

Pencirian Itik Jantan dan Betina Menggunakan Metode *Time-Frequency Principal Component*

Stefan Eka Setiawan
Jurusan Teknik Elektro - Fakultas Teknik Undip
Jl. Prof. Sudharto, Tembalang – Semarang
Email : ipank23@yahoo.com

Abstrak - Metode *Time-Frequency Principal Component* (TFPC), yaitu metode yang mengambil pola waktu-frekuensi sinyal (spektral) dan mencari komponen utama dari urutan vektor spektral sinyal dengan perhitungan *Principal Component Analysis* (PCA). Komponen utama tersebut akan menjadi suatu karakteristik pengucap dan pembandingan suatu sinyal masukan. Pertama yang dilakukan adalah merekam suara itik yang telah diketahui jenis kelaminnya untuk dijadikan basisdata. Kedua, menguji program yang telah dibuat dengan masukan dari basisdata. Ketiga, menguji data baru untuk diketahui persentase pengenalannya. Terakhir, menentukan parameter terbaik rata-rata dari data baru. Rata-rata persentase pengenalan data baru sebelum penambahan basisdata adalah 71,9255%, sedangkan rata-rata persentase pengenalan data baru setelah penambahan basisdata adalah 76,0019 %.

Kata kunci : pola waktu-frekuensi, komponen utama, itik jantan, itik betina.

I. PENDAHULUAN

Dalam ilmu pengetahuan dan teknologi khususnya bidang komunikasi, pengolahan sinyal memegang peranan yang penting. Penelitian yang intensif dalam bidang pengolahan sinyal menyebabkan teknologi komunikasi berkembang dengan pesat, salah satunya pada pengenalan pengucap (*speaker identification*).

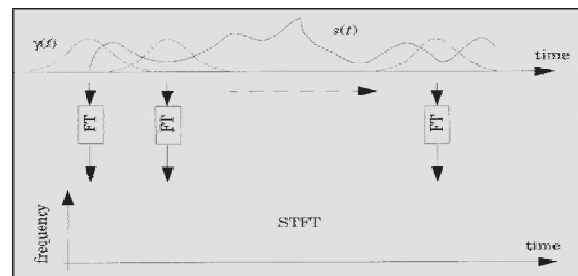
Salah satu teknik yang dapat digunakan untuk mengetahui karakteristik pengucap adalah metode *Time-Frequency Principal Component* (TFPC), yaitu metode yang mengambil pola waktu-frekuensi sinyal (spektral) dan mencari komponen utama dari urutan vektor spektral sinyal dengan perhitungan *Principal Component Analysis* (PCA). Komponen utama tersebut akan menjadi suatu karakteristik pengucap dan pembandingan suatu sinyal masukan.

Berdasarkan beberapa parameter sinyal ucapan, dapat dibuat suatu sistem pengenalan pengucap. Sistem pengenalan pengucap pada dasarnya adalah pembandingan suatu parameter sinyal ucapan yang ingin dikenali (sinyal kunci) dengan parameter sinyal ucapan lain. Pada proses pencocokan (*template matching*) digunakan metode DTW (*Dynamic Time Warping*) yang membandingkan hasil TFPC dari masing-masing pengucap dengan sinyal masukan yang akan dikenali. Identifikasi ditentukan dengan menghitung matriks beda antarsinyal. Matriks beda dengan jarak penyimpangan terkecil merupakan identitas dari sinyal tersebut.

II. DASAR TEORI

2.1 Short Time Fourier Transform

Salah satu penggunaan FFT (*Fast Fourier Transform*) adalah STFT, yaitu suatu fungsi yang terdiri dari 2 variabel, yaitu variabel waktu dan variabel frekuensi yang menyatakan bahwa spektrum sinyal memungkinkan bervariasi setiap waktu. Untuk menunjukkannya adalah dengan memotong sinyal menjadi beberapa *frame* yang kemudian dilakukan perhitungan FFT untuk tiap *frame*.



Gambar 1 Short Time Fourier Transform (STFT)

Secara matematis ditunjukkan dengan persamaan :

$$STFT(k) = \sum_{n=0}^{N-1} x(n) w(n) e^{-j2\pi nk / N} \dots\dots\dots (1)$$

Parameter yang harus ditentukan ialah panjang *frame* dan tipe *window* yang digunakan. Perhitungan STFT yang pertama ialah mengalikan fungsi *windowing* $\gamma(t)$ dengan sinyal $s(t)$ dan yang kedua ialah menghitung FFT hasil perkalian. Karena fungsi *windowing* $\gamma(t)$ adalah durasi waktu pendek, nilai *Fourier Transform* (FT) ialah spektrum tiap *frame*. Pengulangan proses tersebut akan didapatkan spektrum sinyal bervariasi setiap waktu.

