TEXT DEPENDENT SPEAKER VERIFICATION MENGGUNAKAN I-VECTOR EXTRACTION DAN GAUSSIAN MIXTURE MODEL

TEXT DEPENDENT SPEAKER VERIFICATION USING I-VECTOR EXTRACTION AND GAUSSIAN MIXTURE MODEL

Viko Adi Rahmawan[1], Achmad Rizal, S.T., M.T.[2], Ratri Dwi Atmaja,S.T.,M.T.[3]
[1,2,3] Departemen Elektro dan Komunikasi, Fakultas Teknik Elektro, Universitas Telkom
[1]**vikoadi@gmail.com**, [2] **achmadrizal@telkomuniversity.ac.id**, [3] **ratridwiatmaja@telkomuniversity.ac.id**

ABSTRAK

Dibandingkan metode verifikasi identitas biometrik lain, speaker verification memiliki kelebihan yaitu telah banyaknya perangkat mikrofon tersemat pada berbagai perangkat. Hal tersebut tentu menarik karena memungkinkan untuk ditambahnya metode verifikasi ini melalui pembaruan perangkat lunak tanpa memerlukan perangkat keras lain.

Penelitian mengenai speaker verification telah banyak dilakukan beriringan dengan penelitian speaker recognition lainnya. Speaker recognition biasanya menggunakan MFCC (Mel Frequency Cepstral Coefficients) untuk melakukan pengenalan suara. Dalam tugas akhir ini akan dilakukan pengetesan akurasi sebuah sistem Text-Dependent Speaker Verification (TD-SV) yang menggunakan i-vector extraction dan Gaussian Mixture Model (GMM).

I-Vector extraction diketahui memiliki akurasi yang lebih baik pada aplikasi Speaker Recognition dibandingkan dengan MFCC. Penelitian ini dapat menunjukkan berapa besar akurasi TD-SV menggunakan i-vector extraction dan GMM. Mnggunakan i-vector extraction dan GMM, didapatkan *False Rejection Rate* sebesar 60%, *False Acceptance Rate* sebesar 0% dan *Error Rate* sebesar 12%.

Kata kunci : text dependent speaker recognition, i-vector, gaussian mixture model

ABSTRACT

Compared to other biometric identity verification, speaker verification has some advantages, the most obvious one is the inclusion of microphone on a lot of devices. That reason makes speaker verification interesting, as it enable the addition of this new verification method through software update, without the need of additional hardware.

Research in speaker verification have been done in conjunction with other speaker recognition research. In speaker recognition research it is usually done using MFCC (Mel Frequency Cepstral Coefficients) to recognize speaker identity. In this undergraduate paper, an experiment will be conducted to understand how is the accuration of Text Dependent Speaker Verification (TD-SV) using I-vector extraction and Gaussian Mixture Model (GMM).

I-vector extraction is known to have better accuration compared to MFCC in Speaker Recognition aplication. This experiment can show how is the accuration of TD-SV using i-vector extraction and GMM in Speaker Verification compared to MFCC approach. By incorporating i-vector extraction, we achieve False Rejection Rate as low as 60%, False Acceptance Rate at 0% and Error Rate at 12%.

Keyword : text dependent speaker recognition, i-vector, gaussian mixture model

1 . Pendahuluan

Speaker Verification merupakan sebuah solusi pengenalan biometrik yang murah dan mudah untuk diterapkan pada perangkat-perangkat yang telah memiliki mikrofon sebagai perangkat masukannya. Speaker verification dapat diterapkan tanpa membutuhkan penambahan perangkat keras [1].

Tugas Akhir ini berkonsentrasi pada pemodelan pengenalan pembicara dengan memanfaatkan I-Vector backend pada Text Dependent Speaker Verification (TD-SV).

Metode Mel Frequency Cepstral Coefficients (MFCC) dan variasinya telah umum digunakan sebagai feature extraction dalam speaker verification [2]. Dalam penelitian ini akan digunakan I-Vector Extraction untuk melihat akurasi algoritma ini dibandingkan MFCC. I-Vector Extraction dipilih karena telah terbukti menghasilkan ketelitian yang lebih baik dibandingkan MFCC dalam speaker recognition [3].

