

# Identifikasi Suara Pengontrol Lampu Menggunakan Mel-Frequency Cepstral Coefficients dan Hidden Markov Model

Angga Kersana Munggaran<sup>1</sup>, Esmeralda C Djamal, Rezki Yuniarti  
Jurusan Informatika, Fakultas MIPA  
Universitas Jenderal Achmad Yani  
Cimahi, Indonesia  
<sup>1</sup>[anggakersana@gmail.com](mailto:anggakersana@gmail.com)

**Abstract**—Identification of voice signals can be used to command a computer system. Identification can be made of voice owner and spoken word. Identification of voice owner is used for security, while spoken word identification is often used to execute command on external devices. Previous research on voice identification using Mel-Frequency Cepstral Coefficient and Linear Vector Quantisation, other research using Linear Prediction Cepstral Coefficients and Linear Vector Quantisation. This study create a system that can identify the speakers on 15 seconds. Voice signal extracted using Mel Frequency Cepstral Coefficient, then Hidden Markov Model are used for classification. The classification results are tested to execute the command to turn on lamp.

**Keywords**— *Identification of the speaker; mel frequency cepstral coefficient; voice signal; hidden markov model;*

## I. PENDAHULUAN

Komunikasi antara manusia lumrahnya dilakukan menggunakan suara, melalui suara manusia dapat mengirimkan perintah atau informasi kepada manusia lain. Pada perkembangan saat ini pengolahan suara digital sudah banyak diimplementasikan pada perangkat-perangkat terutama perangkat komputer atau yang berhubungan dengan mikrokontroler. Dimana unit logika dapat mengolah sinyal suara analog menjadi sinyal suara digital. Sinyal suara digital tersebut dapat diolah menjadi sebuah informasi yang dikirim kepada manusia lain. Manusia sendiri dapat menerima setiap persepsi dari lawan bicaranya dan mengerjakan apa yang terkandung dalam informasi tersebut. Berbeda halnya dari komputer yang memerlukan perbandingan data yang akan diolah untuk mengelola perintah tersebut. Oleh karena itu, apabila sebuah lampu dalam suatu rumah dikendalikan tanpa harus menyalakan saklar didalam rumah maka peran mikrokontroler, komputer/laptop, serta fasilitas internet sangat penting untuk memberi kemudahan khususnya penyandang cacat fisik atau orang yang sudah tua. Maka dari itu dibutuhkanlah bentuk lain dari pengendalian lampu rumah atau perangkat elektronik lainnya dengan mengenali pemilik atau yang berwenang, salah satunya adalah dalam bentuk suara.

Manusia dapat mengenali seseorang hanya dengan mendengarkan orang tersebut berbicara. Maka, beberapa detik ucapan sudah cukup untuk mengidentifikasi pembicara. Pengenalan pembicara adalah proses untuk mengidentifikasi pembicara secara otomatis berdasarkan karakteristik suara [1].

Melalui konsep ini, pengenalan pembicara memungkinkan untuk menggunakan karakteristik suara pembicara sebagai kata sandi untuk memverifikasi identitas pembicara dan mengakses layanan tersebut.

Ada dua jenis pengenalan pembicara, yaitu [2]: pengenalan pembicara tidak bergantung pada teks dan pengenalan pembicara bergantung pada teks. Pengenalan pembicara tidak bergantung pada teks adalah jenis proses identifikasi suara pembicara tanpa membatasi pengenalan dengan kata-kata tertentu. Pengenalan pembicara tidak bergantung pada teks membutuhkan pelatihan data yang banyak untuk mendapatkan akurasi yang baik, sedangkan Pengenalan pembicara bergantung pada teks adalah jenis proses identifikasi suara pembicara dengan menggunakan kata-kata yang sama, tipe ini lebih cocok untuk digunakan untuk teknik identifikasi suara pembicara karena untuk proses identifikasi pembicara bisa didapatkan hanya dengan mengekstraksi ciri suara melalui beberapa kata sebagai sample.

