Jurnal Teknik Elektro Vol. 7, No. 2, September 2007: 88 - 96

Sistem Pengenalan Penutur dengan Metode *Mel-frequency Wrapping*

Ali Mustofa
Jurusan Teknik Elektro, Universitas Brawijaya
Jl. MT. Haryono 167 Malang 65145
Phone/Fax: 0341 554166
Email: a_tofa@yahoo.com

ABSTRAK

Pengenalan penutur adalah proses identifikasi suara seseorang.. Pengenalan penutur berguna untuk otentikasi biometrik dan komunikasi antara komputer dengan manusia. Teknik *Mel Frequency Cepstral Coefficients* (MFCC) digunakan untuk ekstraksi ciri dari sinyal wicara dan membandingkan dengan penutur tak dikenal dengan penutur yang ada dalam *database*. *Filter bank* digunakan sebagai pembungkus (*wrapping*) mel frekuensi. *Vector Quantization* (VQ) adalah proses meletakkan vektor-vektor ciri yang besar dan menghasilkan ukuran vektor-vektor yang kecil yang berhubungan dengan distribusi *centroid*. Algoritma *K-mean* digunakan untuk kluster. Dalam tahap pengenalan, ukuran distorsi berdasarkan minimisasi jarak Euclidean digunakan untuk mencocokkan penutur tak dikenal dengan penutur dalam *database*. *Database* wicara menggunakan 10 penutur berbeda dengan MFCC 12, 20 *codebook*, dan 16 *centroid*.

Kata kunci: penutur, *mel-frequency cepstral coefficients, vector quantization, K-mean*

ABSTRACT

Speaker recognition is the process of identifying a person based on his voice. Speaker recognition has several useful applications including biometric authentication and intuitive human computer interaction. the Mel Frequency Cepstral Coefficients (MFCC) technique is used to extract features from the speech signal and compare the unknown speaker with the exist speaker in the database. the filter bank is used to wrap the Mel frequency. VQ (vector Quantization) is a process of taking a large set of feature vectors and producing a smaller set of measure vectors that represents the centroids of the distribution. In this method, the K means algorithm is used to do the clustering. In the recognition stage, a distortion measure which based on the minimizing the Euclidean distance was used when matching an unknown speaker with the speaker database. Speech database used 10 different speakers with MFCC 12, 20 codebooks, and 16 centroids.

Keywords: *speaker, mel-frequency cepstral coefficients, vector quantization, K-mean*

PENDAHULUAN

Suara digunakan oleh manusia untuk berkomunikasi. Suara manusia berguna untuk menyampaikan ide, keinginan, informasi kepada manusia lainnya. Lintasan vokal manusia dan artikulasi adalah organ biologi dengan sifat tak linier dan ini beroperasi tidak hanya dibawah kontrol kesadaran tetapi juga dipengaruhi oleh sifat gender dan keadaan emosional [1]. Oleh sebab itu ucapan manusia bervariasi membentuk pengenalan wicara dan ini mempunyai masalah yang sangat kompleks [2].

Secara garis besar sistem pengenalan wicara merupakan suatu usaha untuk dapat menghasilkan suatu mesin cerdas yang mampu mengenali ucapan manusia. Kesulitan yang paling mendasar adalah bagaimana melakukan ekstraksi terhadap sinyal ucapan menjadi beberapa parameter yang dapat digunakan untuk klasifikasi ucapan secara efisien.

Catatan: Diskusi untuk makalah ini diterima sebelum tanggal 1 Desember 2007. Diskusi yang layak muat akan diterbitkan pada Jurnal Teknik Elektro volume 8, nomor 1, Maret 2008.

Pengenalan penutur adalah proses secara otomatis mengenali siapa yang bicara dengan dasar informasi individu yang mengandung gelombang wicara [3] . Teknik ini memungkinkan menggunakan suara penutur untuk memverifikasi identitas wicara dan mengontrol layanan seperti menekan nomor telepon dengan suara (*voice dialing*), perbankan dengan telepon, belanja melalui telepon, layanan akses melalui basis data (*database*), layanan informasi, surat dengan suara (*voice mail*), kontrol keamanan area rahasia, dan akses jarak jauh dengan komputer.

Penelitian ini membangun sistem pengenalan penutur secara otomatis. Semua penutur mengucapkan satu kata tunggal yang sama dalam pelatihan dan akan diuji (*testing*) kemudian. Daftar bahasa yang digunakan adalah kata sering digunakan dalam pengetesan pengenalan penutur karena sering digunakan untuk berbagai aplikasi. Sebagai contoh, pengguna harus mengucapkan PIN (*Personal Identification Number*) untuk membuka pintu laboratorium, atau pengguna harus mengucapkan nomer kartu kredit melalui saluran telepon. Dengan memeriksa karakteristik

suara dari input ucapan dengan menggunakan sistem pengenalan penutur otomatis, sistem ini dapat ditambahkan tingkat keamanan.