2 . SISTEM VERIFICTION

2.1 *Speaker Verifiction*

Speaker Verification merupakan salah satu cabang dari speaker recognition dengan suatu sifat yang spesifik.

Dalam pembandingan model suara, dapat dipastikan bahwa modelnya tidak dapat sama persis bahkan untuk *test speaker* dan *target speaker* yang dibuat oleh orang yang sama. Hal ini disebabkan oleh berbagai macam ketidakpastian dan sifat suara manusia yang berubah-ubah. Dalam mengatasi hal ini ada dua macam pendekatan yang sering digunakan yaitu *Universal Background Model* (UBM) dan *Cohort Model* (CM).

Ide dasar dari UBM adalah dengan membandingkan *test speaker* dengan model yang didapat dari suatu populasi yang besar. Jika *test speaker* lebih dekat dengan rata-rata populasi dibandingkan dengan *target speaker*, maka kemungkinan dia bukanlah *target speaker* [4].

Sedangkan pada CM, *test speaker* tidak dibandingkan dengan rata-rata suatu populasi, tetapi hanya pada model kelompoknya saja. Anggota kelompok tersebut adalah orang yang bersuara mirip dengan *target speaker*. Filosofi pada metode ini adalah jika *test speaker* lebih dekat ke *target speaker* dibandingkan dengan kelompoknya, maka kemungkinan *test speaker* sama dengan *target speaker* [5].

2.2 Text Dependent Speaker Verification

TD-SV merupakan salah satu modalitas *speaker verification* dimana diperlukan kata yang sama dalam proses pelatihan dan tesnya. Secara umum *Text Dependent Speaker Recognition* (TD-SR) menghasilkan akurasi pengenalan yang lebih bagus dibandingkan *Text Independent Speaker Recognition* (TD-SR) terutama ketika digunakan kata yang pendek dalam pelatihan dan testingnya [6]. *Equal Error Rate* dari TI-SR lebih tinggi dibandingkan TD-SR pada kasus yang sama. Sistem TD-SR memerlukan 2-3 detik durasi suara pelatihan dan pengenalan untuk mendapatkan hasil yang bagus. Akan tetapi TD-SR memerlukan lebih banyak data latih dibandingkan TI-SR [7].

2.3 GMM (Gaussian Mixture Model)

GMM merupakan pemodelan ciri menggunakan *density estimator* dan merupakan salah satu jenis yang paling sering digunakan. Dalam metodenya, distribusi vektor x dimodelkan menggunakan campuran dari M Gaussian.

$$P(x|m)=\sum_{i=1}^{m} a_i \frac{1}{(2\Pi)^{D/2}|\sum i|^{1/2}} \exp\left(\frac{-1}{2}(x-\mu_i)^T \sum_{i}^{-1}(x-\mu_i)\right) \qquad \textbf{(2.2)}$$

dimana μ_i , $\sum i$ menunjukkan rata-rata dan kovarian dari campuran i^{th} . Jika diberikan data latih x_1 , x_2 ,... x_n dan banyaknya campuran M , parameter μ_i , $\sum I$, a_i dilatih dengan menggunakan maksimalisasi ekspektasi. Dalam pengenalan, input suara diekstrak kembali ke dalam baris x_1 , x_2 ... x_n yang merupakan jarak dari baris tersebut dengan model yang didapat dari perhitungan kemiripan dari data yang diberikan. Model yang memiliki kemiripan yang tinggi akan menunjukkan identitas pembicara [8].

2.4 I-Vector Feature Extraction

Dari hasil supervektor GMM masih memiliki masalah yaitu masih terdapatnya efek kanal pada supervektor tersebut. Perlu dicari vektor identitas (I-Vector) untuk menghilangkan efek kanal dan didapatkan ciri suara yang teliti.