2.2 Principal Component Analysis

Suatu teknik untuk mengenali pola data dengan menggunakan kesamaan dan perbedaannya, serta meringkaskan jumlah data tanpa menghilangkan informasi dalam suatu data. Penjelasan langkah-langkah pada PCA adalah sebagai berikut.

1. Mencari Harga Kovarians

Kovarians menggambarkan hubungan diantara 2 data atau lebih.

$$X_k = \frac{1}{M} \sum_{t=k+1}^M (x_t - \bar{x}) \cdot (x_{t-k} - \bar{x}) \dots\dots\dots (2)$$

\bar{x} merupakan rata-rata dari sekumpulan data pada x , sedangkan \bar{y} merupakan rata-rata dari sekumpulan data y . Notasi M menunjuk jumlah data pada kedua variabel. Batas nilai k adalah $0 \leq k < M$, maka nilai kovarians adalah $X_0, X_1, X_2, \dots, X_k$. Nilai-nilai kovarians yang diperoleh kemudian dibentuk menjadi matriks Toeplitz, yang mempunyai ciri mempunyai nilai yang sama pada diagonal utamanya. Alasan penggunaan matriks Toeplitz adalah karena untuk membentuk matriks persegi sehingga akan memudahkan penghitungan nilai eigen dan vektor eigen pada langkah selanjutnya.

2. Mencari Nilai dan Vektor Eigen

Nilai eigen berkaitan dengan matriks. Konsep dasarnya adalah jika $A = [a_{jk}]$ adalah suatu matriks $n \times n$ membentuk persamaan sebagai berikut :

$$Ax = \lambda x \dots\dots\dots (3)$$

Vektor nol ($x = 0$) ialah penyelesaian untuk sembarang nilai λ . Nilai λ yang memberikan penyelesaian $x \neq 0$ disebut nilai eigen atau nilai karakteristik dari matriks A . Penyelesaian padanannya, yaitu untuk $x \neq 0$ disebut vektor eigen atau vektor karakteristik. Persamaan di atas dapat ditulis ulang sebagai suatu persamaan homogen adalah sebagai berikut dengan I adalah matriks identitas yang berorde sama dengan matriks A .

$$(A - \lambda I)x = 0 \dots\dots\dots (4)$$

Karena persamaan diatas homogen, maka persamaan selalu mempunyai penyelesaian trivial ($x=0$). Sedangkan penyelesaian non trivial ($x \neq 0$) ada jika determinan dari matriks adalah nol. Setelah nilai eigen ditemukan, maka vektor eigen yang berkaitan dengan nilai eigen diperoleh dengan menyelesaikan persamaan homogen. Dengan menggunakan metode eliminasi Gauss, diperoleh vektor eigen untuk tiap nilai eigennya. Banyaknya vektor eigen yang diperoleh dapat disusun dalam bentuk matriks kolom.

3. Menentukan komponen utama

Vektor eigen yang didapat masing-masing sepadan dengan nilai eigen. Vektor eigen yang sepadan dengan nilai eigen terbesar disebut komponen utama (*principal component*). Jadi langkah terakhir dalam PCA adalah menentukan komponen utama sebagai vektor ciri.

2.3 Time-Frequency Principle Component

Ide TFPC adalah mengambil sedikit pola waktu-frekuensi urutan vektor dari keseluruhan urutan. Maksudnya ialah mencari spektrum sinyal utama pada durasi waktu pendek. Apabila pola telah diambil maka akan digunakan untuk mencocokkan dengan pola pada basisdata. Teknik ini kemudian dapat diaplikasikan pada vektor baru sebagaimana dilakukan seperti biasa pada vektor spektral atau vektor cepstral ataupun sembarang representasi vektor dari sinyal asli. Sebagai contoh, kita

mengaplikasikan TFPC pada pengenalan pengucap dan mengambil pola waktu-frekuensi untuk setiap pengucap dari basisdata.

$\{x_t\} 1 \leq t \leq M$ adalah sebuah urutan vektor spektral. Jika X_0 merupakan matriks kovarians dari urutan $\{x_t\}$ maka :

$$X_0 = \frac{1}{M} \sum_{l=1}^M (x_l - \bar{x}) \cdot (x_l - \bar{x})^T \dots\dots\dots (5)$$

dan X_k merupakan matriks kovarians tertinggal pada orde k :

$$X_k = \frac{1}{M} \sum_{l=k+1}^M (x_l - \bar{x}) \cdot (x_{l-k} - \bar{x})^T \dots\dots\dots (6)$$

