

Penggunaan Pengenal Pengucap Tidak Berdasarkan Teks (*Speaker Recognition Text-Independent*) Sebagai Otorisasi Pengaksesan Pintu

Ardha Ananda A. (L2F 002 555)
Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro,
Semarang, Indonesia
cutey_bee@yahoo.com

Abstrak - Tindak kriminal merupakan salah satu hal penting yang tidak dapat diabaikan. Tingkat keamanan dapat ditingkatkan dengan memanfaatkan perkembangan teknologi. Salah satu perkembangan teknologi itu adalah pengenal pengucap sebagai otorisasi pengaksesan pintu. Teknik pengenalan ini memungkinkan sinyal suara digunakan untuk mengetahui identitas seseorang. Tujuan dari Tugas Akhir ini untuk membuat sistem sederhana dengan menggunakan pengenal pengucap tidak berdasarkan teks sebagai otorisasi pengaksesan pintu.

Dalam Tugas Akhir ini, proses pengenalan pengucap tidak berdasarkan teks menggunakan metode LPC (Linear Predictive Coding) untuk pengekstraksi ciri, pencocokan template menggunakan metode VQ (Vector Quantization) dan menggunakan persamaan jarak Euclidean untuk menghitung jarak penyimpangan minimum antara pengucap yang tidak diketahui dengan pengucap yang telah ada di dalam basisdata. Jika hasil pengenalan cocok dengan salah satu pengucap yang ada di dalam basisdata maka sistem akan memberikan logika 1 yang membuka kunci pintu. Proses antarmuka sistem yang dibuat menggunakan komunikasi paralel.

Hasil pengujian sistem menunjukkan persentase pengenalan untuk 3 responden sebagai basisdata sebesar 91,67 %, untuk 4 responden sebesar 89,38 %, dan persentase untuk 6 responden sebesar 69,17 %. Penggunaan pengenal pengucap tidak berdasarkan teks sebagai otorisasi pengaksesan pintu dapat diaplikasikan dengan baik jika hanya sedikit orang yang dapat mengakses pintu tersebut.

Kata kunci : pengenal pengucap, LPC (Linear Predictive Coding), VQ (Vector Quantization), otorisasi pintu.

I. Pendahuluan

Banyak dikalangan masyarakat yang memanfaatkan perkembangan teknologi dibidang komunikasi, kesehatan bahkan keamanan. Perkembangan teknologi dalam bidang keamanan dapat meningkatkan tingkat keamanan dan kenyamanan. Tingkat keamanan yang tidak maksimal dapat menimbulkan resiko terjadinya kriminalitas. Pemasangan sistem pengamanan pintu merupakan salah satu cara mencegah kriminalitas. Pengamanan pintu dengan memeriksa otorisasi pengakses pintu merupakan salah satu usaha untuk dapat memperkecil kemungkinan terjadi tindak kejahatan.

Latar Belakang

Perkembangan teknologi pengolahan sinyal suara manusia akhir-akhir ini sedang gencar diteliti dan dikembangkan. Salah satu contoh teknologi pengolahan sinyal suara manusia yang sedang dikembangkan adalah sistem pengenal pengucap. Sistem pengenal pengucap dapat digunakan sebagai salah satu sistem pengamanan pintu karena dapat mengeliminir siapa saja yang dapat mengakses pintu, sehingga tidak diperlukan lagi kunci konvensional yang mudah hilang. Sinyal informasi berupa suara pengucap akan digunakan sebagai otorisasi untuk mengakses pintu.

Pengenal pengucap (*speaker recognition*) adalah suatu proses pengenalan untuk mengetahui siapa yang mengucapkan sinyal informasi tersebut dengan mencocokkan karakteristik ucapan yang ada di dalam basisdata dengan ucapan masukan. Karakteristik ucapan dapat dibedakan melalui ekstraksi dengan suatu teknik pengkodean. Teknik pengkodean yang digunakan dalam pengekstraksian ciri sinyal ucapan adalah LPC (*Linear*

Predictive Coding) dan menggunakan metode VQ (*Vector Quantization*) dalam pencocokan ciri (*feature matching*).

Tujuan

Tujuan dari Tugas Akhir ini adalah untuk membuat aplikasi pengaksesan pintu dengan menggunakan identitas suara pengucap sebagai otorisasi akses pintu.

Batasan Masalah

Untuk memaksimalkan kinerja sistem yang dibuat maka dilakukan hal-hal sebagai berikut:

1. Basisdata yang diambil berupa sinyal suara berbahasa Indonesia yang berdurasi ± 1 detik.
2. Perekaman basisdata mengambil 6 responden, yang terdiri dari 3 pria dewasa dan 3 wanita dewasa yang berusia 22 tahun – 23 tahun.
3. Sistem pengenal pengucap yang digunakan sebagai otorisasi pengakses pintu ini tidak bergantung teks yang diucapkan (*Text-Independent*), sehingga hanya mengenali siapa pengucap yang mengucapkan sinyal informasi tersebut.
4. Masukan yang berupa sinyal ucapan diproses menggunakan pengekstraksian ciri LPC dengan metode autokorelasi dan menggunakan VQ dalam pencocokan ciri sinyal tersebut.
5. Derau (*noise*) yang terekam pada proses perekaman basisdata diabaikan.
6. Identifikasi dilakukan dengan menghitung jarak penyimpangan antara *codebook* pada basisdata dengan koefisien LPC sinyal yang ingin dikenali (sinyal uji).
7. Antarmuka perangkat lunak dengan perangkat keras menggunakan komunikasi paralel.
8. Tampilan program menggunakan bahasa pemrograman Visual Basic 6.0, sedangkan proses

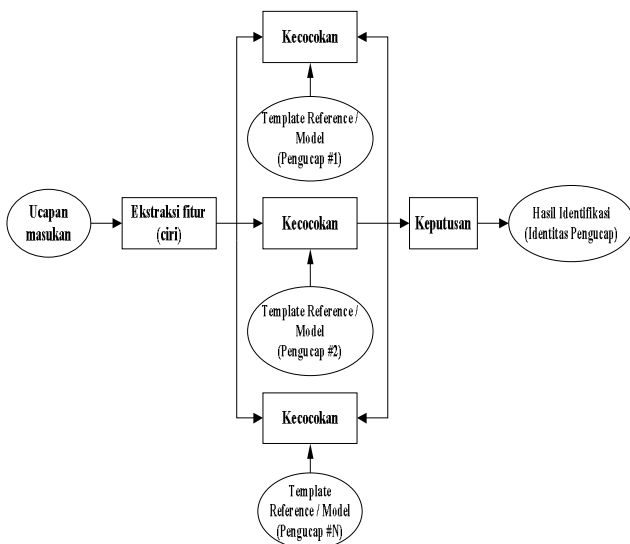
pengenalan dan antarmuka piranti dengan program pengenalan pengucap menggunakan bahasa pemrograman Matlab 6.5.

