by Machines and Humans: A tutorial review. *IEEE Signal Processing Magazine*, p. 32(6).
- Hernando, J. & Nadeu, C., 1997. CDHMM speaker recognition by means of frequency filtering of filter-bank energies. *In Proc. eurospeech*, p. 2363–2366.
- Hornsby, B. W., 2004. The Speech Intelligibility Index: What is it and what's it good for?. *The Hearing Journal*, pp. 10-17.
- Ibrahim, N. S. & Ramli, D. A., 2018. *I-Vector Expectation for Speaker Recognition Based on Dimensionality Reduction*. Nibong Tebal, Pulau Pinang 14300, Elsevier Ltd, pp. 1534-1540.
- J. Pelecanos, S. S., 2001. Feature Warping for Robust Speaker Verification. *ISCA Speaker Odyssey-The Speaker Recognition Workshop*.
- Littlewood, B. & A. A. L., 2003. *UK Essays*. [Online] Available at: <https://www.ukessays.com/> [Accessed Rabu November 2019].
- M. Mandasari, M. M. a. D. L., 2011. Evaluation of ivector speaker recognition systems for forensic application. *INTERSPEECH'11*, pp. 21-24.
- Mandasari, M. I., Firmanto, A. D. & Fathurrahman, F., 2009. Kalibrasi Rasio kemungkinan pada Sistem Rekognisi Pengucap Otomatis untuk Aplikasi Forensik di Indonesia. *Jurnal Linguistik Komputasional (JLK)*, 2(2).
- N. Dehak, P. K. R. D. P. D. a. P. O., 2011. Front end factor anlysis for speaker verification. *IEEE Trans. Audio, Speech, and Languge Processing*, 16(4), pp. 788-798.
- Nadeu , C., Macho, D. & Hernando, J., 2001. Time and frequency filtering of filterbank energies for robust HMM speech recognition. *Speech Communication*, pp. 34(1–2), 93–114..
- Nadiroh, A., 2019. *EFEKTIVITAS BABBLE SPEECH MASKER TERHADAP JUST NOTICEABLE DIFFERENCE UNTUK PENINGKATAN PRIVASI PADA RUMAH SAKIT DENGAN TIPOLOGI PERKANTORAN TAPAK TERBUKA*. Surabaya: Institut Teknologi Sepuluh Nopember.
- Nautsch, A., 2014. *SPEAKER VERIFICATION USING I-VECTORS Evaluation of text-independent speaker verification systems based on identity-vectors in*



*short and variant duration scenarios*, Daimlerstraße 32: Department of Computer Science .

- O. Glembek, L. B. N. B. a. P. K., 2009. Comparison of Scoring Methods used in Speaker Recognition with Joint Factor Analysis. *IEEE International Conference on Acoustics, Speech, and Signal Processing*.
- P. Kenny, P. O. N. D. V. G. a. P. D., 2008. A Study of Interspeaker Variability in Speaker Verification. *IEEE Transaction on Audio, Speech and Language*, Volume 16, p. 980–988.
- P.Kenny, G. B. a. P. D., 2005. Eigenvoice modeling with sparse training data. *IEEE Trans. Speech Audio Processing*, 13(3), pp. 345-354.
- Reynolds, D. A., Quatieri, T. F. & Dunn, R. B., 2000. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, pp. 10(1-3), 19–41.
- Reynolds, D. & Rose, R., 1995. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, p. 72–83..
- Rose, P., 1996. *Between- and within-speaker variation in the fundamental frequency of Cantonese citation tones*, in P. J. Davis and N. Fletcher (eds) *Vocal Fold Physiology – Controlling Complexity and Chaos: 307–24*. 1 ed. San Diego: Singular Publishing Group.
- Rose, P., 2002. *Forensic Speaker Identification*. 1 ed. USA and Canada : Taylor & Francis .

*Halaman ini sengaja dikosongkan*

## BIODATA PENULIS



**Roudhotul Jannah Rouf** merupakan nama lengkap penulis. Penulis dilahirkan di kota Sampang, Jawa Timur pada tanggal 25 April 1998 sebagai anak pertama dari dua bersaudara pasangan Abd. Roup (alm) dan Siti Hamitiyah (alm). Penulis telah menyelesaikan Pendidikan formal di SDN Krampon 1 (2004 – 2010), SMP Negeri 1 Torjun (2010 – 2013), SMA Negeri 1 Sampang (2013 – 2016), selanjutnya melanjutkan program Sarjana di Departemen Teknik Fisika ITS pada tahun 2016 – 2020. Selama menjadi mahasiswa, penulis aktif sebagai asisten peneliti di Laboratorium Vibrasi dan Akustik, Departemen Teknik Fisika ITS. Bagi pembaca yang memiliki kritik, saran, dan atau ingin berdiskusi lebih lanjut mengenai topik penelitian penulis, dapat menghubungi penulis melalui email [roudhotuljannah525@gmail.com](mailto:roudhotuljannah525@gmail.com) atau [roudhotul.rouf16@mhs.ep.its.ac.id](mailto:roudhotul.rouf16@mhs.ep.its.ac.id).