Penelitian terdahulu melakukan identifikasi dan klasifikasi sinyal pembicara yaitu, studi komparasi untuk pengenalan bahasa dengan LPCC dan MFCC [15], adapun identifikasi menggunakan GFCC dan klasifikasi Gaussian [4], penelitian terdahulu juga telah meneliti tentang kekasaran suara antara GFCC dan MFCC [5], mengidentifikasi pembicara menggunakan SVM [6], mengidentifikasi pembicara dengan klasifikasi LVQ [7], menggunakan metode penggabungan Jaringan Syaraf Tiruan dan HMM [8], mengidentifikasi ucapan dengan Mel dan klasifikasi LVQ [9], identifikasi pembicara dengan ekstraksi MFCC dan klasifikasi menggunakan Jaringan Saraf Tiruan Backpropagation [10].

Metode-metode yang dapat digunakan untuk mengekstraksi fitur suara adalah LPC, LPCC (Linear Predictive Cepstral Coefficients), dan MFCC. Berdasarkan penelitian yang dilakukan, metode MFCC yang digunakan untuk pengenalan suara dengan kondisi noise sebesar 20dB memberikan akurasi sebesar 97.03% sedangkan metode LPCC hanya memberikan akurasi sebesar 73.76% [1].

Pada penelitian ini, metode yang digunakan untuk ekstraksi fitur suara adalah Mel-frequency cepstral coefficients. Metode ini merupakan metode yang paling sering digunakan dalam melakukan ekstraksi fitur suara dalam penelitian pengenalan identitas pembicara dan pengenalan perintah suara, karena mampu untuk mengekstraksi karakteristik sinyal suara secara

kelas, yang relatif berbeda dari setiap sifat saluran suara pembicara dan lebih efektif digunakan untuk pengenalan suara yang mengandung noise. Metode HMM digunakan sebagai pengenalan pola untuk mengidentifikasi pembicara sesuai dengan fitur suara yang telah diekstraksi menggunakan metode MFCC. Pada penelitian ini data diambil dari 16 naracoba dengan mengucapkan dua perintah yaitu “nyalakan” dan “matikan” yang digunakan untuk mengklasifikasi kelas karakter dan perintah yang diucapkan. Keluaran tersebut akan digunakan untuk mengirimkan data yang akan mengontrol lampu melalui koneksi internet sesuai dari perintah yang diucapkan.

## II. METODE

### A. Akuisisi Data

Input File Suara yang digunakan pada perolahan data suara berupa file suara yang didapat dari rekaman secara langsung. Pengambilan data dengan cara merekam sinyal suara dari 16 naracoba dimana dibagi menjadi dua yaitu delapan pria dan delapan wanita berumur 4-65 tahun dan dengan kondisi sehat. Setiap file suara berdurasi satu-tiga detik tergantung ucapan yang diberikan sebagai input dan diucapkan masing-masing 10 kali dan setiap perulangan terdapat dua sesi yaitu perulangan kata nyalakan dan perulangan kata matikan. Terdapat 320 data latih pertama yang digunakan pada sistem, 320 data tersebut dihasilkan dari hasil perekaman 16 naracoba ((16 naracoba x 2 sesi) x 10 perulangan = 320). File suara yang digunakan pada penelitian ini memiliki frekuensi sampling sebesar 8000 Hz karena pada umumnya suara percakapan manusia adalah 300 Hz – 8000 Hz. Untuk memenuhi Nyquist Shannon Criterion [1], maka frekuensi sampling yang dipakai adalah 8000 Hz, yang dimana minimum frekuensi sampling adalah dua kali frekuensi sinyal.

### B. Perancangan Sistem Identifikasi

Input pada penelitian ini yaitu data rekaman suara dari 16 orang naracoba melalui mikrofon dengan frekuensi sampling 8000Hz dan resolusi 32bit serta menggunakan *channel* mono dengan format *.wav*. Praproses digunakan untuk setiap input data yaitu *noise reduction* untuk menghilangkan noise dari hasil perekaman audio, kemudian *silence removal* dilakukan untuk menghilangkan sinyal yang bernilai nol hasil dari *noise reduction*.