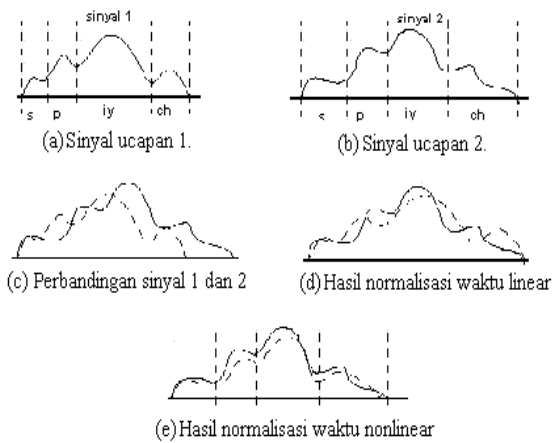
Dimensi matriks kovarians dan matriks kovarians tertinggal adalah $p \times p$. Kita kemudian mendefinisikan matriks baru. X_{2q+1} , yaitu :

$$X_{2q+1} = \begin{bmatrix} X_0 & X_1 & \dots & X_{2q} \\ X_1^T & X_0 & \dots & X_{2q-1} \\ \vdots & \vdots & \vdots & \vdots \\ X_{2q}^T & X_{2q-1}^T & \dots & X_0 \end{bmatrix}$$

2.4 Dynamic Time Warping

Merupakan salah satu metode pencocokan *template* (*template matching*). Algoritma DTW pada dasarnya merupakan program dinamik. DTW melakukan perbandingan secara dinamik antara matriks parameter sinyal uji dengan sinyal referensi.

Representasi bentuk gelombang sinyal ucapan dari sebuah kata bisa berbeda sama sekali, hal ini disebabkan karena adanya perbedaan cara pengucapan setiap individu, dan sebagainya. Meskipun diucapkan oleh pengucap yang sama, perbedaan tetap saja bisa terjadi karena kecepatan pengucapan, keras-lemah bunyi, dan penekanan yang berbeda. Beberapa bentuk gelombang bisa saja mempunyai bentuk yang panjang dan mempunyai amplitude yang tinggi, sedangkan gelombang lain mempunyai bentuk yang lebih pendek dan amplitude yang lebih kecil. Gambar 2 memberikan contoh mengenai perbedaan bentuk gelombang dari sebuah kata. Permasalahan dalam membandingkan antara sinyal 1 dengan sinyal 2 adalah bahwa panjang kedua sinyal tersebut berbeda. Dari pembahasan sebelumnya, kita tahu bahwa sinyal ucapan dapat direpresentasikan sebagai deret vektor yang berisi ciri-ciri ucapan. Misalnya vektor sinyal 1 dituliskan sebagai $\vec{a} = \{ a_1, a_2, a_3, \dots, a_I \}$ dan vektor sinyal 2 dituliskan sebagai $\vec{r} = \{ r_1, r_2, r_3, \dots, r_J \}$, karena panjang sinyal 1 dan 2 berbeda, maka besar I dan J juga berbeda.



Gambar 2 Perbandingan antara 2 sinyal dari sebuah kata “speech”

Beda antarsinyal ditulis dengan Persamaan

$$D(r, a) = \frac{r_m a_n}{\sqrt{\sum_{m=1}^i r_m^2} \cdot \sqrt{\sum_{n=1}^j a_n^2}} \dots\dots (7)$$

Hasil normalisasi waktu linear diilustrasikan pada Gambar 2d. Gambar 2d menunjukkan bahwa laju pengucapan sebanding dengan waktu pengucapan, dan tidak terpengaruh oleh bunyi yang diucapkan. Tetapi pada kondisi nyata, normalisasi yang baik juga harus mengubah durasi sinyal yang ingin dibandingkan sesuai dengan karakter-karakter dari sinyal pembandingan agar kesesuaian antar keduanya lebih baik, sehingga beda antar kedua sinyal, $D(f, \check{a})$, dapat diminimalkan. Contoh dari normalisasi waktu non-linear ini adalah seperti pada Gambar 2e.