Dalam penelitian ini dilakukan proses pengenalan penutur dengan menggunakan metode *Mel-frequency Wrapping*. Dengan pemrosesan *Mel-frequency Wrapping* ini adalah menirukan perilaku dari pendengaran manusia sehingga dapat mengenali ucapan dari penutur.

PENGENALAN PENUTUR

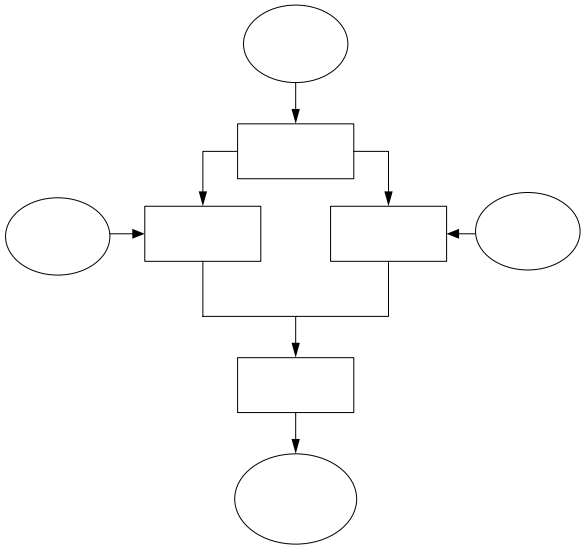
Pengenalan penutur dapat diklasifikasikan menjadi identifikasi dan verifikasi. Identifikasi penutur adalah proses penentuan penutur yang terdaftar sesuai dengan ucapannya. Verifikasi penutur adalah proses diterima atau ditolaknya klaim identitas penutur [4].

Metode pengenalan penutur dibagi menjadi metode *text-independent* dan *text-dependent*. Dalam sistem dengan ucapan bebas (*text-independent*), model penutur menangkap karakteristik wicara seseorang dengan kata yang bebas. Dalam sistem ucapan tertentu (*text-dependent*), pengenalan identitas penutur berdasarkan frasa yang spesifik, seperti kata sandi (*password*), nomer kartu kredit, kode PIN, dan sebagainya [5].

Semua teknologi pengenalan penutur, identifikasi dan verifikasi, *text-independent* dan *text-dependent*, masing-masing mempunyai keuntungan dan kelemahan dan mungkin memerlukan perlakuan dan teknik yang berbeda. Pemilihan teknologi yang digunakan tersebut harusnya disesuaikan dengan aplikasi tertentu

Pada tingkat tertinggi, semua sistem pengenalan penutur berisi dua modul utama seperti dalam Gambar 1 yaitu ekstraksi ciri dan penyesuaian ciri. Ekstraksi ciri adalah proses yang mengekstrak sejumlah kecil data dari sinyal suara kemudian digunakan untuk merepresentasikan masing-masing penutur. Penyepadanan ciri melibatkan prosedur untuk mengidentifikasi penutur tak dikenal dengan membandingkan ciri-ciri terekstraksi dari input suara pria atau wanita dengan kumpulan (*set*) penutur yang dikenal.

Semua sistem pengenalan penutur harus melayani dua fasa yang berbeda. Pertama mengacu pada bagian yang terdaftar atau fasa pelatihan kemudian yang kedua mengacu pada bagian operasi atau fasa pengetesan. Dalam fasa pelatihan, masing-masing penutur yang terdaftar mempunyai sampel wicaranya sehingga sistem dapat melatih model referensi wicaranya tersebut. Dalam sistem verifikasi ambang spesifik penutur dikomputasi dari sampel-sampel pelatihan. Selama fasa pengetesan (operasional), *input* wicara disepadankan dengan model-model referensi yang disimpan dan keputusan pengenalan dibuat [5].



Gambar 1. Struktur dasar sistem pengenalan penutur.

Pengenalan penutur adalah tugas yang sulit dan ini masih diteliti lebih lanjut. Pengenalan penutur otomatis bekerja berdasarkan karakteristik wicara seseorang/*person* yang unik dalam wicaranya. Akan tetapi tugas ini tertantang oleh varians input sinyal wicara. Prinsip sumber varians berasal dari penutur-penutur itu sendiri. Sinyal wicara dalam bagian pelatihan dan pengetesan dapat berbeda karena banyak fakta seperti perubahan suara karena waktu, kondisi kesehatan (karena sakit flu), tingkat bicara, dan sebagainya [5]. Ada juga karena faktor lain, kemajemukan penutur, sehingga menantang teknologi pengenalan penutur. Contoh ini adalah *noise* akustik dan variasi dalam lingkungan perekaman data (seperti penutur menggunakan perangkat rekaman) yang berbeda.