Setelah dilakukan praproses kemudian dilakukan proses ekstraksi fitur sinyal suara. Ekstraksi fitur merupakan proses perhitungan dari sinyal suara untuk merepresentasikan karakteristik dari sinyal tersebut dalam bentuk data diskrit. Salah satu metode yang digunakan untuk mengekstraksi fitur adalah MFCC. MFCC merupakan metode untuk mengekstraksi fitur yang menghitung koefisien cepstral berdasarkan variasi frekuensi kritis pada sistem pendengaran manusia karena mampu memperhitungkan sifat non linier nada. Terdapat lima langkah proses yang dibutuhkan untuk proses MFCC yaitu frame blocking, windowing, fft, mel-frequency wrapping, dan discrete cosine transform. Proses pertama yang dilakukan adalah frame blocking dimana proses ini berguna untuk membagi sinyal suara kedalam bentuk *short segment*, hal ini perlu dilakukan dikarenakan sinyal suara mengalami perubahan pada jangka waktu tertentu. Pengambilan data pada setiap

perekaman dilakukan setiap rentang waktu 30ms, sehingga untuk satu perekaman menghasilkan 61frame/detik. Windowing berfungsi untuk mengurangi diskontinuitas sinyal dari hasil frame blocking, metode yang digunakan pada proses windowing adalah hamming window, hal ini dikarenakan windowing menghasilkan noise yang tidak terlalu tinggi. FFT dilakukan untuk mengubah sinyal dari domain waktu ke domain frekuensi, hal ini diperlukan karena proses selanjutnya yang memproses sinyal dalam domain frekuensi. Mel-frequency wrapping dilakukan melalui mel-filterbank yang memiliki sinyal dalam bentuk *triangular-window*, jumlah filter yang digunakan sebanyak 32 filter. Sehingga terdapat 34 titik untuk setiap *frame*. DCT dilakukan untuk mendapatkan nilai konversi dari frekuensi mel. Koefisien DCT adalah nilai amplitudo dari spektrum yang dihasilkan. Jumlah cepstrum yang diambil sebanyak 13 fitur pada satu frame dimulai dari indeks ke-1.

Proses pelatihan dan pengujian dilakukan menggunakan HMM. Sebelum dilakukan pelatihan dan pengujian fitur vektor dari MFCC harus dilakukan kuantisasi vektor untuk merepresentasikan nilai vektor menjadi simbol yang akan menjadi input pada HMM sebagai variabel observasi menggunakan metode clustering dan KNN. Setiap file suara akan diambil fitur nya sebagai titik pusat untuk merepresentasikan kelas suara tersebut lalu dilakukan pencarian eucliden distance. Titik pusat yang diambil sebanyak 32 kelas karena setiap orang mengucapkan 2 kelas perintah yaitu “nyalakan” dan “matikan” Hasil tersebut akan diambil labelnya yang akan menjadi input observasi pada HMM.

### C. Mel-Frequency Cepstrum Coefficients

Terdapat lima proses dalam proses ekstraksi menggunakan MFCC, yaitu frame blocking, windowing, fft, mel-wrapping, dan DCT. Frame blocking merupakan proses yang harus dilalui untuk mendapatkan koefisien mel pada tahap Frame Blocking, sinyal suara dibagi menjadi beberapa *frame* dengan panjang pada umumnya sebesar 20-30ms yang berisi N sampel masing-masing frame dipisahkan oleh M (M<N) dimana M adalah banyaknya pergeseran antar *frame*.

Konsep dari Windowing adalah meruncingkan ujung sinyal menjadi nol pada bagian awal dan akhir setiap *frame*. Proses Windowing dilakukan dengan cara mengalikan tiap *frame* dari dengan jenis window yang digunakan, jenis windowing yang digunakan adalah *hamming window* dapat dituliskan dalam persamaan 1 berikut:

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \leq n \leq N-1 \quad (1)$$

Algoritma Fast Fourier Transform adalah algoritma untuk menghitung nilai Discrete Fourier Transform (DFT). DFT adalah metode untuk melakukan transformasi pada domain waktu menjadi domain frekuensi. Penelitian terdahulu yang pernah menggunakan FFT untuk tuning gitar non elektrik, berdasarkan pengujian tingkat akurasi 99.43% [11]. Salah satu algoritma dari FFT yang paling populer adalah radix-2. Sebagai pembandingan dengan DFT, untuk jumlah sampel N yang besar seperti contoh N = 512, dengan menggunakan perhitungan DFT membutuhkan perhitungan sebesar 114 kali lebih banyak daripada yang dibutuhkan oleh perhitungan FFT. Semakin

besar jumlah sampel N, maka perhitungan semakin kompleks jika menggunakan DFT dapat dilihat pada persamaan 2 berikut:

$$f(n) = \sum_{k=0}^{N-1} y_k e^{-2\pi jkn/2}, n = 0,1,2, \dots, n-1 \quad (2)$$

Mel Frequency Wrapping memerlukan filterbank untuk menyaring suara yang telah diubah ke dalam bentuk frekuensi. Filterbank yang digunakan oleh Mel Frequency Wrapping adalah Mel-Filterbank yang terdiri dari rangkaian triangular window yang saling overlap akan menyaring N sampel [12]. Untuk mencari nilai mel dapat dihitung dengan persamaan 1 berikut:

$$mel(f) = 1127 * \ln \left( 1 + \left( \frac{f}{700} \right) \right) \quad (4)$$