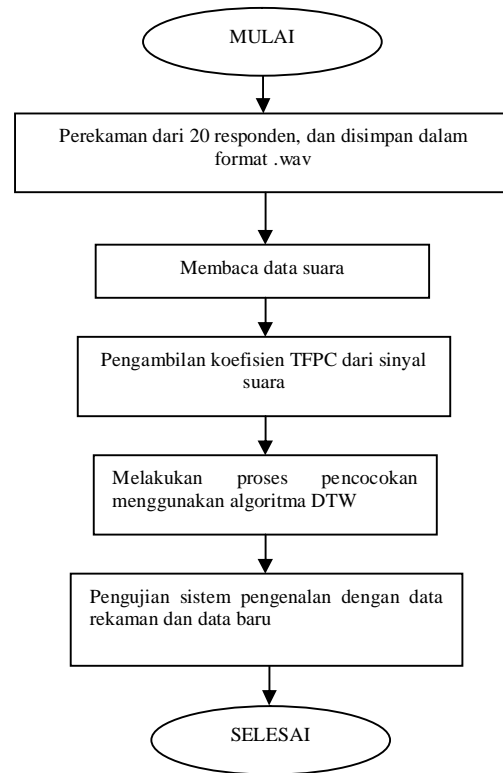
2.5 Program Dinamis

Program dinamis merupakan algoritma yang biasa digunakan untuk menyelesaikan pengambilan keputusan permasalahan-permasalahan sekuensial. Pada sistem pengenalan pengucap, program dinamis digunakan untuk mendapatkan jalur kecocokan antar dua *template* sinyal masukan yang telah dipetakan dan membentuk suatu matriks beda.

Beda antar sinyal dituliskan dalam *grid* matriks seperti pada Gambar 4. Sinyal yang dibandingkan, \check{a} , dituliskan pada sumbu x, dan sinyal pembandingan, r , dituliskan pada sumbu y. Titik pada koordinat (0,0) disebut titik awal/*original node*, yang merupakan awal dari jalur (*path*) yang akan dihitung. Titik/*node* didefinisikan sebagai *frame* jarak antara vektor ciri sinyal $a(i)$ dengan $r(j)$.

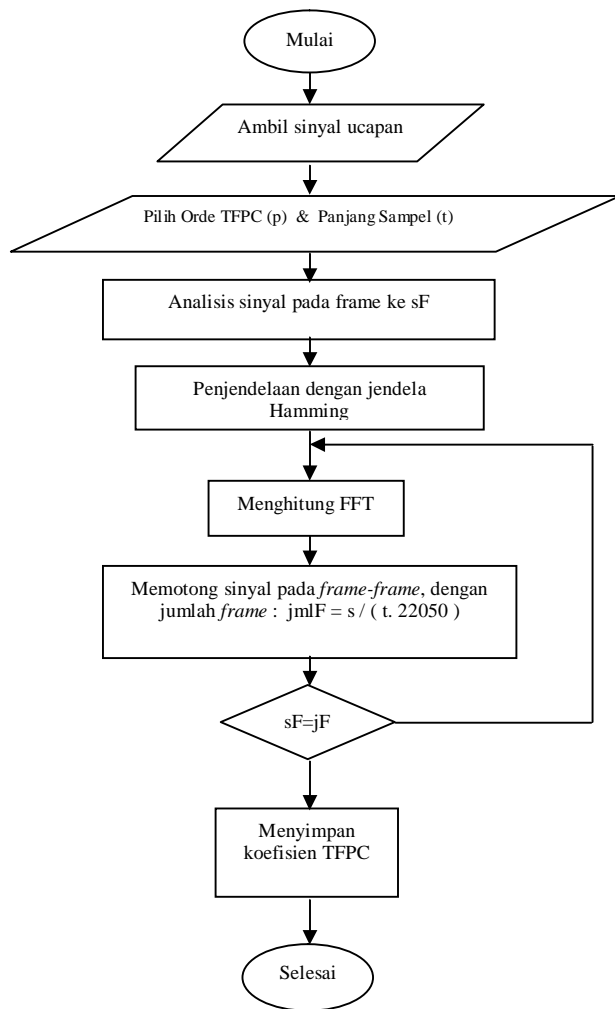
III. PERANCANGAN SISTEM

Sistem pengenalan pengucap dapat diwujudkan ke dalam suatu perangkat lunak (*software*) menggunakan bahasa pemrograman Matlab 6.5. Sistem pengenalan pengucap dalam perancangan ini terdiri atas tiga bagian. Bagian pertama adalah pembacaan dan normalisasi sinyal ucapan masukan, bagian kedua adalah pengambilan komponen sinyal ucapan, bagian terakhir adalah proses pencocokan dan pengambilan keputusan. Dalam pembuatan program simulasi pengenalan pengucap, perancangan sistem merupakan suatu tahapan penting agar nantinya diperoleh sistem yang sistematis, terstruktur dan rapi, sehingga diperoleh hasil yang optimal dan berjalan sesuai dengan yang dikehendaki. Diagram alir dari sistem pengenalan pengucap dapat dilihat pada Gambar 3.