EKSTRAKSI CIRI WICARA

Dalam penelitian ini mengubah bentuk gelombang wicara dengan beberapa jenis representasi parametrik (pada laju informasi rendah).

Sinyal wicara mempunyai waktu lambat terhadap perubahan sinyal (atau disebut *quasi-stationary*). Jika ini diujikan dengan periode waktu yang cukup singkat (antara 5 dan 100 msec), karakteristiknya cukup stasioner. Akan tetapi, dengan periode waktu yang panjang (pada 1/5 sec atau lebih) karakteristik sinyal

berubah terhadap reflek suara wicara yang dihasilkan. Analisis spektral waktu singkat adalah cara paling umum digunakan untuk mengkarakterisasi sinyal wicara.

PEMROSESAN KOEFISIEN *MEL-FREQUENSCY CEPSTRUM*

Tujuan utama dari pemroses MFCC adalah meniru-kan perilaku dari pendengaran manusia. Adapun prosesnya sebagai berikut.

Frame Blocking

Dalam langkah ini sinyal wicara kontinyu diblok menjadi *frame-frame N* sampel, dengan *frame-frame* berdekatan dengan spasi *M* (*M* < *N*). *Frame* pertama terdiri dari *N* sampel pertama. *Frame* kedua dengan *M* sampel setelah *frame* pertama, dan overlap dengan *N–M* sampel. Dengan cara yang sama, *frame* ketiga dimulai 2*M* sampel setelah *frame* pertama (atau *M* sampel setelah *frame* kedua) dan *overlap* dengan *N*–2*M* sampel. Proses ini berlanjut hingga semua wicara dihitung dalam satu atau banyak *frame*. Nilai tipikal untuk *N* dan *M* adalah *N* = 256 dan *M* =100.

Windowing

Langkah berikutnya adalah pemrosesan dengan *window* pada masing-masing *frame* individual untuk meminimalisasi sinyal tak kontinyu pada awal dan akhir masing-masing *frame*. *Window* dinyatakan sebagai $w(n)$, $0 \le n \le N-1$, dengan N adalah jumlah sampel dalam masing-masing *frame*, $x_1(n)$ adalah sinyal input dan hasil *windowing* adalah $y_1(n)$.

$$y_1(n) = x_1(n)w(n), \quad 0 \le n \le N-1 \qquad (1)$$

Jenis *window* yang digunakan adalah *window* Hamming [4].

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

dengan *N* adalah jumlah sampel.

Transformasi Fourier Cepat

Langkah pemrosesan berikutnya adalah transformasi fourier cepat/ *fast fourier transform* (FFT), FFT ini mengubah masing-masing *frame N* sampel dari domain waktu menjadi domain frekuensi. FFT adalah algoritma cepat untuk mengimplementasikan *discrete fourier transform* (DFT) dengan didefinisikan pada kumpulan (*set*) *N* sampel, $\{X_n\}$, seperti berikut ini [7].

$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N}, \quad n = 0,1,2,\ldots, N-1 \qquad (3)$$

dengan,

x_k = deretan aperiodik dengan nilai *N*
N = jumlah sampel

Mel-Frequency Wrapping

Studi psikofisikal menunjukkan bahwa persepsi manusia dari kandungan frekuensi suara pada sinyal wicara tidak mengikuti skala linier. Untuk masing-masing nada dengan frekuensi aktual, *f* dalam Hz, *pitch* diukur dengan skala '*mel*'. Skala *mel-frequency* adalah frekuensi linier berada dibawah 1000 Hz dan bentuk logaritmik berada diatas 1000 Hz. Sebagai titik referensi adalah *pitch* dengan *tone* 1 kHz, 40 dB diatas nilai batas ambang pendengaran, ini dinyatakan 1000 *mel*. Pendekatan persamaan untuk menghitung *mel* dalam frekuensi *f* (Hz) adalah [1][6].

$$mel(f) = 2595 \text{ x } \log_{10}(1 + f/700) \qquad (4)$$

Salah satu pendekatan simulasi spektrum yaitu menggunakan *filter* bank, satu *filter* untuk masing-masing komponen *mel-frequency* yang diinginkan. *Filter* bank mempunyai respon frekuensi *bandpass* segitiga dan jarak *bandwidth* ditentukan oleh konstanta interval *mel-frequency*.