Pada tahap terakhir ini, nilai mel akan dikonversikan kembali menjadi domain waktu, yang dimana hasilnya disebut MFCC. DCT merupakan teknik konversi yang dilakukan. Koefisien DCT adalah nilai amplitudo dari spektrum yang dihasilkan. Perhitungan DCT dapat dilihat pada persamaan 5 berikut:

$$c_i = \sum_{m=0}^{M-1} S[m] \cos \left( \frac{\pi n (m-0.5)}{M} \right) \quad 0 \leq n < M \quad (5)$$

#### D. Hidden Markov Model

Hidden markov model (HMM) merupakan pendekatan yang dapat mengelompokkan sifat-sifat spectral dari tiap bagian suara pada beberapa pola. Teori dasar dari HMM adalah dengan mengelompokkan sinyal suara sebagai proses parametric acak, dan parameter proses tersebut dapat dikenali (diperkirakan) dalam akurasi yang tepat.

HMM memiliki lima komponen yaitu:

- Jumlah State (N)

State merupakan parameter tersembunyi (hidden state). Pada penerapannya, jumlah state ini menjadi salah satu parameter uji. Jadi, jumlah state diset sedemikian rupa hingga didapatkan keluaran yang optimal, maka state dilabelkan  $\{1,2,\dots,N\}$  dan state pada waktu t dinotasikan sebagai  $Q_t$ .

- Parameter model (M)

Parameter model merupakan parameter observasi, atau keadaan yang dapat di observasi. Parameter model ini direpresentasikan oleh vektor ciri sinyal suara.

$$O = \{O_{t_1}, O_{t_2}, O_{t_3} \dots, O_{t_n}\} \quad (2.1)$$

- Initial state atau state awal ( $\pi = \pi_i$ )

$$\pi_i = P(q = i) \quad 1 \leq i \leq N \quad (2.2)$$

- Probabilitas transisi ( $A=[A_{ij}]$ ), yaitu probabilitas dari perpindahan dari state i ke state j, dimana transisi antar state-nya dilakukan berdasarkan masukan observasi.

$$A_{ij} = P(q_{i-1} = j | q_j = j) \quad ,1 \leq i, j \leq N \quad (2.3)$$

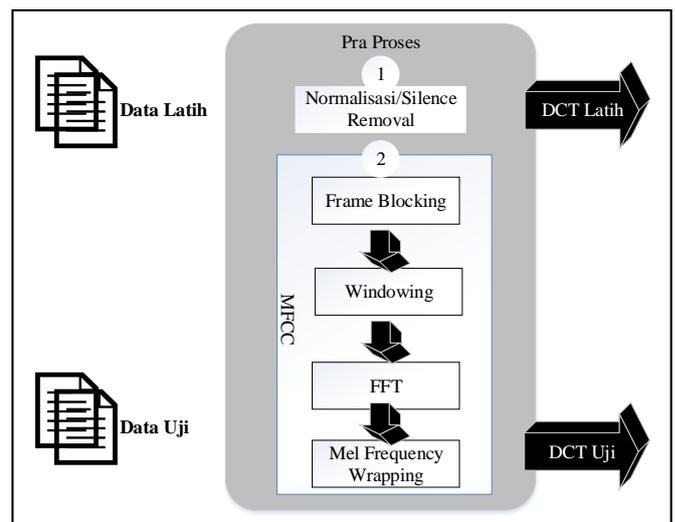
- Probabilitas simbol observasi ( $B=\{b_i(k)\}$ ), yaitu probabilitas observasi yang dibangkitkan state.

$$b_j(k) = P(O_t = V_k | Q_t = j), 1 \leq j \leq N, 1 \leq k \leq M \quad (2.4)$$