Gambar 3 Diagram alir program pengenalan pengucap

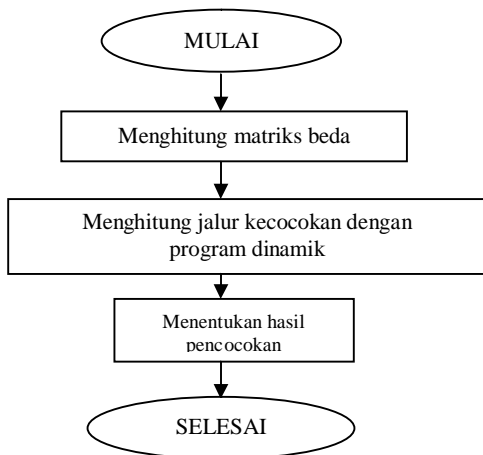
Sistem pengenalan pengucap diawali dengan perekaman sinyal ucapan dari masing-masing 10 responden yang akan dimasukkan ke dalam basisdata. Hasil perekaman kemudian dibaca untuk memperoleh bentuk diskret atas sinyal ucapan, sehingga dapat mempermudah pengolahan selanjutnya. Pembacaan data suara diikuti dengan normalisasi sinyal ucapan. Sinyal ucapan kemudian diambil cirinya sampai didapatkan koefisien TFPC. Hasil pengambilan kemudian masuk pada proses pencocokan menggunakan algoritma DTW. Pengujian data rekaman dan data baru dilakukan setelah didapatkan masing-masing koefisien TFPC.



Gambar 4 Diagram alir program pengambilan sinyal

3.1 Pencocokan Template (Template Matching)

Proses pencocokan *template* dilakukan melalui dua tahap, yaitu penghitungan beda antar-*template* menggunakan DTW dan pengambilan keputusan.



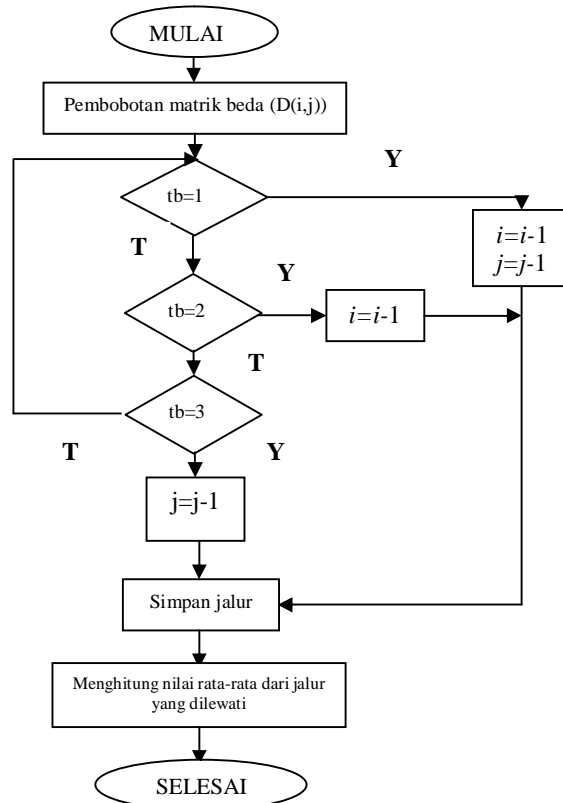
Gambar 5 Diagram alir *template matching*.

Pencocokan *template* dimulai dengan penghitungan matriks beda (pemetaan) dari komponen sinyal 1 dengan komponen sinyal 2. Berdasarkan matriks beda tersebut dihitung (ditelusuri) jalur kecocokan terbaik dengan menggunakan program dinamik. Langkah selanjutnya adalah penentuan hasil pencocokan (pengambilan keputusan) berdasarkan hasil program dinamik.

3.2 Penghitungan Matriks Beda

Seperti telah disebutkan di atas, penghitungan beda antar *template* dilakukan dengan menggunakan program DTW (*dynamic time warping*). Pada program DTW, ada dua tahap yang harus dilakukan, yaitu membentuk pemetaan antar parameter sinyal sehingga didapatkan matriks beda antarsinyal.

3.3 Program Dinamik



Gambar 6 Diagram alir program dinamik

Program dinamik dimulai dengan pembobotan matriks beda yang telah didapatkan. Hasil pembobotan adalah nilai bobot 1, 2, atau 3, sesuai dengan besarnya titik beda dibandingkan dengan titik yang lain. Proses selanjutnya adalah penelusuran jalur. Jalur dihitung berdasarkan hasil pembobotan yang telah didapatkan. Jalur dimulai dari titik $D(I,J)$. Apabila titik tempat kedudukan (i,j) berbobot 1, maka jalur dilanjutkan pada titik dengan koordinat $(i-1,j-1)$, apabila berbobot 2, maka jalur dilanjutkan pada titik dengan koordinat $(i-1,j)$, apabila berbobot 3, maka jalur

dilanjutkan pada titik dengan koordinat $(i,j-1)$. Penelusuran jalur dilakukan sampai pada titik D(0,0). Apabila telah didapatkan jalur, langkah selanjutnya adalah menghitung besar beda rata-rata dari titik-titik yang dilalui oleh jalur. Masukan dari program dinamik ini adalah matriks beda. Berdasarkan matriks beda ini kemudian dicari nilai jalur terendah. Setelah didapatkan matriks pembobotan, langkah selanjutnya membuat jalur yang mempunyai nilai terendah berdasarkan hasil pembobotan yang telah dilakukan. Apabila telah didapatkan jalur terendah kemudian dihitung nilai rata-rata titik yang dilewati.