Cepstrum

Langkah selanjutnya yaitu mengubah spektrum *log mel* menjadi domain waktu. Hasil ini disebut *mel frequency cepstrum coefficient* (MFCC). Reprentasi cepstral dari spectrum wicara memberikan reprentasi baik dari sifat-sifat spektral lokal sinyal untuk analisis *frame* yang diketahui. Karena koefisien *mel* spektrum adalah bilangan nyata. Dengan mengubahnya menjadi domain waktu menggunakan *discrete cosine transform* (DCT). Jika koefisien spektrum daya *mel* hasilnya adalah $\widetilde{S}_k, k = 1,2,\ldots, K,$ sehingga MFCC dapat dihitung, $\widetilde{c}_n$ adalah [8]

$$\widetilde{c}_n = \sum_{k=1}^{K} \left(\log \widetilde{S}_k\right)\cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{K}\right], \quad n = 1,2,\ldots, K \qquad (5)$$

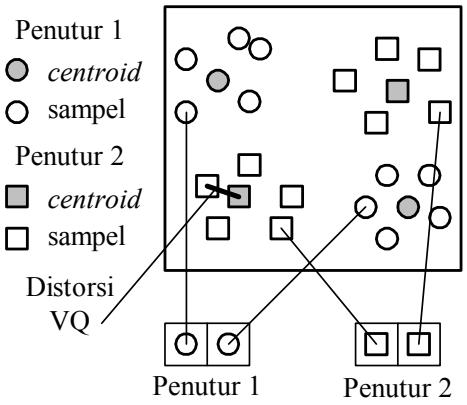
Dimana $\widetilde{c}_n$ adalah koefisien cepstrum *mel-frequency* dan $\widetilde{S}_k$ adalah koefisien daya *mel.*

Vektor Kuantisasi

VQ atau vektor kuantisasi adalah proses memetakan vektor-vektor dari ruang vektor besar menjadi jumlah terbatas daerah ruang vektor. Masing-masing daerah disebut kluster dan dapat direpresentasikan oleh pusatnya yang disebut *codeword*. Kumpulan dari semua *codeword-codeword* disebut *codebook*.

Dalam Gambar 5 menunjukkan konsep diagram untuk ilustrasi proses pengenalan. Hanya dua penutur dan dua dimensi dari ruang akustik ditunjukkan. Putaran-putaran mengacu pada vektor-vektor akustik dari penutur 1 dengan tanda lingkaran dan penutur 2 dengan tanda kotak. Dalam fasa pelatihan, VQ *codebook* penutur-spesifik dibangkitkan untuk masing-masing penutur yang dikenali oleh pengklusteran vektor-vektor akustik pelatihan dari laki-laki atau wanita.

Hasil *codeword-codeword* (*centroid*) ditunjukkan dalam Gambar 5 dengan tanda lingkaran hitam dan tanda kotak hitam untuk penutur 1 dan 2. Jarak terdekat antara vektor *codeword* dari *codebook* disebut distorsi VQ. Dalam fasa pengenalan ini, input wicara dari suara tak dikenal adalah "vektor terkuantisasi" dengan menggunakan masing-masing *codebook* yang dilatih dan jarak total distorsi VQ. Penutur dengan VQ *codebook* dan total distorsinya terkecil akan diidentifikasi.



Gambar 5. Formasi VQ *codebook* antara penutur 1 dan penutur 2.

Pelatihan Vektor-Vektor

Selajutnya vektor-vektor akustik diekstraksi dari input wicara dari seorang penutur sebagai *set* pelatihan vektor-vektor. Sebagaimana penjelasan diatas, langkah penting berikutnya adalah membangun VQ *codebook* dari penutur yang spesifik dengan menggunakan pelatihan vektor-vektor ini. Algoritma ini dikenal sebagai algoritma LBG (Linde, Buzo, dan Gray) [9], untuk kluster *set L* pelatihan vektor-vektor menjadi *set M codebook* vektor-vektor. Algoritma ini secara formal diimplementasikan dengan prosedur rekursif berikut ini:

1. Desain satu vektor *codebook*, ini adalah *centroid* dari masukan set pelatihan vektor-vektor (karenanya tak diperlukan iterasi disini).
2. Gandakan ukuran *codebook* dengan membagi masing-masing *codebook* sekarang y_n sesuai dengan aturan

$$y_n^+ = y_n(1+\varepsilon)$$
$$y_n^- = y_n(1-\varepsilon) \quad (6)$$

dimana n berubah dari 1 ke ukuran *codebook* sekarang dan ε adalah parameter pembagi (*splitting*) (misalnya $\varepsilon = 0.01$)