Berdasarkan kelima komponen di atas, untuk merancang HMM, dibutuhkan dua parameter model, yaitu N dan M. Selain itu, dibutuhkan tiga nilai probabilitas ( $\pi, A, B$ ) yang dimodelkan dengan menggunakan notasi  $\lambda [\lambda = (A, B, \pi)]$ . Urutan hidden state optimal adalah urutan yang paling tepat yang menjelaskan kejadian yang dapat diamati. Masalah ini disebut juga masalah evaluasi, decoding, dan learning

#### E. Sistem Pra-Proses Fitur Suara

Sistem kontrol lampu jarak jauh dengan menggunakan karakteristik sinyal suara seseorang untuk memberikan perintah yang dimulai dengan pengambilan data suara yang dilakukan secara offline. Data suara diproses melalui tahap pra proses yaitu filter menggunakan MFCC untuk ekstraksi sinyal suara, hasil filter digunakan sebagai data latih atau input deret observasi HMM untuk diidentifikasi karakteristik dari sinyal tersebut yang nantinya digunakan untuk kontrol lampu. Proses ekstraksi fitur pada sistem menggunakan karakteristik sinyal suara dapat dilihat pada Gambar 1.

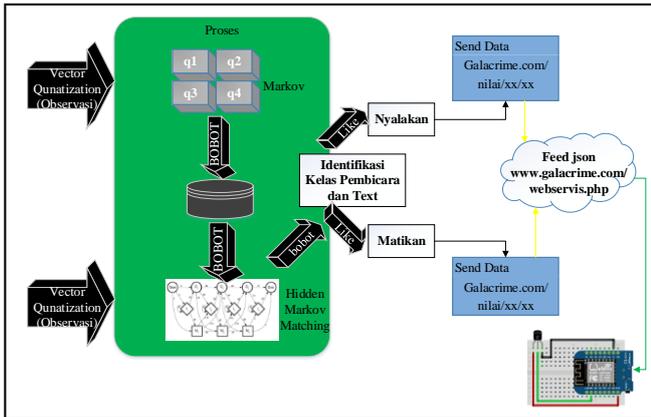


Gambar 1. Proses ekstraksi fitur pada sistem

#### F. Aksi Identifikasi Sistem

Untuk sistem identifikasi karakteristik sinyal suara dengan menggunakan HMM dilakukan terhadap hasil ekstraksi dari filter (DCT) dengan menggunakan MFCC. Proses inialisasi HMM dimulai dengan pengelompokan (*clustering*) menggunakan metode k-means untuk membuat beberapa vektor pusat sebagai wakil dari keseluruhan vektor yang ada [3]. Vektor pusat atau codebook diambil dari sebuah deret vektor pada setiap file, kemudian dilakukan k-means clustering terhadap seluruh deret vektor data latih. Hasil dari klustering tersebut diambil sebagai deret input pada diskrit Hidden Markov Model. Inialisasi parameter initial state, probabilitas transisi, probabilitas emisi. Pelatihan pada Hidden Markov Model dengan mengupdate nilai distribusi transisi dan emisi untuk setiap model yang dibangun. Selanjutnya dilakukan pengujian terhadap setiap model dan hasil log-likelihood terbesar adalah keluaran sebagai kelas teridentifikasi. Setelah teridentifikasi maka sistem akan mengirimkan data berupa nilai di galacrine.com kemudian data tersebut akan diterima oleh

mikrokontroler melalui *feeds json* yang diberikan. Tahapan sistem identifikasi dapat dilihat pada Gambar 2.



Gambar 2. Proses Identifikasi Sistem

Kelas hasil perhitungan HMM dibagi menjadi 16 kelas pada pembicara sesuai dengan suara yang terdaftar, setelah pembicara diketahui maka akan dicari lagi kelas kata yang dibagi menjadi 2 kelas yaitu kata matikan dan nyalakan.

### III. HASIL DAN DISKUSI

#### A. Pengujian

Komponen pengujian :

- 9 Naracoba, dengan mengucapkan 2 kalimat perintah nyalakan dan matikan.
- Total ucapan ((9 naracoba x 2 sesi) x 10 perulangan = 188)
- Jumlah frame sebanyak 512 dan jumlah state =5

TABLE I. KLASIFIKASI PENGUJIAN

No	Epoch	Selisih log	Smoothi ng	Akurasi Pengujian (%)	
				Data Latih	Data Uji
1	500	0.30000	0.01	66	36
2	1000	0.00010	0.01	89	41
3	1000	0.00001	0.001	97.8	48.34

Hasil pengujian dapat dilihat dari Table I . Table I menunjukkan akurasi tertinggi berada pada selisih log 0.00001 dan smoothing 0.001 dengan akurasi terhadap data latih sebesar 97.8% dan terhadap data uji 48.34%.