II. Landasan Teori

Metode pengenalan pengucap dapat dibagi menjadi pengenalan pengucap berdasarkan teks dan pengenalan pengucap tidak berdasarkan teks. Proses pengenalan identitas pengucap berdasarkan pada teks yang diucapkan. Sebaliknya, proses pengenalan pengucap tidak berdasarkan teks adalah proses pengenalan identitas pengucap yang tidak memperhatikan teks yang diucapkan.

Prinsip Dasar Pengenalan Pengucap

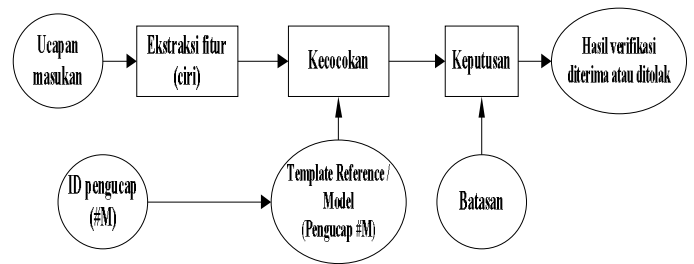
Proses pengenalan pengucap dapat diklasifikasikan menjadi dua proses, yaitu proses identifikasi dan verifikasi. Gambar 1 menunjukkan diagram blok proses identifikasi pengucap yang terjadi pada proses pengenalan.



Gambar 1 Diagram blok proses identifikasi pengucap.

Proses identifikasi merupakan proses mengenali ucapan yang diberikan pengucap yang telah terdaftar dalam basisdata. Proses identifikasi dimulai dengan pemberian suatu sinyal ucapan masukan pada sistem, yang kemudian diekstraksi cirinya. Hasil ekstraksi ciri sinyal ucapan masukan ini kemudian masuk dalam proses pencocokan *template* yang digambarkan dengan bagan 'kecocokan'. Pencocokan dilakukan dengan membandingkan hasil ekstraksi sinyal masukan dengan *template* referensi yang berupa hasil ekstraksi sinyal basisdata yang telah dikelompokkan kedalam kategori-kategorinya (kategori pengucap #1, kategori pengucap #2, dsb). Hasil kecocokan terbaik (jarak penyimpangan paling kecil) akan dipilih untuk kemudian menjadi hasil keputusan.

Setelah proses identifikasi, kemudian dilakukan proses verifikasi. Proses verifikasi merupakan proses diterima atau ditolaknya identitas dari pengucap. Gambar 2 menunjukkan diagram blok proses verifikasi pengucap.



Gambar 2 Diagram blok proses verifikasi pengucap.

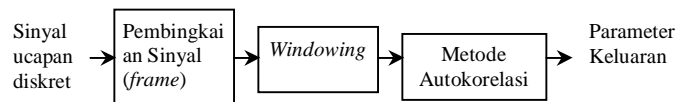
Hasil ekstraksi kemudian dicocokkan dengan *template* referensi dari suatu kategori (misal kategori pengucap #M). Proses selanjutnya pengambilan keputusan, jika hasil pencocokan ada di dalam batasan yang diberikan, maka hasil verifikasi adalah 'diterima' (masukan cocok dengan referensi). Sebaliknya, jika melebihi batasan, maka hasil verifikasi adalah 'ditolak' (masukan tidak cocok dengan referensi).

Analisis LPC

Analisis LPC adalah salah satu cara untuk mendapatkan sebuah pendekatan mengenai spektrum bunyi. LPC secara khusus merupakan metode yang cocok untuk pengolahan sinyal suara (ucapan). Prinsip dasar dari pemodelan sinyal dengan menggunakan LPC adalah bahwa contoh sinyal ucapan $s(n)$ pada waktu ke- n dapat diperkirakan sebagai kombinasi linear dari p sampel sinyal ucapan sebelumnya yaitu :

$$s(n) \approx a_1s(n-1) + a_2s(n-2) + \dots + a_p s(n-p)$$

dengan koefisien a_1, a_2, \dots, a_p diasumsikan bernilai konstan selama *frame* analisis ucapan. Gambar 3 menunjukkan diagram blok analisis LPC.



Gambar 3 Diagram blok analisis LPC.

VQ (Vector Quantization)

Algoritma yang dipakai untuk membentuk *codebook* adalah algoritma LBG (*Linde Buzo Gray Algorithm*). Algoritma LBG mengelompokkan vektor ciri $X = \{x_i | i = 1, 2, \dots, L\}$ menjadi *codebook* $C = (c_1, c_2, \dots, c_C)$, dimana L merupakan jumlah vektor pada vektor ciri dan C merupakan ukuran *codebook* yaitu jumlah vektor yang terbentuk pada sebuah *codebook*. Nilai L harus lebih besar dari nilai C ($L \gg C$).

Algoritma tersebut diimplementasikan dalam prosedur berulang sebagai berikut :

1. Menentukan vektor *codebook* pertama, yang merupakan *centroid* awal dari keseluruhan vektor iri.
2. Menggandakan jumlah *codebook* dengan membagi tiap vektor y_n dari *codebook* yang telah terbentuk, dengan aturan : $y_n^+ = y_n (1 + \epsilon)$
 $y_n^- = y_n (1 - \epsilon)$

dengan n memiliki nilai antara 1 sampai dengan M (ukuran dari *codebook* yang diinginkan), dan ϵ merupakan parameter pembagi ($\epsilon = 0,01$).