2.5 Nilai I-Vector

I-Vector dapat dihitung sebagai berikut [9]

$$\omega=\left(I+T^t \Sigma^{-1} N(u) T\right)^{-1}.T^t \Sigma^{-1} \tilde{F}(u)$$

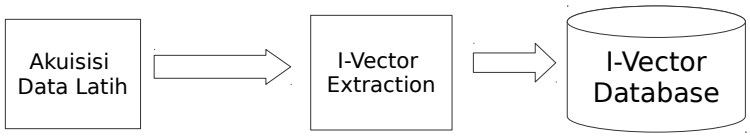
s = conversation side supervector

T = total-variability matrix

ω = i-vector

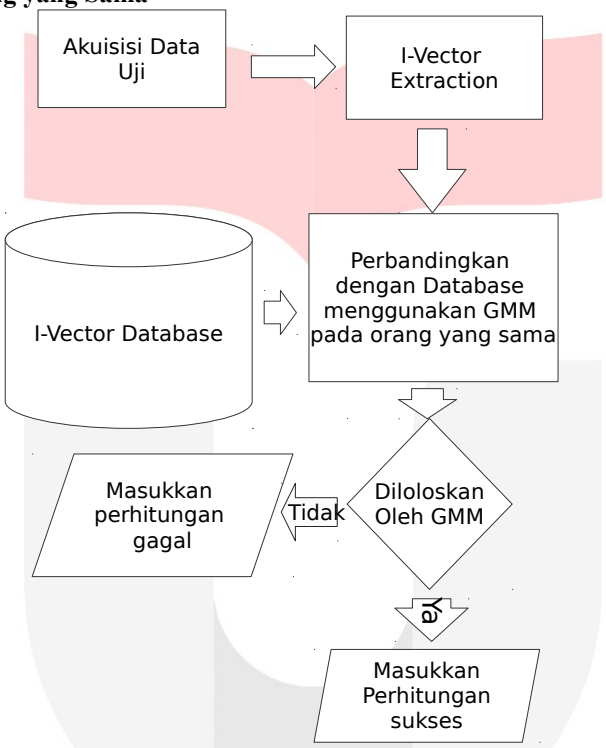
2.6 Desain Sistem

Sistem ini perlu dilakukan pada awal percobaan untuk mendapatkn data model dari masing-masing orang. Data I-Vector akan disimpan pada database I-Vector untuk selanjutnya digunakan pada proses pengujian.



Gambar 2.1: Pengambilan database data latih

2.7 Sistem Uji untuk Orang yang Sama



Gambar 2.2: Pengujian Pada Orang yang Sama

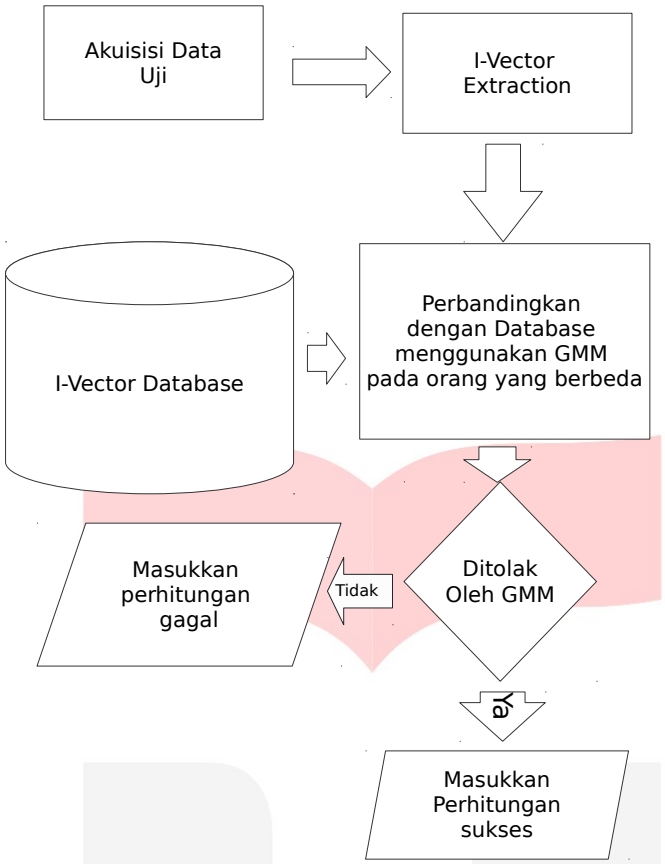
Setelah *database* model didapat maka akan dilakukan pengujian terhadap *test speaker*. Untuk mendapatkan nilai FRR maka akan dilakukan pengujian terhadap data uji dengan model yang didapat dari orang yang sama. Sistem yang baik harus meloloskan tes ini.