3. Pencarian *neighbor*(tetangga) terdekat: untuk masing-masing pelatihan vektor, tentukan *codebook* dalam *codebook* yang terdekat dan menetapkan vektor-vektor tersebut yang berhubungan dengan sel (berhubungan dengan *codebook* terdekat).
4. *Centroid* terbarukan (*update*): mem-perbarui *codebook* dalam masing-masing sel dengan menggunakan *centroid* dari pelatihan vektor-vektor ini yang ditentukan untuk sel tersebut.
5. Iterasi 1: mengulangi langkah 3 dan 4 sampai jarak rata-rata jatuh dibawah nilai ambang.
6. Iterasi 2: mengulangi langkah 2, 3, dan 4 sampai *codebook* dengan ukuran *M* didesain. Algoritma LBG mendesain *M* vektor *codebook* dalam langkah ini. Langkah pertama dengan mendesain satu vektor *codebook*, kemudian menggunakan teknik pemecahan (*splitting*) pada *codebook-codebook* untuk menginialisasi pencarian untuk 2 vektor *codebook* dan melanjutkan proses pemecahan sampai *M* vektor *codebook* yang diinginkan akan dapat ditentukan.

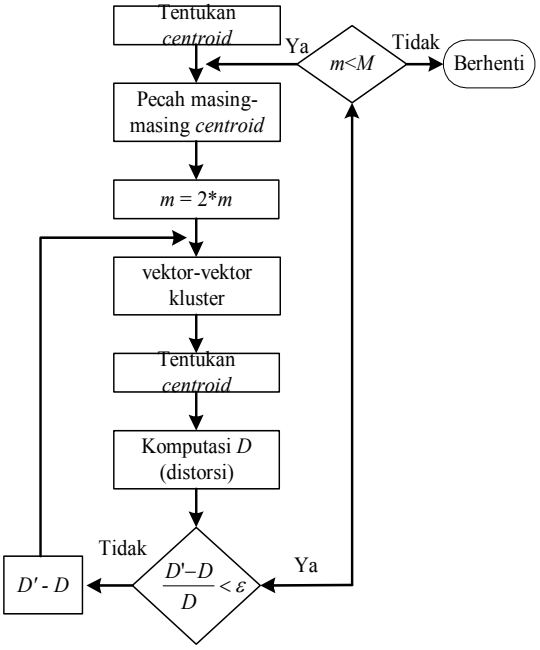
Dalam Gambar 6 menunjukkan langkah detil algoritma LBG. Kluster vektor-vektor adalah *neighbor* terdekat yang menentukan masing-masing pelatihan vektor pada kluster yang berhubungan dengan *codebook* terdekat. "Penentuan *centroid*" adalah prosedur *centroid* terkini. "Mengkomputasi *D* (distorsi)" yaitu menjumlahkan jarak semua pelatihan vektor-vektor pada *neighbor* terdekat dan menentukan apakah prosedurnya telah konvergen.

Algoritma *K-Means*

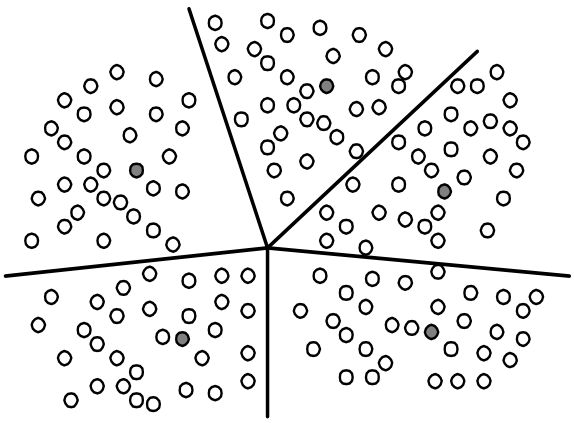
Algoritma *K-mean* adalah cara untuk mengkluster vektor-vektor pelatihan untuk mendapatkan vektor-vektor ciri. Dalam algoritma ini vektor-vektor dikluster berdasarkan atribut menjadi *K* partisi. Ini menggunakan *K-mean* data dengan distribusi gaussian untuk mengkluster vektor-vektor tersebut. Tujuan *K-mean* adalah untuk meminimkan total varians intra-kluster, *V*.

$$V = \sum_{i=1}^{k} \sum_{j \in s_i} \left| x_j - \mu_i \right|^2 \quad (7)$$

dimana ada *K* kluster $S_i, i = 1,2,3,...,k$ dan μ_i adalah *centroid* atau titik *mean* dari semua titik $x_j \in S_i$



Gambar 6. Diagram alir algoritma LBG.



Gambar 7. Ilustrasi *K-mean* membentuk lima kluster.