3. Pencarian *Nearest-Neighbour* : untuk tiap vektor ciri, temukan *codeword* di dalam *codebook* tersebut (*codebook* saat itu) yang paling dekat (jarak penyimpangannya paling kecil), dan tempatkan vektor tersebut dalam kelompok *codeword* tersebut.
4. Pembaharuan *centroid* : memperbaharui *codeword* pada tiap kelompok dengan menggunakan *centroid* dari vektor ciri terletak di sel tersebut.
5. Iterasi I : mengulang langkah 3 dan 4 sampai diperoleh jarak penyimpangan rata rata (D) yang besarnya dibawah batasan yang telah ditentukan (δ). D' merupakan nilai distorsi awal yang nilainya ditentukan pada saat inialisasi pada awal program.
6. Iterasi II : mengulang langkah 2, 3 dan 4 sampai diperoleh *codebook* dengan ukuran M.

Perhitungan Jarak Minimum

Perhitungan jarak penyimpangan dilakukan dengan membandingkan antara koefisien LPC dari sinyal ucapan yang akan dikenali dan *codebook* dari tiap-tiap pengucap pada basisdata.

$$d(X, C_i) = \frac{1}{L} \sum_{j=1}^L \min_{k=1}^K d_E(x_j, c_{ik})$$

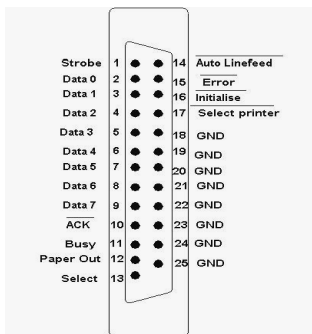
dengan *d* merupakan jarak penyimpangan yang akan diperoleh, *L* merupakan banyaknya *code vector* di dalam setiap *codebook*, *C_i* merupakan jumlah *codebook* (*C_i* = {*c_{i1}*, *c_{i2}*, ... , *c_{iK}*}), *X* adalah vektor ciri hasil ekstraksi LPC sinyal yang akan dikenali (*X* = {*x₁*, *x₂*, ... , *x_L*}) dan *d_E* adalah jarak Euclidean.

$$d_E(x, y) = \sqrt{\sum_{i=1}^{dim} (x_i - y_i)^2}$$

Jarak Euclidean yaitu persamaan yang digunakan untuk mengetahui jarak antara dua vektor.

Konfigurasi Port Paralel

Port paralel dapat digunakan untuk *data in* dan *data out*. Saat *port* paralel mengirimkan data keluaran melalui 8 buah pin-nya, maka pin ini akan mempunyai tegangan 5 volt, tergantung data biner yang dikirim. Gambar 4 menunjukkan konfigurasi *port* paralel DB-25 female.



Gambar 4 Konfigurasi port paralel DB-25 female.

Aktuator

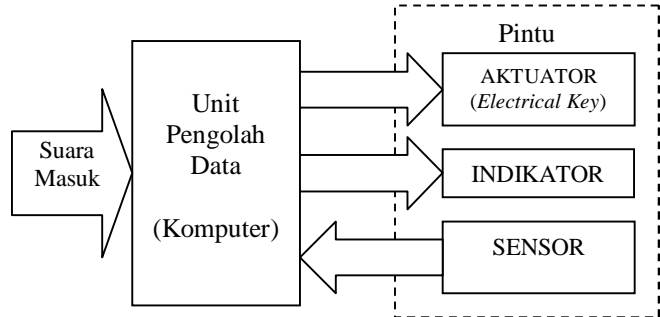
Aktuator adalah bagian dari sistem yang merealisasikan perintah dari unit pengolah yaitu komputer untuk membuka dan mengunci pintu. Aktuator pada sistem pengatur akses pintu berupa kunci elektrik.

Kunci pintu elektrik bekerja seperti sebuah *relay*, yaitu bila ada arus yang melewati kumparan akan menimbulkan medan magnet dan kekuatan medan ini

akan membuat posisi tuas pengunci berubah. Jika sebelum ada arus tuas pengunci dalam posisi mengunci, maka setelah arus mengalir posisi tuas akan berubah terbuka.

III. Perancangan Sistem dan Implementasi Sistem

Sistem ini dibuat untuk pengaturan pengaksesan pintu menggunakan suara sebagai otorisasi. Setiap pengakses pintu yang suaranya telah tersimpan sebagai basisdata akan dapat mengakses pintu tersebut. Gambar 5 menunjukkan diagram blok sistem pengaturan akses pintu menggunakan program pengenalan pengucap tidak berdasarkan teks.

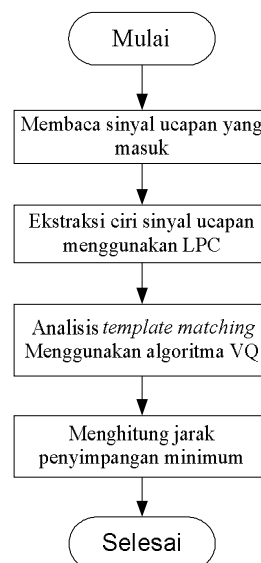


Gambar 5 Diagram blok sistem pengaturan akses pintu menggunakan program pengenalan pengucap tidak berdasarkan teks.

Aktuator merupakan bagian mekanik yang digunakan untuk membuka dan mengunci pintu. Sensor yang terdapat pada pintu berguna untuk mengetahui keadaan pintu. Sinyal ucapan yang masuk akan diproses di unit pengolah data (komputer) menggunakan program pengenalan pengucap. Hasil program pengenalan pengucap akan memberikan logika 1 (*high*) yang akan menggerakkan aktuator.

Pengenalan Pengucap

Proses awal program pengenalan pengucap yaitu membaca sinyal ucapan yang masuk, dan kemudian dibaca untuk mendapatkan bentuk diskret dari sinyal ucapan. Gambar 6 menunjukkan diagram alir proses pengenalan pengucap yang digunakan.



Gambar 6 Diagram alir proses pengenalan pengucap.