3.4 Pengambilan Keputusan

Setelah didapatkan nilai jalur terendah dari matriks beda, langkah selanjutnya adalah menentukan apakah sinyal yang dibandingkan cocok dengan sinyal pembanding. Proses pengambilan keputusan ini diawali dari penghitungan persentase keterkenalan. Pada penghitungan persentase keterkenalan ini sinyal yang memiliki pola yang sama (satu sinyal) diberi persentase 100%, sedangkan sinyal yang mempunyai nilai rata-rata jalur yang paling besar di antara semua basisdata, diberi persentase 0%. Untuk menentukan kecocokan maka pada pengambilan keputusan diberi suatu ambang (*threshold*) nilai persentase, yaitu persentase terendah yang dibolehkan agar sinyal dapat dikenali. Batasan ini juga dimaksudkan agar sinyal yang memiliki nilai persentase paling tinggi di antara sinyal lain pada basisdata tetapi tidak melewati batasan tersebut tidak dapat dikenali. Batasan yang diberikan adalah 50,2061 %.

IV. HASIL PENGUJIAN

Pengujian dilakukan dengan menggunakan kombinasi 2 parameter tersebut. Pengujian yang dilakukan diharapkan akan diperoleh kombinasi parameter yang menghasilkan pengenalan yang paling baik. Hasil pengujian ini diharapkan dapat memberikan gambaran dan kesimpulan atas hasil penelitian yang telah dilakukan.

TABEL 1 VARIASI NILAI PARAMETER

Parameter	Variasi
Orde TFPC	5
	10
	15
	20
Panjang <i>Frame</i>	10 ms
	20 ms
	30 ms

4.1 Pengujian Data

Pengujian ini dilakukan dengan menggunakan ciri-ciri sinyal yang telah ditentukan. Pengujian dibagi menjadi 2 yaitu yang pertama adalah menguji program pengenalan dengan sinyal data latih, yaitu sinyal yang sama dengan salah satu sinyal pada basisdata. Kemudian yang kedua adalah menguji program pengenalan dengan

sinyal data baru. Pada masing-masing pengujian dilakukan pembandingan beda kumulatif untuk setiap parameter masukan sehingga didapat pengenalan terbaik.

Pengujian untuk mendapatkan hasil pengenalan terbaik dilakukan dengan menghitung persentase pengenalan. Pada perhitungan ini, sinyal yang memiliki pola yang sama (satu sinyal) diberi persentase 100 %, sedangkan sinyal yang mempunyai nilai jalur rata-rata (beda kumulatif) terbesar dari basisdata diberikan persentase 0 %. Pengujian dengan membandingkan beda kumulatif ini untuk mengetahui seberapa besar kecocokan antara sinyal uji dengan sinyal referensi (pada basisdata). Langkah terakhir adalah membandingkan hasil dari proses pengenalan setelah dipersentasekan dengan mengubah-ubah parameter sehingga diketahui parameter mana yang menghasilkan pengenalan terbaik.

4.2 Pengujian Data Latih

Pada program pengenalan ini telah disediakan masing-masing 5 sinyal basisdata sebagai sinyal referensi itik jantan dan betina. Basisdata itik jantan berisi sinyal JANTAN1, JANTAN2, JANTAN3, JANTAN4, JANTAN5, sedangkan pada basisdata itik betina berisi sinyal BETINA1, BETINA2, BETINA3, BETINA4, dan BETINA5. Pengujian yang dilakukan dengan sinyal yang sama dengan sinyal basisdata akan menghasilkan persentase pengenalan sebesar 100 %. Pengujian ini bertujuan untuk mengetahui apakah program pengenalan telah dapat bekerja dengan benar. Persentase pengenalan yang dihasilkan adalah 100% untuk semua hasil pengujian.