2.8 Sistem Uji untuk orang yang Berbeda

Untuk penggunaan keamanan sangat tidak diinginkan orang lain dapat lolos dari proses verifikasi ini. Dengan memperoleh hasil dari percobaan ini dapat dihitung FAR dari sistem verifikasi I-Vector yang telah dibuat.

2.9 Evaluasi Akurasi

Selanjutnya akan dilakukan perhitungan akurasi ketelitian untuk *I-Vector Extraction* dan GMM. Untuk melihat akurasi dari sistem yang telah dibangun akan digunakan tiga buah parameter yaitu False Acceptance Rate, False Rejection Rate dan Error Rate.



Gambar 2.3: Pengujian Pada Orang yang Berbeda

2.10 Database Suara

Sebagai data latih dan data uji akan digunakan rekaman suara yang diambil dari lima orang yang berbeda. Untuk masing-masing orang akan diambil sepuluh suara untuk masing-masing panjang kata.

Panjang kata yang dipakai adalah sebanyak satu hingga tiga buah kata. Tiga kata uji ini dipilih agar dapat ditentukan berapa panjang kata yang optimal digunakan untuk SV menggunakan Algoritma I-Vector dan GMM ini.

2.11 Bahasa Pemrograman dan Toolbox

Sistem pelatihan dan pengetesan akan diimplementasikan dalam bahasa pemrograman Python. Dalam jurnalnya Glover, John dkk [10].

Dalam sistem latih dan ujinya akan digunakan *toolbox* pemrosesan sinyal bernama Spear [11]. Spear merupakan *toolbox* yang bersumber terbuka dan mudah dikembangkan untuk *speaker recognition*. *Toolbox* ini dikembangkan di atas Bob, pustaka pemrosesan sinyal dan *machine learning* yang bebas. Spear mengimpelementasi satu proses utuh *speaker recognition*, termasuk semua tahap pemrosesan, dari *feature extractor* hingga tahap penentuan dan evaluasi.

3 . PENGUJIAN SISTEM DAN HASIL

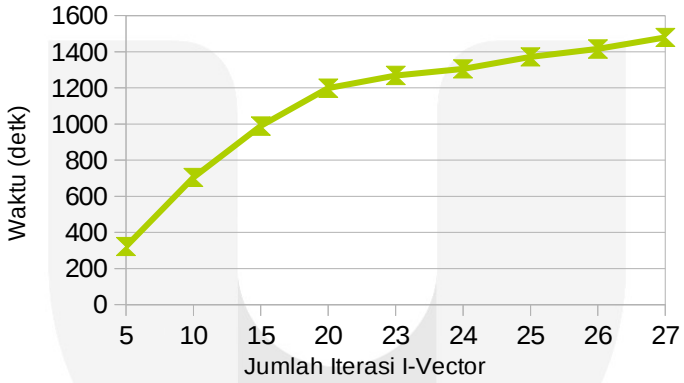
3.1 Pengujian terhadap berkas Development

Dalam menentukan parameter untuk iterasi minimal untuk I-Vector maka dibandingkan sample suara yang sama pada proses pemodelan dan pengujian. Untuk hal itu dilakukan pengujian iterasi hingga didapati nilai yang optimal.

Tabel 3.1: Hasil pengujian sampel development

Iterasi I-Vector	Tes Penerimaan	Missed Identification	FRR	Tes Penolakan	False Alarm Identification	FAR	ER	Waktu (detik)
5	25	0	0.00%	100	0	0.00%	0.00%	322
10	25	0	0.00%	100	0	0.00%	0.00%	705
15	25	0	0.00%	100	0	0.00%	0.00%	989
20	25	0	0.00%	100	0	0.00%	0.00%	1198
23	25	0	0.00%	100	0	0.00%	0.00%	1269
24	25	0	0.00%	100	0	0.00%	0.00%	1305
25	25	0	0.00%	100	0	0.00%	0.00%	1372
26	25	0	0.00%	100	0	0.00%	0.00%	1415
27	25	0	0.00%	100	0	0.00%	0.00%	1479

Dari tabel di atas dapat dilihat bahwa hanya dengan iterasi yang minimal sekali saja algoritma I-Vector sudah dapat bekerja dengan baik dan dapat memilih sampel katanya sendiri.