Pembacaan sinyal ucapan diikuti dengan normalisasi sinyal ucapan, yang kemudian dianalisis dengan menggunakan analisis LPC. Hasil dari analisis LPC yang berupa koefisien LPC yang akan dibuat *template* dengan menggunakan metode VQ, sehingga diperoleh representasi dari koefisien LPC berupa *codebook* yang terdiri dari beberapa *centroid*. Hasil program pengenalan pengucap didapat dari menghitung jarak penyimpangan minimum dari *codebook* yang sudah dikelompokkan sebagai basisdata.

Perekaman Sinyal Ucapan

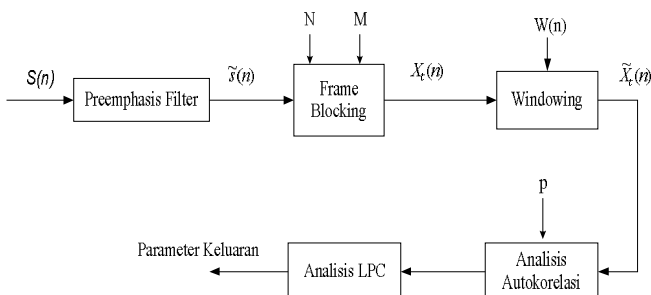
Proses perekaman sinyal ucapan yang digunakan sebagai otorisasi pengaksesan pintu dilakukan secara *on-line*. Sinyal ucapan yang dilakukan pada proses perekaman akan digunakan sebagai pembanding di dalam proses pengenalan pengucap tersebut. Dalam perekaman sinyal ucapan untuk pembuatan basisdata, digunakan program bantu Cool Edit Pro 2. Setiap basisdata yang diambil memiliki frekuensi cuplik 8000 Hz, 16 bit, mono. Basisdata yang digunakan diperoleh dari 6 orang responden pengakses pintu yang masing-masing 3 orang berjenis kelamin pria dan 3 orang yang berjenis kelamin wanita dengan rentang usia 22 – 23 tahun. Ucapan masukan yang digunakan untuk proses pelatihan berupa beberapa kata yang direkam selama ± 1 detik. Masing-masing pengakses mengucapkan 5 teks yang sama.

Ekstraksi Ciri dengan Metode LPC

Proses pengekstraksian ciri merupakan proses untuk mendapatkan parameter-parameter sinyal suara. Parameter-parameter inilah yang nantinya digunakan untuk membedakan satu kata dengan kata yang lain. Proses mencari nilai koefisien LPC dapat dilakukan dengan menggunakan fungsi *lpctakuok*. Pemakaian fungsi *lpctakuok* adalah sebagai berikut;

$$[v, err] = lpctakuok(s, p, t);$$

dengan v adalah nilai dari koefisien LPC yang dicari, err adalah nilai kesalahan peramalan, sedang koefisien LPC diambil dari sinyal yang telah dinormalisasi disimbolkan dengan s , p adalah orde dari LPC, yang bernilai 8 dan t adalah panjang segmen *frame*. Parameter keluaran dari ekstraksi ciri diatas merupakan koefisien LPC yang akan dibuat *codebook* berisi karakteristik masing-masing pengakses pintu. Gambar 7 menunjukkan blok diagram pengekstraksian ciri hingga menghasilkan parameter keluaran.



Gambar 7 Diagram blok pengekstraksian ciri.

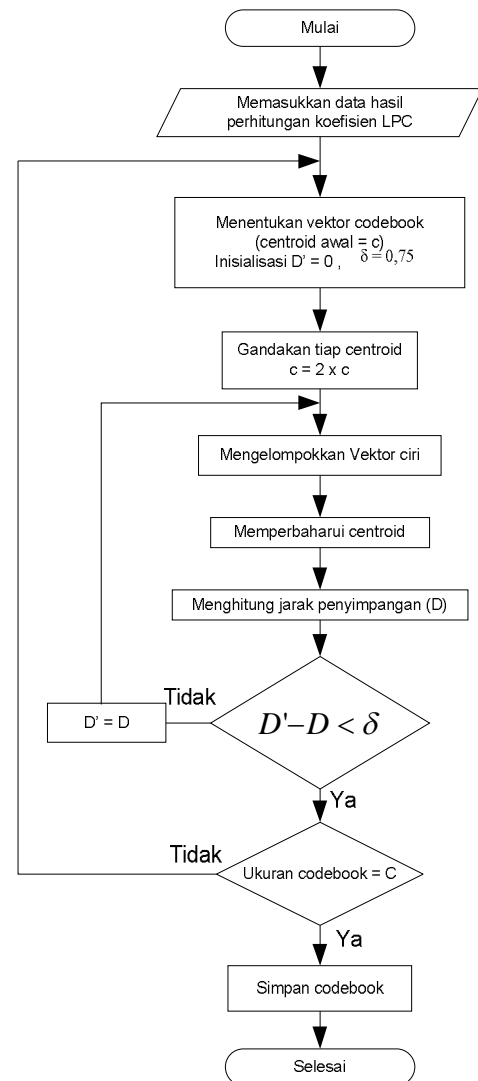
Pembuatan *Template* dengan Metode VQ

Hasil ekstraksi LPC yang berupa koefisien LPC untuk basisdata kemudian dibuat *codebook* dengan menggunakan metode VQ. Dengan menggunakan metode VQ diperoleh kumpulan *codebook* yang isinya merepresentasikan koefisien LPC yang telah diperoleh, yang berisi tentang informasi karakteristik dari masing-masing pengucap.

Sebelum *codebook* diperoleh maka terlebih dahulu harus menentukan ukuran *codebook* yang diinginkan. Untuk mendapatkan nilai *codebook* cukup dengan menggunakan fungsi *vq_lbg*. Pemakaian fungsi *vq_lbg* adalah sebagai berikut :

$$code = vq_lbg(v, c);$$

dengan $code$ adalah nilai dari *codebook* yang dicari, v merupakan koefisien LPC yang sudah didapatkan di dalam proses pengekstraksian ciri dan c merupakan ukuran *codebook* yang ingin diperoleh. Hasil *codebook* dari masing-masing pengucap yang diperoleh kemudian disimpan sebagai basisdata. Gambar 8 menunjukkan diagram alir pembuatan *template* dengan metode VQ.