4.3 Pengujian Data Baru

TABEL 2 PARAMETER TERBAIK PENGUJIAN DATA BARU

Data Baru	Orde TFPC	Panjang <i>Frame</i> (ms)	Persentase (%)
JANTAN11	5	20	69,6700
JANTAN12	5	20	74,2845
JANTAN13	20	30	72,1519
JANTAN14	15	20	62,6225
JANTAN15	15	30	67,4561
JANTAN16	10	30	62,1130
JANTAN17	15	30	77,4915
BETINA11	20	20	81,5449
BETINA12	10	30	71,1446
BETINA13	20	30	91,6540
BETINA14	20	20	63,3349
BETINA15	20	30	91,6911
BETINA16	10	20	51,0867
BETINA17	5	30	70,7117
Rata-rata			71,9255

Pada pengujian dilakukan menggunakan sinyal data baru, yaitu sinyal yang tidak terdapat dalam basisdata. Sinyal uji yang digunakan dalam pengujian ini adalah sinyal JANTAN11, JANTAN12, JANTAN13, JANTAN14, JANTAN15, JANTAN16, JANTAN17, BETINA11, BETINA12, BETINA13, BETINA14,

BETINA15, BETINA16, dan BETINA17. Pengujian data baru yang pertama dilakukan tanpa menambah jumlah basisdata, sedangkan pengujian data baru yang kedua dilakukan dengan menambah jumlah basisdata. Pengujian ini dilakukan dengan mengubah-ubah parameter yang digunakan untuk satu sinyal. Dengan pengujian ini, akan dicari kombinasi parameter terbaik pada proses pengenalan.

Karena sinyal masukan merupakan sinyal baru, maka untuk meningkatkan kinerja program dilakukan penambahan basisdata. Pada pengujian ini, diberikan masing-masing 10 basisdata itik jantan dan 10 basisdata itik betina. Pada basisdata itik jantan, berisi sinyal JANTAN1, JANTAN2, JANTAN3, JANTAN4, JANTAN5, JANTAN6, JANTAN7, JANTAN8, JANTAN9, dan JANTAN10. Sedangkan pada basisdata itik betina berisi sinyal BETINA1, BETINA2, BETINA3, BETINA4, BETINA5, BETINA6, BETINA7, BETINA8, BETINA9, dan BETINA10.

TABEL 3 PARAMETER TERBAIK PENGUJIAN DATA BARU (PENAMBAHAN BASISDATA)

Data Baru	Orde TFPC	Panjang Frame (ms)	Persentase (%)
JANTAN11	5	20	69,6700
JANTAN12	5	20	74,2845
JANTAN13	20	30	72,1519
JANTAN14	5	30	76,7830
JANTAN15	15	30	67,6753
JANTAN16	5	30	74,7793
JANTAN17	5	30	84,9992
BETINA11	20	30	86,8081
BETINA12	10	30	75,9906
BETINA13	20	30	91,6540
BETINA14	5	30	76,6819
BETINA15	20	30	91,6911
BETINA16	10	20	53,1408
BETINA17	20	30	67,7173
Rata-rata			76,0019

Rata-rata persentase pengenalan data baru meningkat setelah dilakukan penambahan basisdata yang ditunjukkan pada Tabel 2 dan Tabel 3. Rata-rata persentase pengenalan data baru sebelum penambahan basisdata adalah 71,9255 %, sedangkan rata-rata persentase pengenalan data baru setelah penambahan basisdata adalah 76,0019 %. Panjang sinyal ucapan yang digunakan pada proses pengenalan juga mempengaruhi hasil pengenalan. Semakin panjang sinyal ucapan maka ciri-ciri akustik untuk setiap responden akan lebih banyak disarikan sehingga hasil pengenalan akan semakin bagus. Namun sebagai konsekuensinya, proses pengenalan akan semakin lama. Normalisasi sinyal ucapan yang sempurna juga akan meningkatkan hasil pengenalan, karena sinyal ucapan yang murni bisa didapatkan.

Berdasarkan hasil analisa data pengujian, maka dalam pengenalan pengucap menggunakan parameter TFPC ini perlu diperhatikan hal-hal sebagai berikut.

1. Panjang *frame*

Panjang *frame* ditentukan dengan membagi-bagi sinyal dalam segmen-segmen pendek. Semakin pendek *frame* (dalam ms), maka jumlah *frame* untuk setiap sinyal juga semakin banyak. Dengan jumlah *frame* yang semakin banyak, maka jumlah sampel yang terdapat dalam suatu *frame* akan semakin sedikit.

2. Orde TFPC

Penghitungan koefisien menghasilkan suatu matriks yang mempunyai jumlah baris sesuai dengan jumlah *frame* dan jumlah kolom sesuai dengan (orde TFPC). Semakin besar pengesetan orde parameter, maka matriks koefisien yang dihasilkan juga akan semakin besar. Hal ini akan mempengaruhi penggunaan memori dan kecepatan sistem dalam proses ekstraksi koefisien.

3. Kualitas data sinyal ucapan hasil perekaman

Karena pada pembuatan tugas akhir ini proses perekaman dilakukan pada ruangan biasa (tidak kedap suara), maka kualitas sinyal yang dihasilkan juga kurang sempurna. Hal ini ditandai dengan besarnya derau yang ikut terekam dalam proses perekaman.