Data masukan untuk sistem identifikasi suara untuk menyalakan lampu ini menggunakan sinyal suara yang direkam dengan SoX sebagai perekam berdurasi 1-2 detik dari 9 naracoba dimana naracoba dibagi oleh 5 pria dan 4 wanita dengan rentang umur 4-60 tahun dalam kondisi suara yang sehat. Setiap perekaman naracoba dilakukan 10-15 kali perulangan dan setiap perulangan terdapat dua sesi yaitu perulangan kata nyalakan dan perulangan kata matikan. Terdapat 188 data latih pertama yang digunakan pada sistem, 188 data tersebut dihasilkan dari hasil perekaman 9 naracoba ((9 naracoba x 2 sesi) x 10 perulangan = 188).

TABLE II. HASIL PENGUJIAN TERHADAP PEMBICARA DAN UCAPAN

No	Naracoba	Kelas	Data Uji	Pembicara di kenali	Kata dikenali
1	Naracoba 1	Nyalakan	12	12	12
	Naracoba 1	Matikan	11	11	11
2	Naracoba 2	Nyalakan	11	11	11
	Naracoba 2	Matikan	10	10	10
3	Naracoba 3	Nyalakan	10	10	10
	Naracoba 3	Matikan	10	10	10
4	Naracoba 4	Nyalakan	10	10	10
	Naracoba 4	Matikan	10	10	10
5	Naracoba 5	Nyalakan	10	10	10
	Naracoba 5	Matikan	10	8	7
6	Naracoba 6	Nyalakan	10	10	10
	Naracoba 6	Matikan	10	9	9
7	Naracoba 7	Nyalakan	10	10	10
	Naracoba 7	Matikan	10	10	10
8	Naracoba 8	Nyalakan	14	14	14
	Naracoba 8	Matikan	10	10	10
9	Naracoba 9	Nyalakan	10	10	10
	Naracoba 9	Matikan	10	10	10
<b>Total Dikenali</b>			<b>188</b>	<b>185</b>	<b>184</b>
<b>Akurasi (%)</b>			100	98.89	97.87

Dari Table II dapat dilihat setiap naracoba menyebutkan masing-masing perintah, jumlah data uji untuk kelas naracoba 1 adalah 23 dan kelas perintah nyalakan terdapat 12 dan 11 untuk matikan, jumlah data dapat dilihat pada kolom data uji. Hasil identifikasi untuk masing-masing data uji dapat dilihat dari kolom pembicara dikenali untuk hasil pembicara yang dikenali. Untuk pengucapan kata yang dikenali dapat dilihat dari kolom kata dikenali untuk masing-masing data uji.

Berikut adalah perhitungan indikator dari Table II:

Spesifitas pengujian terhadap kata yang dikenali

$$= \frac{\text{kata yang benar}}{\text{kata yang benar} + \text{kata yang salah}} * 100\%$$

$$\frac{184}{184 + 4} * 100\% = 97,87\%$$

Sensitifitas pengujian terhadap kata yang tidak dikenali

$$= \frac{\text{kata salah}}{\text{kata benar} + \text{kata salah}} * 100\%$$

$$\frac{4}{184 + 4} * 100\% = 2,12\%$$

Spesifitas pengujian terhadap penutur yang dikenali

$$= \frac{\text{Penutur benar}}{\text{Penutur Benar} + \text{Penutur Salah}} * 100\%$$

$$\frac{185}{185 + 3} * 100\% = 98,40\%$$

Sensitifitas pengujian penutur yang tidak dikenali

$$= \frac{\text{penutur salah}}{\text{penutur benar} + \text{penutur salah}} * 100\%$$

$$\frac{3}{185 + 3} * 100\% = 1,595\%$$

Rata-rata Akurasi terhadap penutur dan ucapan

$$= \frac{(\text{spesifisitas} + \text{spesifisitas})}{\text{jumlah spesifisitas}} * 100\%$$