Pengukuran Jarak

Dalam tahap pengenalan penutur, suara penutur yang tak dikenal direperentasikan oleh deretan vektor-vektor ciri $\{x_1, x_2 \ldots x_i\}$,dan kemudian ini dibandingkan dengan *codebook* dari *database*. Untuk mengidentifikasi pembicara yang tak dikenali, ini dapat dilakukan dengan pengukuran jarak distorsi dari dua kumpulan vektor yang berdasarkan peminiman jarak Euclidean. Jarak Euclidean adalah jarak antar dua titik yang akan diukur dengan suatu aturan, yang dapat dibuktikan oleh aplikasi teorema *Pythagorean*. Persamaan yang digunakan untuk menghitung jarak Euclidean dapat didefisikan dengan jarak Euclidean antara dua titik $P = (p_1, p_2 \ldots p_n)$ dan $Q = (q_1, q_2 \ldots q_n)$.

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2}$$

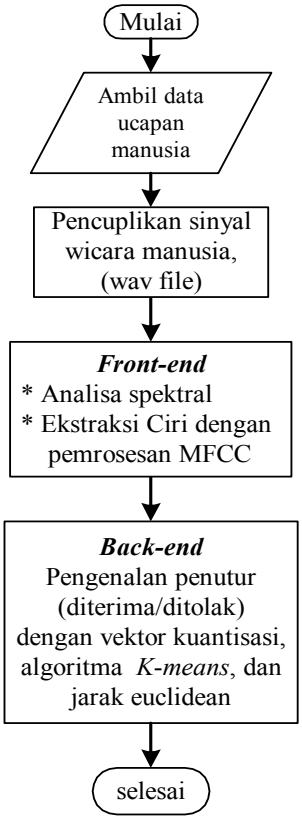
$$= \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \qquad (8)$$

Penutur dengan jarak distorsi terkecil dipilih untuk diidentifikasi seperti orang yang tak dikenal.

METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah pertama pengambilan sampel dilakukan sebanyak 10 orang penutur dengan masing-masing penutur mengucapkan satu pola kata yang telah ditentukan. Kata-kata tersebut adalah kata /kampus/. Pola kata dari masing-masing penutur tersebut disimpan dan kemudian dilatihkan secara bersamaan ke dalam sistem pengenalan penutur. Yang kedua adalah memroses koefisien *mel-frequency cepstrum* tujuannya adalah menirukan perilaku dari pendengaran manusia. Yang ketiga adalah proses pelatihan. Dalam proses pelatihan pola kata dimasukkan secara urut mulai penutur 1 dengan pola kata /kampus/. Kemudian penutur 2 dengan pola kata yang sama, demikian seterusnya sampai pembicara ke-10 (pelatihan *data set*). Dalam proses pelatihan ini untuk mengenali pola kata yang dilatihkan sesuai target yang ditentukan pula. Yang keempat adalah menguji penutur dengan MFCC dan VQ untuk mengenali penutur. Yang kelima yaitu menganalisa dan pengambilan kesimpulan.

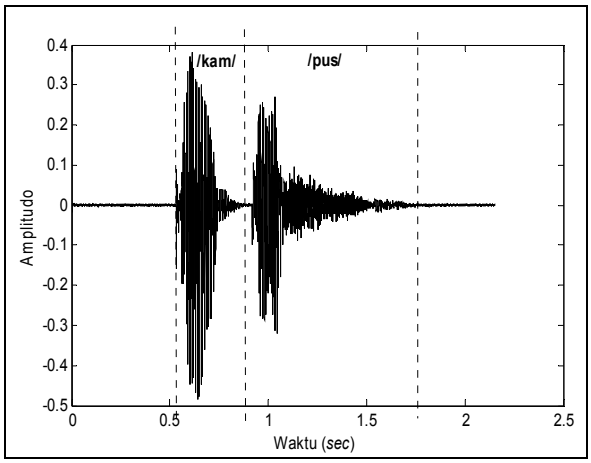
Lebih jelasnya tahapan penelitian ini dapat dilihat dalam Gambar 8.



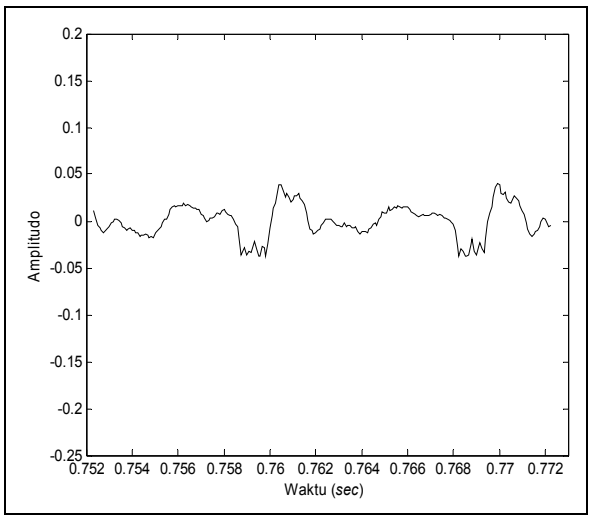
Gambar 8. Diagram alir sistem pengenalan penutur dengan metode *mel-frequency*.