4. Kondisi suara responden

Kondisi suara responden sangat mempengaruhi hasil pengenalan. Setiap pengucapan tidak bisa sama persis 100 %. Untuk mendapatkan hasil pengenalan yang baik, maka suara responden diharapkan sesuai dengan kondisi normalnya.

5. Pemilihan basisdata

Pemilihan basisdata sangat berpengaruh pada akurasi pengenalan. Semakin banyak basisdata yang diberikan, maka peluang keterkenalan suatu sinyal uji akan semakin besar. Basisdata yang sebaiknya dipilih adalah hasil perekaman yang tidak dilakukan pada satu waktu, sehingga ragam cara pengucapan yang dimiliki responden akan terwakili semua. Semakin beragam basisdata yang diberikan, maka peluang keterkenalan sinyal uji juga akan semakin baik.

V. KESIMPULAN

Dari pengujian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut.

1. Persentase pengenalan data latih itik jantan maupun betina adalah 100 % untuk seluruh parameter.
2. Hasil pengenalan terbaik dapat diperoleh jika proses perekaman sinyal basisdata maupun sinyal uji terbebas dari derau.
3. Rata-rata persentase pengenalan data baru meningkat setelah dilakukan penambahan basisdata.
4. Rata-rata persentase pengenalan data baru sebelum penambahan basisdata adalah 71,9255 %, sedangkan rata-rata persentase pengenalan data baru setelah penambahan basisdata adalah 76,0019 %.
5. Pengujian data baru itik jantan sebelum penambahan basisdata diperoleh parameter terbaik rata-rata pada orde TFPC 15 dan panjang *frame* 30 ms.
6. Pengujian data baru itik betina sebelum penambahan basisdata diperoleh parameter terbaik rata-rata pada orde TFPC 20 dan panjang *frame* 30 ms.
7. Pengujian data baru itik jantan setelah penambahan basisdata diperoleh parameter terbaik rata-rata pada orde TFPC 5 dan panjang *frame* 30 ms.

8. Pengujian data baru itik betina setelah penambahan basisdata diperoleh parameter terbaik rata-rata pada orde TFPC 20 dan panjang *frame* 30 ms.

DAFTAR PUSTAKA

- [1] Aminudin, A., *Analisis Eigen Sinyal Suara*, Skripsi S-1, Universitas Diponegoro, Semarang, 2003.
- [2] Dwiharjanto, I.Y., *Pengenalan Pengucap Bergantung Teks Dengan Menggunakan Metode Time-Frequency Principal Component (TFPC)*, Skripsi S-1, Universitas Diponegoro, Tugas Akhir, Semarang, 2003.
- [3] Hanselman, D., and B. Littlefield, *MATLAB : Bahasa Komputasi Teknis*, Penerbit ANDI, Yogyakarta, 2000.
- [4] Kurnia, A., *Penala Nada Menggunakan Alihragam Fourier*, Skripsi S-1, Universitas Diponegoro, Semarang, 2004.
- [5] Kurnianto, N.A., *Penentuan Jenis Kelamin Itik Dengan Metode Dynamic Time warping (DTW)*, Skripsi S-1, Universitas Diponegoro, Semarang, 2005.
- [6] Magrin-Chagnolleau, I. and G. Durou, *Time-Frequency Principal Components of Speech : Application to Speaker Identification*, <http://citeseer.ist.psu.edu/>, 2000.
- [7] Qian, S., and D. Chen, *Short-Time Fourier Transform Joint Time-Frequency Analysis*, <http://zone.ni.com/devzone/conceptd.nst/webmain/>, 1996.
- [8] Rabiner, R.L., and B.H. Juang, *Fundamentals of Speech Recognition*, PTR Prentice-Hall Inc., New Jersey, 1993.
- [9] Selesnick, I., *STFT*, <http://cnx.org/content/m10570/latest>, August 2005.
- [10] Smith, L.I., *A Tutorial on Principal Component Analysis*, http://csnet.atago.ac.nz/coscs453/student_tutorials/, February 2002.
- [11] Sodikin, A., *Wisata Itik Alabio di Kalsel*, <http://www.kompas.com/kompas-cetak/0409/18/Wisata/125216.htm>, September 2004.



**Stefan Eka Setiawan
(L2F001642)**

Saat ini sedang menyelesaikan Tugas Akhir pada program S1 jurusan Teknik Elektro konsentrasi Elektronika Telekomunikasi Universitas Diponegoro.

Menyetujui dan mengesahkan

Pembimbing I

Pembimbing II

Achmad Hidayatno, S.T., M.T.
NIP. 132 288 515

R. Rizal Isnanto, S.T., M.M., M.T.
NIP. 132 137 933