$$\frac{0,97 + 0,984}{2} * 100\% = 97,77\%$$

Pengujian terhadap data latih yang diujikan menggunakan HMM dengan selisih log likelihood sebesar 0.00001 jumlah state 5, maksimum iterasi 1000 dan jumlah sample perframe sebesar 512 untuk identifikasi pembicara menghasilkan spesifitas sebesar 98,40% dengan sensitifitas sebesar 1,595% dan ucapan menunjukkan hasil spesifitas sebesar 97,77% dengan sensitifitas sebesar 2,12%. Akurasi rata-rata dari penutur dan ucapan sebesar 97.89%, Hasil pengujian terhadap 188 data latih dapat dilihat pada Table II.

TABLE III. HASIL PENGUJIAN TERHADAP PEMBICARA

No	Naracoba	Data Uji	Pembicara di kenali
1	Naracoba 1	23	23
2	Naracoba 2	21	21
3	Naracoba 3	20	20
4	Naracoba 4	20	20
5	Naracoba 5	20	18
6	Naracoba 6	20	19
7	Naracoba 7	20	20
8	Naracoba 8	24	24
9	Naracoba 9	20	20
<b>Total Dikenali</b>		<b>188</b>	<b>185</b>
<b>Akurasi(%)</b>		<b>100</b>	<b>98.48</b>

Dari Table III dapat dilihat setiap naracoba menyebutkan masing-masing perintah, yang dikelompokkan berdasarkan penutur jumlah data uji untuk kelas naracoba 1 adalah 23 dapat dilihat pada kolom data uji. Hasil identifikasi untuk masing-masing data uji dapat dilihat dari kolom pembicara dikenali untuk hasil pembicara yang dikenali.

Berikut adalah perhitungan indikator dari Table III:

$$\text{Spesifitas pengujian} = \frac{\text{Positif benar}}{\text{Positif Benar} + \text{Negatif Salah}} * 100\%$$

$$\frac{185}{185 + 3} * 100\% = 98,4\%$$

$$\text{Sensitifitas pengujian} = \frac{\text{Negatif benar}}{\text{Positif benar} + \text{Negatif salah}} * 100\%$$

$$\frac{3}{185 + 3} * 100\% = 1,595\%$$

Sedangkan Pengujian terhadap data latih yang diujikan menggunakan HMM dengan pengelompokan untuk karakter pembicara, dengan parameter selisih log likelihood sebesar 0.00001 dengan jumlah state 5, maksimum iterasi 1000, dan jumlah sampel perframe sebesar 512 untuk identifikasi karakter pembicara menunjukkan hasil spesifitas sebesar 98,4% dan sensitifitas sebesar 1,595%. Akurasi sebesar 98.48% dari hasil pengujian terhadap 188 data latih dapat dilihat pada Table III.

#### IV. KESIMPULAN

Suara dapat diolah menjadi sebuah instruksi pada sistem komputer, yang dipengaruhi oleh karakter suara dan bentuk pengucapan. Hal itu dapat dijadikan variabel untuk membentuk

banyak pola dengan mengekstrak fitur pada suara tersebut menggunakan MFCC kemudian diidentifikasi dengan Hidden Markov Model. Parameter smoothing dan selisih likelihood mempengaruhi akurasi pada HMM, semakin kecil selisih nilai log maka ketelitian semakin tinggi. Hal tersebut telah dibuktikan dalam penelitian ini, dimana identitas suara dan perintah dikenali oleh sistem dengan ukuran sample perframe sebesar 512 dan jumlah state pada HMM 5 state dengan selisih likelihood 0.000001 mencapai akurasi sebesar 97.87% dengan 3 data suara tidak dikenali dan 4 data perintah tidak dikenali. Sedangkan untuk pengenalan pembicara saja mencapai akurasi 98.84%. dengan hasil pengenalan 185 data dan 3 data tidak dikenali, sedangkan dari tabel pengujian pada data uji menunjukkan hasil akurasi yang rendah. Hal ini disebabkan klasifikasi yang digunakan pada saat melakukan kuantisasi vektor sangat mempengaruhi hasil identifikasi dari HMM.