Gambar 8 Diagram alir pembuatan *template* dengan metode VQ.

Perhitungan Jarak Minimum

Jarak penyimpangan diperoleh setelah menghitung terlebih dahulu besarnya jarak Euclidean dengan menggunakan perintah :

$$d_n = \text{disteu}(v, \text{code}_n);$$

$$d_{\text{tot } n} = \text{sum}(\min(d_n, [], 2)) / \text{size}(d_n, 1);$$

dengan d merupakan nilai jarak Euclidean, d_{tot} adalah jarak penyimpangan, v adalah koefisien LPC dari sinyal uji, code merupakan *codebook* pada basisdata dan n merupakan banyaknya pengucap yang ada pada basisdata.

Nilai jarak penyimpangan yang diperoleh dari perbandingan koefisien LPC sinyal uji terhadap *codebook* dari masing-masing pengucap, diambil yang paling kecil (jarak penyimpangan minimum). Untuk mendapatkan nilai jarak penyimpangan total minimum maka digunakan perintah:

$$d_{\text{min}} = \min(d_{\text{tot}1} \ d_{\text{tot}2} \ d_{\text{tot}3} \ \dots \ d_{\text{tot}n});$$

Dalam proses pengenalan ini nilai jarak penyimpangan minimum yang masih dapat dikenali diberi batasan, sehingga apabila nilai jarak penyimpangan minimum yang diperoleh melebihi batas yang telah diberikan, maka suara yang tidak terdaftar sebagai basisdata tidak akan dikenali. Penentuan nilai ambang sebagai batasan didapat dengan menghitung standar deviasi dari hasil jarak penyimpangan minimum dari proses pengujian.

Proses Antarmuka Sistem

Nilai jarak penyimpangan minimum d yang diperoleh dapat diketahui identitas pengucap yang ingin mengakses pintu. Proses antarmuka yang menggunakan bahasa pemrograman Matlab 6.5, sebuah driver adaptor perangkat keras diinisialisasi terlebih dahulu agar *data acquisition engine* yang terdapat pada Matlab 6.5 dapat mempergunakannya. Untuk mendaftarkan *port* paralel yang sudah tersedia di belakang panel komputer, menggunakan instruksi seperti dibawah ini :

```
out = daqregister('parallel');
dio = digitalio('parallel', 'LPT1');
```

Data Acquisition Toolbox menyediakan akses ke sistem cadangan melalui objek digital I/O. Dengan perintah `digitalio` membuat objek Matlab yang dapat mewakili sistem digital I/O.

```
lines = addline(dio, 0:7, 'out');
```

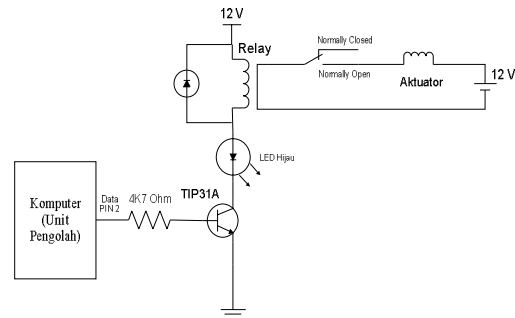
dengan perintah `lines` dapat menentukan pin operasi untuk mengirimkan data keluaran pada data pin 0 sampai data pin 7 pada *port* paralel yang telah terdaftar. Setelah menentukan pin operasi selanjutnya, dengan perintah `putvalue`, memberikan keluaran logika 1 pada data pin 0 untuk membuka pintu.

```
putvalue(dio, [1 0 0 0 0 0 0 0]);
portval = getvalue(dio);
```

Perintah `getvalue` akan membaca masukan dari sensor pintu dan akan memberikan tanda bahwa pintu masih terbuka atau belum tertutup dengan benar.

Rangkaian Sistem

Rangkaian sistem memerlukan komponen semikonduktor berupa transistor yang digunakan sebagai saklar. Saat transistor mencapai keadaan jenuh (saturasi), arus akan mengalir dari basis ke kolektor. Sebaliknya saat transistor mencapai keadaan *cut-off* arus mengalir dari basis ke emiter. Gambar 9 menunjukkan rangkaian skematik dari sistem pengaturan pengaksesan pintu.

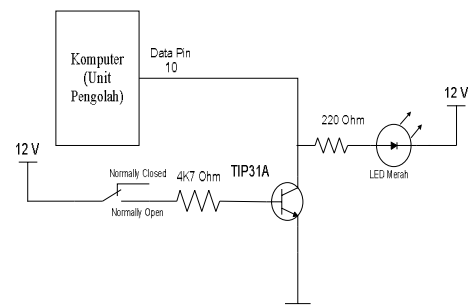


Gambar 9 Rangkaian skematik dari sistem pengaturan pengaksesan pintu.

Unit pengolah memberikan keluaran logika 1 (*high*) melalui data pin 2 yang terdapat pada DB-25. Jika unit pengolah memberikan keluaran logika 1 (*high*), maka *port* paralel memberikan tegangan sebesar 5 volt dan arus mengalir dari basis ke kolektor kemudian LED hijau menyala. LED hijau menyala menandakan pintu terbuka, kemudian arus mengalir melalui relay, menyebabkan relay yang mempunyai kondisi awal *normally open* menjadi *close*, sehingga menggerakkan aktuator dan menyebabkan kunci elektrik terbuka.

Sensor dan Indikator

Sensor pintu bekerja untuk mengetahui apakah pintu masih dalam keadaan terbuka atau tidak. *Limit switch* dirangkaian ini berperan sebagai sensor yang mendeteksi kapan pintu harus mengirimkan masukan berupa logika 1 dan logika 0. Gambar 10 menunjukkan rangkaian skematik dari rangkaian sensor pintu.



Gambar 10 Rangkaian sensor pintu.

Rangkaian sensor kunci pintu bekerja dengan cara memanfaatkan arus yang mengalir untuk memberikan logika 1 saat keadaan *limit switch* tersebut terbuka. Saat pintu tertutup *limit switch*, yang mempunyai kondisi awal *normally open* menjadi *close*, oleh karena itu saat pintu tertutup masukan yang diterima oleh sistem merupakan logika 0. Begitu juga sebaliknya saat pintu terbuka masukan yang diterima oleh sistem merupakan logika 1 dan LED merah sebagai indikator mati.