Gambar 3.1: Waktu perhitungan *development set*

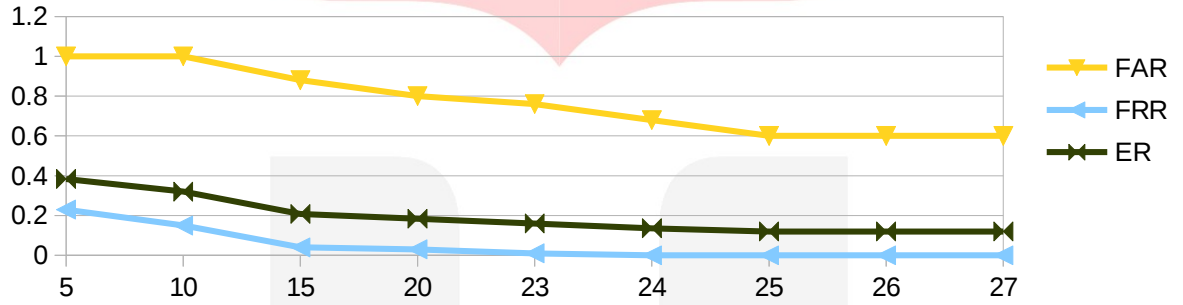
3.2 Proses Pengujian Berkas Evaluation terhadap Iterasi I-Vector

Berkas evaluation merupakan lima buah sampel suara uji yang berbeda dengan sampel suara saat digunakan sebagai model. Berkas *development* diuji untuk mencari nilai iterasi yang optimal terhadap nilai EER yang diinginkan. Untuk pengujian iterasi akan dilakukan perngujian dengan selisih iterasi yang berubah-ubah.

dapat dilihat bahwa pada iterasi sebanyak 24 kali kesalahan penolakan telah mendapatkan False Alarm yang baik, dimana tidak ada penyusup yang di loloskan. Sedangkan pada iterasi selanjutnya nilai FAR terlihat tidak ada perubahan dan stabil di angka 15.

Tabel 3.2: Hasil pengujian terhadap perubahan iterasi I-Vector

Iterasi I-Vector	**Tes Penerimaan**	**Missed Identification**	**FAR**	**Tes Penolakan**	**False Alarm Identification**	**FRR**	**ER**	**Iterasi I-Vector**
5	25	25	100.00%	100	23	23.00%	38,40%	5
10	25	25	100.00%	100	15	15.00%	32,00%	10
15	25	22	88.00%	100	4	4.00%	20,80%	15
20	25	20	80.00%	100	3	3.00%	18,40%	20
23	25	19	76.00%	100	1	1.00%	16,00%	23
24	25	17	68.00%	100	0	0.00%	13,60%	24
25	25	15	60.00%	100	0	0.00%	12,00%	25
26	25	15	60.00%	100	0	0.00%	12,00%	26
27	25	15	60.00%	100	0	0.00%	12,00%	27



Gambar 3.2: Hasil percobaan terhadap perubahan iterasi I-Vector

3.3 Proses Pengujian Terhadap Panjang Kata yang Berbeda

Data latih berupa rekaman penyebutan tiga buah kata masing-masing sebanyak lima kali. Kata diucapkan oleh lima orang yang berbeda.

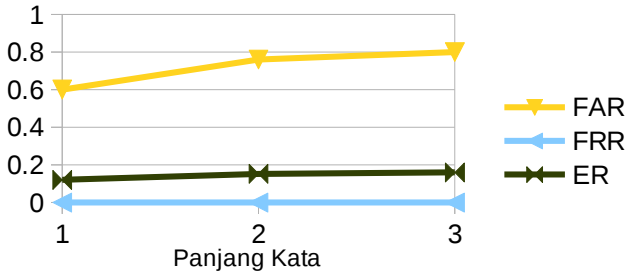
Kata yang diucapkan adalah

1. “satu” (satu kata)
2. “satu, dua” (dua kata)
3. “satu, dua, tiga” (tiga kata)

Dalam Pengujian ini digunakan Iterasi sebanyak 25 kali untuk semua panjang kata. Dapat dilihat bahwa ternyata dengan digunakannya kata yang lebih panjang justru memperburuk hasil verifikasi. Hasil verifikasi terbaik adalah pada kata sepanjang 1 kata saja yaitu FAR sebesar 60% FRR sebesar 0%, dan ER sebesar 12%.