HASIL DAN PEMBAHASAN

Berdasarkan hasil pengujian yang dilakukan berulang-ulang dengan frekuensi *sampling* 12 kHz, ternyata suatu sinyal wicara memiliki suatu ciri yang istimewa. Suatu sinyal wicara merupakan suatu fungsi yang bergantung waktu. Walaupun demikian pada suatu selang waktu tertentu yaitu kira-kira sepanjang 20 ms, sinyal tersebut merupakan fungsi yang tidak bergantung waktu. Pada analisa ini akan diberikan sebuah contoh suatu bentuk sinyal wicara dari suatu penutur. yang mengucapkan kata /kampus/ selama 2.1535 *second*.



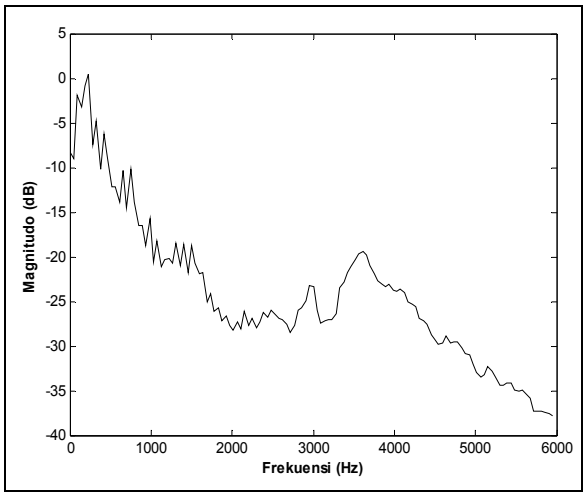
Gambar 9. Bentuk sinyal wicara /kampus/ sepanjang 2.1535 *second*.



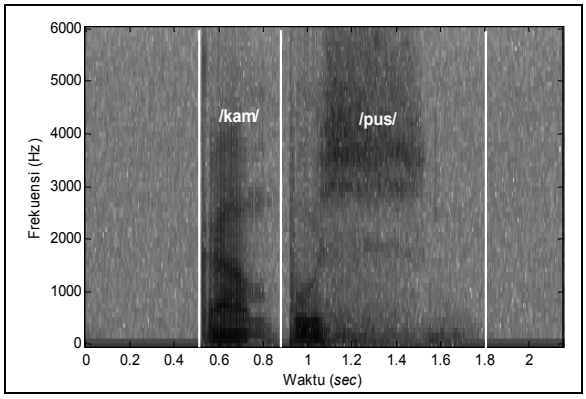
Gambar 10. Bentuk sinyal sepanjang 20 ms.

ANALISA DOMAIN FREKUENSI

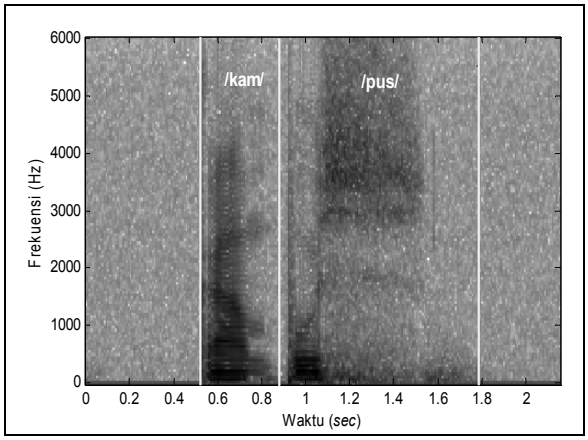
Setelah pemrosesan dengan *window*, maka akan dianalisa sinyal dalam domain frekuensi yaitu mengubah domain waktu ke domain frekuensi dengan menggunakan transformasi *Fourier*. Dan hasilnya sinyal tersebut akan dinyatakan dalam bentuk decibel (dB). Setelah proses pe-*window*-an dan transformasi fourier maka dapat digambarkan spektrumnya dengan panjang *window* atau jumlah sampel per *frame* (*N*) adalah 256 dan pergeseran ke *frame* berikutnya (*M*) adalah 100.



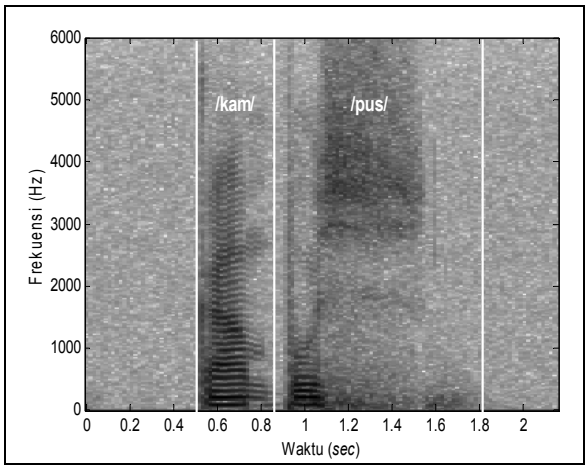
Gambar 11. Spektum sinyal dengan panjang *window, N* = 256 dan *M* = 100 pada kata /kampus/.