IV. Pengujian dan Analisis Sistem Langkah Pengujian

Pengujian program ini menggunakan sinyal ucapan sebagai masukan yang direkam secara *on-line*. Dalam pengambilan sampel sinyal ucapan diharapkan dilakukan dalam kondisi normal. Pengujian dilakukan dengan mengucapkan teks bebas yang dilakukan sebanyak 40 kali oleh masing-masing pengucap. Pengujian dilakukan dengan memvariasikan banyaknya jumlah basisdata untuk panjang *frame* yang berbeda-beda. Panjang *frame* yang digunakan 10 ms, 20 ms, dan 30 ms. Kombinasi pengujian dengan 3 pengucap, 4 pengucap dan 6 pengucap yang ada di basisdata menggunakan panjang *frame* yang telah ditentukan. Proses pengenalan ini menggunakan banyaknya basisdata tiap pengucap sebanyak 5 file *.wav.

Proses pelatihan yang dilakukan di dalam program pengenalan pengucap bertujuan untuk mendapatkan *codebook* yang akan digunakan sebagai pembandingan data sinyal ucapan baru (pengenalan *on-line*). Proses pelatihan ini dilakukan diawal program, sehingga untuk melakukan proses pengujian faktor pembandingan yang digunakan tidak berubah. Setiap proses pengujian menghasilkan jarak penyimpangan yang akan digunakan untuk menghitung batasan. Penentuan batasan ini bertujuan untuk menentukan nilai ambang yang digunakan pengucap yang tidak terdapat di dalam basisdata tidak dikenali sehingga tidak dapat mengakses pintu.

Pengujian dan Analisis Sistem

Pengujian dilakukan dengan mengucapkan suara sebagai data uji. Suara masukan diucapkan oleh responden, dengan ketentuan rentang usia 22 – 23 tahun. Untuk mendapatkan data digunakan persamaan sebagai berikut:

$$(\%) \text{Pengenalan} = \frac{\text{Jumlah pengucap yang dapat dikenali}}{\text{Jumlah pengucap}} \times 100\%$$

Hasil pengujian dengan menggunakan panjang *frame* 10 ms, 20 ms dan 30 ms pada 3 responden sebagai basisdata (1 wanita dan 2 pria). Pengujian ini ditambahkan 1 orang pengucap yang tidak terdapat di dalam basisdata. Tabel 1 menunjukkan hasil pengujian dengan 3 responden, panjang *frame* 10 ms.

Tabel 1 Hasil pengujian 3 responden, panjang *frame* 10 ms.

Pengucap	Panjang Frame = 10 ms, Dikenali Sebagai Suara (%)			Tidak Dikenali
	Ardha Ananda	Dewa Komang	Miftahur Rahim	(Pintu tertutup)
Ardha Ananda	90	-	2,5	7,5
Dewa Komang	2,5	87,5	-	10
Miftahur Rahim	-	-	97,5	2,5
Dinar Atika	7,5	-	-	92,5

Persentase pengenalan untuk 3 responden sebagai basisdata dan panjang *frame* 10 ms sebesar 91,67 %. Tabel 2 dan Tabel 3 menunjukkan hasil pengujian dengan panjang *frame* 20 ms dan 30 ms.

Tabel 2 Hasil pengujian 3 responden, panjang *frame* 20 ms.

Pengucap	Panjang Frame = 10 ms, Dikenali Sebagai Suara (%)			Tidak Dikenali
	Ardha Ananda	Dewa Komang	Miftahur Rahim	(Pintu tertutup)
Ardha Ananda	90	-	2,5	7,5
Dewa Komang	2,5	87,5	-	10
Miftahur Rahim	-	-	97,5	2,5
Dinar Atika	7,5	-	-	92,5

Tabel 3 Hasil pengujian 3 responden, panjang *frame* 30 ms.

Pengucap	Panjang Frame = 30 ms, Dikenali Sebagai Suara (%)			Tidak Dikenali
	Ardha Ananda	Dewa Komang	Miftahur Rahim	(Pintu tertutup)
Ardha Ananda	85	-	7,5	7,5
Dewa Komang	5	82,5	-	12,5
Miftahur Rahim	2,5	-	85	7,5
Dinar Atika	12,5	-	5	82,5

Dari data pengujian untuk 3 responden sebagai basisdata diatas, penggunaan panjang *frame* 10 ms mendapatkan hasil pengenalan yang lebih baik dibandingkan dengan penggunaan panjang *frame* 20 ms dan 30 ms. Untuk pengucap yang tidak terdapat di dalam basisdata didapat hasil 92,5 % pada panjang *frame* 10 ms, sehingga hasil perhitungan persentase selanjutnya menggunakan data pengujian dengan panjang *frame* 10 ms.

Hasil pengujian dengan menggunakan panjang *frame* 10 ms pada 4 responden sebagai basisdata yang terdiri 2 wanita dan 2 pria dan 1 pengucap yang tidak terdapat di dalam basisdata ditunjukkan pada Tabel 4.

Tabel 4 Hasil pengujian 4 responden, panjang *frame* 10 ms.

Pengucap	Panjang Frame = 10 ms, Dikenali Sebagai Suara (%)				Tidak Dikenali
	Ardha Ananda	Dewa Komang	Miftahur Rahim	Dinar Atika	(Pintu tertutup)
Ardha Ananda	87,5	-	10	-	2,5
Dewa Komang	2,5	90	-	-	7,5
Miftahur Rahim	-	-	95	-	5
Dinar Atika	5	-	-	85	10
Wastu	12,5	-	-	5	82,5

Persentase pengenalan untuk 4 responden sebagai basisdata dan panjang *frame* 10 ms sebesar 89,38 %. Untuk pengucap yang tidak terdapat di dalam basisdata didapat hasil 82,5 %.

Hasil pengujian dengan menggunakan panjang *frame* 10 ms pada 6 responden sebagai basisdata yang terdiri 3 wanita dan 3 pria dan 1 pengucap yang tidak terdapat di dalam basisdata ditunjukkan pada Tabel 5.