Tabel 3.3: Hasil pengujian terhadap panjang kata

Panjang Kata	Tes Penerimaan	Missed Identificaiti	FAR	Tes Penolakan	False Alarm Idt	FRR	ER
1	25	15	60.00%	100	0	0.00%	12,00%
2	25	19	76.00%	100	0	0.00%	15,20%
3	25	20	80.00%	100	0	0.00%	16,00%



Gambar 4.1: Hasil percobaan pada beberapa panjang kata

4 . Kesimpulan dan Saran

4.1 Kesimpulan

Dari hasil percobaan di atas dapat dilihat berapa nilai EER dari masing-masing panjang kata. Dari percobaan diatas dapat disimpulkan bahwa konfigurasi *speaker verification* yang terbaik adalah ketika digunakan iterasi I-Vector sebanyak 25 kali dan panjang kata sebanyak satu kata yaitu dengan nilai FRR sebesar 60%, FAR sebesar 0% dan ER sebesar 12%.. Dengan hasil FAR yang baik di atas maka sistem I-Vector ini cocok digunakan sebagai metode verifikasi keamanan karena dapat melindungi dari percobaan akses oleh orang yang tidak berwenang. Meskipun demikian masih perlu diperhatikan bahwa FRR nya tinggi, sehingga pengguna mungkin memerlukan pengulangan verifikasi sebanyak beberapa kali.

5 . Saran

Untuk penelitian selanjutnya perlu diidentifikasi penyebab tingginya penolakan ini. Penolakan dapat saja terjadi akibat buruknya channel sistem, banyaknya noise suara, ataupun hal-hal yang lain. Dengan dilakukan identifikasi diharapkan kedepannya dapat dicari penyelesaian dari masalah di atas dan dihasilkan algoritma/kumpulan algoritma baru yang tidak hanya aman melindungi sistem namun juga nyaman digunakan oleh penggunanya.

DAFTAR PUSTAKA:

[1] Chandana Krishna, Dr. Hariprasad S.A. Speaker Verification. IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), 2013
[2] Todor Ganchev, Nikos Fakotakis, George Kokkinakis . Comparative Evaluation of Various MFCC Implementations on the Speaker Verification Task. Wire Communications Laboratory, University of Patras,, 2009
[3] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann and P.Dumouchel. I-Vector/PLDA Variants for Text-Dependent SpeakerRecognition. Centre de Recherche Informatique de Montreal (CRIM), 2013
[4] Beigi, H.S. Adaptive and Learning-Adaptive Control Techniques based on an Extension of the Generalized Secant Method. Intelligent Automation and Soft Computing Journal 3(2), 1997
[5] Akaike, H. A new look at the statistical model identification. IEEE Transactions on Auto-matic Control 19(6), 1974
[6] Naik, J. M. Speaker verification: A tutorial. IEEE Comm. Mag. Vol. 28, No. 1, 1990
[7] Peacocke, R. D. and Graf, D. H. An Introduction to Speech and Speaker Recognition. IEEE Trans Computer . Vol. 23. No. 8, 1990
[8] Jeet Kumar, Om Prakash Prabhakar, Navneet Kumar Sahu. Comparative Analysis of Different Feature Extractionand Classifier Techniques for Speaker Identification Systems: A Review. International Journal of Innovative Research in Computerand Communication Engineering Vol. 2, Issue 1, 2014
[9] N.Dehak, dkk. Front-end factor analysis for speaker verification[J]. Audio, Speech, and Language Processing, IEEE Transactions on, 2011
[10] John Glover, Victor Lazzarini, dan Joseph Timoney. Python for Audio Signal Processing. National University of Ireland,
[11] Khoury, E., El Shafey,L. and Marcel,S. Spear: An open source toolbox for speaker recognition based on Bob. IEE intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP), 2014