#### DAFTAR PUSTAKA

- [1] X. Zhao dan 2. DeLiang Wang, "ANALYZING NOISE ROBUSTNESS OF MFCC AND GFCC FEATURES IN SPEAKER IDENTIFICATION," *ICASSP 2013 978-1-4799-0356-6/13/\$31.00 ©2013 IEEE*, vol. 6, no. 13, pp. 7204-7208, 2013.
- [2] E. Widiyanto, S. N. Endah, S. Adhy dan Sutikno, "APLIKASI SPEECH TO TEXT BERBAHASA INDONESIA MENGGUNAKAN MEL FREQUENCY CEPSTRAL COEFFICIENTS DAN HIDDEN MARKOV MODEL (HMM)," *Prosiding Seminar Nasional Ilmu Komputer Undip 2014*, pp. 39-44, 2014.
- [3] A. R. Widiarti dan P. N. Wastu, "Javanese Character Recognition Using Hidden Markov Model," *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 3, no. 9, pp. 2201-2204, 2009.
- [4] Z. W. A. Larcher, K. A. Lee, E. S. Chng dan T. Kinnunen, "Vulnerability evaluation of speaker verification under voice conversion spoofing: the effect of text constraints," *IEEE*, vol. 5, no. 1, pp. 26-31, 2014.
- [5] A. Setiawan, A. Hidayatno dan R. R. Isnanto, "Aplikasi Pengenalan Ucapan dengan Ekstraksi Mel-Frequency Cepstrum Coefficients (MFCC) Melalui Jaringan Syaraf Tiruan (JST) Learning Vector Quantization (LVQ) untuk Mengoperasikan Kursor Komputer," *ISSN 1411-0814*, vol. 13, no. 3, pp. 82-6, 2011.
- [6] Y. Pan, P. Shen dan P. Shen, "Speech Emotion Recognition Using Support Vector Machine," *International Journal of Smart Home*, vol. 6, no. 2, pp. 101-108, April 2012.
- [7] OssamaAbdel-Hamid dan HuiJiang, "FAST SPEAKER ADAPTATION OF HYBRIDNN/HMM MODEL FOR SPEECH RECOGNITION BASED ON DISCRIMINATIVE LEARNING OF SPEAKER CODE," *ICASSP*, vol. 6, no. 13, pp. 7942-7947, 2013.
- [8] G. Nijhawan dan D. M. Soni, "Speaker Recognition Using MFCC and Vector Quantisation," *Int. J. on Recent Trends in Engineering and Technology*, vol. 11, no. 1, pp. 212-218, July 2014.
- [9] S. Nakagawa, L. Wang dan S. Ohtsuka, "Speaker Identification based on GFCC using GMM," *International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163*, vol. 1, no. 8, pp. 224-232, 2014.
- [10] P. LI, S. ZHANG, H. FENG dan Y. LI, "Speaker Identification using Spectrogram and Learning Vector Quantization," *Journal of Computational Information Systems*, vol. 11, no. 9, pp. 3087-3095, 2015.
- [11] X. Huang, A. Acero dan H.-W. Hon, *Spoken Language Processing: a guide to theory, algorithm, and system development*, Upper Saddle River, New Jersey: Prentice Hall PTR, 2001.
- [12] M. Gales dan S. Young, "The Application of Hidden Markov Models in Speech Recognition," *DOI: 10.1561/2000000004*, vol. 1, no. 3, p. 195-304, 2007.

- [13] R. Dianputra, D. Puspitaningrum dan Ernawati, "IMPLEMENTASI ALGORITMA FAST FOURIER TRANSFORM UNTUK PENGOLAHAN SINYAL DIGITAL PADA TUNING GITAR DENGAN OPEN STRING," *Jurnal Teknologi Informasi, Volume 10 Nomor 2, Oktober 2014, ISSN 1414-9999*, vol. 10, no. 2, pp. 240-248, 2014.
- [14] K. Dash, D. Padhi, B. Pand dan P. S. Mohanty, "Speaker Identification using Mel Frequency Cepstral Coefficient and BPNN," *International Journal of Advanced Research in Computer Science and Software Engineering ISSN: 2277 128X*, vol. 2, no. 4, pp. 327-332, April 2012.
- [15] Bhattacharjee dan Utpal, "A Comparative Study Of LPCC And MFCC Features For The Recognition Of Assamese Phonemes," *International Journal of Engineering Research & Technology (IJERT)ISSN: 2278-0181*, vol. 2, no. 1, pp. 1-6, 2013.