Tabel 5 Hasil pengujian 6 responden, panjang *frame* 10 ms.

Speaker	Panjang Frame = 10 ms, Dikenali Sebagai Suara (%)						Not Known
	Ar dha	De wa	Ra him	Di nar	Bay haki	Ar yati	
Ardha	67,5	-	2,5	-	12,5	10	7,5
Dewa	-	70	-	-	7,5	-	22,5
Rahim	5	-	77,5	-	-	-	12,5
Bayhaki	15	-	2,5	2,5	67,5	-	12,5
Aryati	22,5	-	-	2,5	5	65	2,5
Dinar	2,5	-	-	67,5	12,5	10	7,5
Wastu	10	-	2,5	-	5	12,5	70

Persentase pengenalan untuk 6 responden sebagai basisdata dan panjang *frame* 10 ms sebesar 69,17 %. Untuk pengucap yang tidak terdapat di dalam basisdata didapat hasil 70 %.

Tabel 6 menunjukkan rata-rata persentase keseluruhan dari setiap variasi pengujian.

Tabel 6 Rata-rata persentase hasil pengenalan dari keseluruhan variasi pengujian.

Banyaknya Responden sebagai basisdata	Panjang <i>Frame</i> yang digunakan (ms)			Rata-rata Persentase pengenalan yang digunakan(%)
	10	20	30	
3 responden	91,67	89,17	84,17	91,67
4 responden	89,38	88,13	86,88	89,38
6 responden	69,17	68,75	68,75	69,17

Hasil rata-rata persentase yang digunakan ke dalam sistem pengaksesan pintu adalah pengenalan pengucap dengan panjang *frame* 10 ms. Dari hasil pengujian dapat disimpulkan bahwa jumlah responden pengakses pintu ternyata sangat berpengaruh untuk program pengenalan pengucap tidak bergantung teks. Aplikasi dari penggunaan pengenalan pengucap tidak bergantung teks sebagai otorisasi pengaksesan pintu dapat digunakan pada pintu yang sedikit orang dapat mengaksesnya, karena semakin banyak pengakses sebagai basisdata semakin rendah persentase hasil pengenalanya.

Berdasarkan hasil analisis data pengujian, maka dalam pengenalan pengucap menggunakan analisis LPC ini perlu diperhatikan hal-hal sebagai berikut :

1. Kondisi Lingkungan

Karena kondisi ruangan saat perekaman tidak kedap suara sehingga memungkinkan adanya suara-suara lain yang ikut terekam. Dengan adanya derau maka akan dihasilkan vektor ciri yang tidak mencirikan sinyal asli sehingga menghasilkan nilai jarak penyimpangan minimum pada tempat yang salah. Untuk mengatasi hal ini diperlukan ruangan perekaman yang seminimal mungkin bebas derau.

2. Kondisi suara responden

Kondisi suara responden sangat mempengaruhi hasil pengenalan. Setiap sinyal suara yang diucapkan oleh seseorang selalu memiliki karakteristik yang berbeda, baik itu panjang-pendek, keras-pelan dan lain-lain. Hal ini dapat diatasi dengan mengusahakan

pengucapan sesuai kondisi normalnya.

3. Letak mikrofon.

Karena perekaman dilakukan secara bertahap sehingga dalam peletakan mikrofon tidak sama. Jarak dan sudut mikrofon sangat mempengaruhi pada sinyal suara yang dihasilkan. Untuk mengatasi hal ini dapat dilakukan dengan mengatur jarak mikrofon dan sudut mikrofon yang sebisa mungkin sama.

4. Kondisi mikrofon.

Perekaman sinyal ucapan harus dilakukan saat mikrofon tidak rusak. Keadaan rusak disini dimaksudkan keadaan mikrofon tidak menambahkan derau. Karena saat mikrofon sering jatuh dan kabel yang terkelupas pun akan dapat mempengaruhi proses pengenalan dan pengambilan ciri sinyal ucapan.

5. Cara perekaman sinyal suara

Perekaman sinyal suara yang tepat juga ikut mempengaruhi ketepatan pengenalan. Cara perekaman yang baik adalah ucapan diucapkan sewajar mungkin dan tidak dibuat-buat, suara diucapkan tidak terlalu keras dan tidak terlalu lemah.

6. Parameter yang digunakan.

Parameter yang digunakan antara lain panjang *frame*, orde LPC dan ukuran *codebook*. Panjang *frame* digunakan untuk membagi sinyal menjadi *frame-frame* pendek, semakin pendek ukuran *frame* maka jumlah *frame* yang diperoleh semakin banyak. Orde LPC digunakan untuk menggeser sinyal, semakin besar orde LPC maka pergeseran yang dilakukan akan semakin banyak. Ukuran *codebook* digunakan untuk mendapatkan banyaknya vektor dalam tiap-tiap *codebook* pada *template* yang terbentuk.

7. Penggunaan basisdata

Basisdata sangat berpengaruh pada akurasi pengenalan, semakin banyak basisdata yang digunakan maka ciri karakteristik sinyal suara yang diperoleh juga akan semakin beragam, sehingga peluang pengenalan sinyal yang diujikan akan semakin besar. Basisdata yang digunakan sebaiknya direkam tidak pada suatu waktu, sehingga ragam cara pengucapan yang dimiliki seseorang akan terwakili, maka peluang pengenalan akan semakin besar.

V. Penutup Kesimpulan

Dari pengujian sistem yang telah dilakukan dapat diambil beberapa kesimpulan sebagai berikut :

1. Penggunaan pengenalan pengucap tidak berdasarkan teks sebagai otorisasi pengaksesan pintu dapat diaplikasikan dengan baik, jika hanya sedikit orang yang dapat mengakses pintu tersebut. Rata-rata persentase hasil pengujian dengan menggunakan panjang *frame* 10 ms untuk 3 responden 91,67 %, untuk 4 responden 89,38 %, dan untuk 6 responden 69,17 %. Rata-rata persentase hasil pengujian dengan menggunakan panjang *frame* 20 ms untuk 3 responden 89,17 %, untuk 4 responden 88,13 %, dan untuk 6 responden 68,75 %.
2. Rata-rata persentase hasil pengujian dengan menggunakan panjang *frame* 30 ms untuk 3 responden 84,17 %, untuk 4 responden 86,88 %, dan untuk 6 responden 68,75 %.

3. Persentase hasil pengenalan terbaik sebesar 91,17 % didapat pada pengujian 3 responden sebagai basisdata dengan panjang *frame* 10 ms.
4. Hasil pengenalan sangat berpengaruh pada panjang *frame* yang digunakan pada proses pencuplikan. Semakin lama *frame* yang digunakan semakin tidak bagus pengenalan yang dihasilkan.
5. Intonasi dari pengucap tidak begitu berpengaruh pada hasil kerja pengenalan.
6. Hasil kerja sistem sangat dipengaruhi oleh keadaan lingkungan dan kondisi dari pengucap pada saat perekaman suara untuk basisdata dan perekaman suara untuk pengujian.
7. Kecepatan waktu dalam mendapatkan hasil kerja sistem bergantung dengan banyaknya basisdata yang dilatih pada setiap perekaman suara dan kapasitas memori komputer.

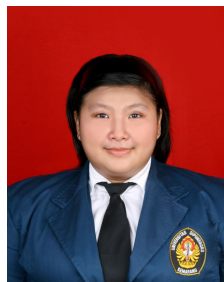
Saran

Adapun saran yang dapat diberikan sehubungan dengan pelaksanaan penelitian ini adalah :

1. Penelitian ini bisa dikembangkan dengan menggunakan program pengenalan pengucap berdasarkan teks, dan *Digital Signal Processor* sebagai unit pengolah sistem.
2. Perekaman untuk penambahan basisdata pengguna pintu baru sebaiknya dilakukan tidak lebih dari 1 detik untuk mendapatkan hasil pencirian yang lebih tepat.
3. Sistem perlu disediakan sumber tegangan cadangan yang akan digunakan jika listrik dari PLN mati atau terputus.

Daftar Pustaka

- [1]. Apriyono, Fachrudin, *Pengenalan Pengucap Tak Bergantung Teks Dengan Metode Vector Quantization (VQ) Melalui Ekstraksi Linear Predictive Coding (LPC)*, Skripsi S-1, Jurusan Teknik Elektro Universitas Diponegoro, Semarang, 2004.
- [2]. Barisevicius, Gintaras, *Text-Independent Speaker Verification*, Kaunas University of Technology, Lithuania.
- [3]. Cohen, Arnon, and Yaniv Zigel, *On Feature Selection for Speaker Verification*, Ben-Gurion University, Israel.
- [4]. Do, Minh N, *An Automatic Speaker Recognition System*, Swiss Federal Institute of Technology, Switzerland.
- [5]. Furui, Sadaoki, *Digital Speech Processing, Synthesis, and Recognition*, Marcel Dekker, Inc., New York and Basel, Musashino-Shi, Tokyo, Japan, 1989.
- [6]. Gold, Ben, and Nelson Morgan, *Speech and Audio Signal Processing : Processing and Perception of Speech and Music*, John Willey & Sons, Inc., New York, 1999.
- [7]. Liu Zhongmin, Qizhang Yin and Weimin Zhang, *A Speaker Identification and Verification System*, Final Project. Oktober, 2006.
- [8]. Malvino, Albert Paul, Diterjemahkan oleh Hanapi Gunawan, *Prinsip-prinsip Elektronika*, Erlangga, Jakarta, 1981.
- [9]. Martell, Mike, *Using Bipolar Transistor As Switches*, www.rason.org, Oktober, 2006.
- [10]. Morris, Dan, *Using Transistor As Switches*, Oktober, 2006.
- [11]. Mut, Ozan, and Mehmet Gokturk, *Improved Weighted Matching for Speaker Recognition*, Oktober, 2006.
- [12]. Phan, Till T, and Thomas Soong, *Text Independent Speaker Identification*, 1999.
- [13]. Prasetya, Ratna, dan Catur Edi Widodo, *Interfacing Port Paralel dan Port Serial Komputer dengan Visual Basing 6.0*, Andi Yogyakarta, Semarang, 2004.
- [14]. Prasetya, Nanda, *Robust Linier Prediction Analysis For Low Bit-Rate Speech Coding*, Desertasi S-3, Griffith University, Australia, 2002.
- [15]. Prasetyo, Deddy, *Sistem Pengatur Akses Pintu Menggunakan Kartu Magnetik Berbasis Mikrokontroler AT89C51*, Skripsi S-1, Universitas Diponegoro, Semarang, 2003.
- [16]. Rabiner, R. Lawrence, and Biing-Hwang Juang, *Fundamentals of Speech Recognition*, PTR Prentice-Hall Inc., New Jersey, 1993.
- [17]. Rastogi, Swati, and David Mayor, *An Automatic Speaker Recognition System*, swati.rastogi@epfl.ch.
- [18]. Sajjad, Mueen, *Parallel Port Interfacing (Lecture)*, Oktober, 2006.
- [19]. S Hodgkiss, William, David J. Defatta and Joseph G. Lucas, *Digital Signal Processing: A System Design Approach*, John Wiley & Sons Inc, Toronto, 1988.
- [20]. Wildermoth, Brett Richard, *Text-Independent Speaker Recognition Using Source Based Features*, Thesis S-2, Griffith University, Australia, 2001.
- [21]. -----, *A Tutorial on Text-Independent Speaker Verification*, EURASIP Jurnal, Oktober, 2006.



Ardha Ananda A lahir dan besar di Jakarta pada tanggal 30 Agustus 1983. Telah menjalani pendidikan di Taman Kanak-kanak BPS&K V Jakarta Timur, Sekolah Dasar BPS&K V Jakarta Timur, Sekolah Lanjutan Tingkat Pertama PUTRA I Jakarta Timur, Sekolah Menengah Umum Negeri 81 Jakarta. Dan sekarang tengah menyelesaikan pendidikan Strata Satu di Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Semarang, Indonesia.

Mengetahui,

Dosen Pembimbing I,

Dosen Pembimbing II,

Wahyudi, ST, MT
NIP. 132 086 662

Achmad Hidayatno, ST, MT
NIP. 132 137 933