Gambar 12. Spektogram sinyal wicara /kampus/, *M* = 50 dan *N* = 128.



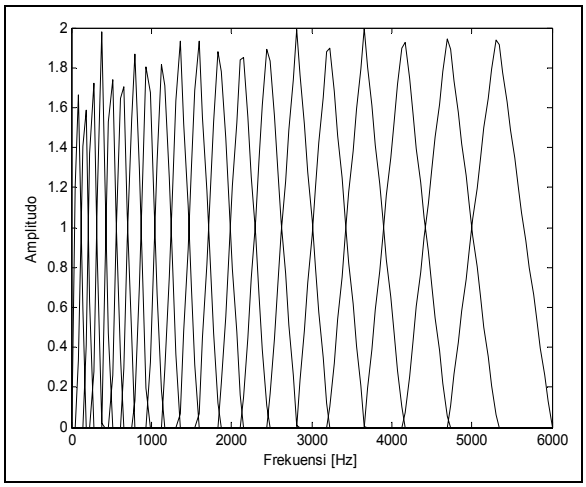
Gambar 13. Spektogram sinyal wicara /kampus/, *M* = 100 dan *N* = 256.



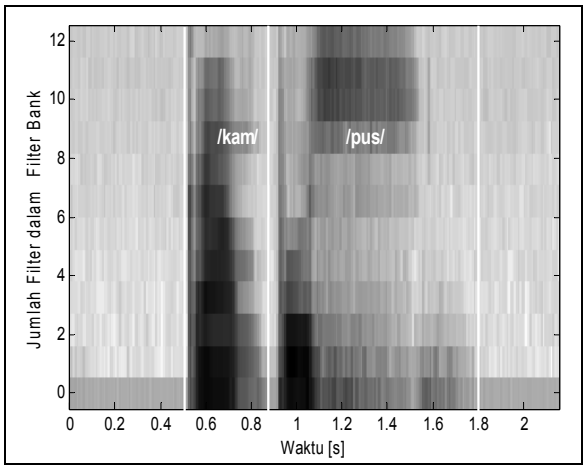
Gambar 14. Spektogram sinyal wicara /kampus/, *M* = 200 dan *N* = 512.

Untuk *N*=128 mempunyai resolusi tinggi terhadap waktu. *Frame* mempunyai periode waktu sangat singkat. Hasil ini menunjukkan bahwa sinyal untuk sebuah *frame* tidak mengubah alamiahnya (untuk vokal atau konsonan yang sama). Untuk *N*=256 mempunyai kompromi antara resolusi waktu dan resolusi frekuensi.Untuk *N*=512 mempunyai resolusi frekuensi yang bagus tetapi ada *frame-frame* yang kurang, artinya bahwa resolusi dalam waktu direduksi dengan kuat. Nilai *N*=256 adalah kompromi yang dapat diterima. Lebih jauh jumlah *frame* adalah relatif lebih kecil, sehingga mengurangi waktu komputasi.

Selanjutnya *filter bank* yang digunakan untuk proses *mel* frekuensi ada 20 dan frekuensi *sampling*-nya 12000 Hz hasilnya seperti dalam Gambar 16.



Gambar 15. Hasil filter bank dalam proses mel frekuensi.



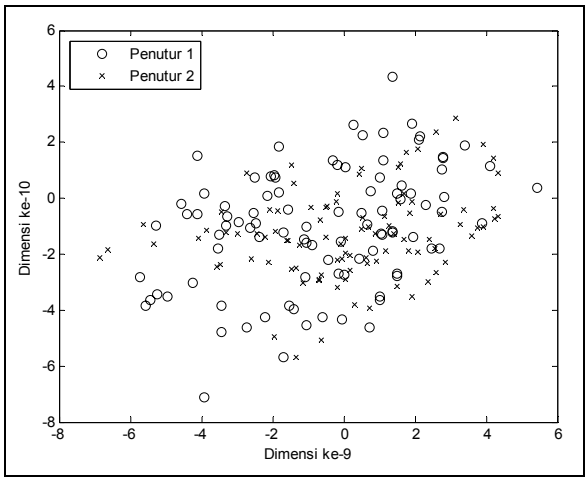
Gambar 16. Spektrum daya dimodifikasi dengan *mel* cepstrum *filter* (*M*=100, *N*=256).

ANALISA MFCC DAN VQ

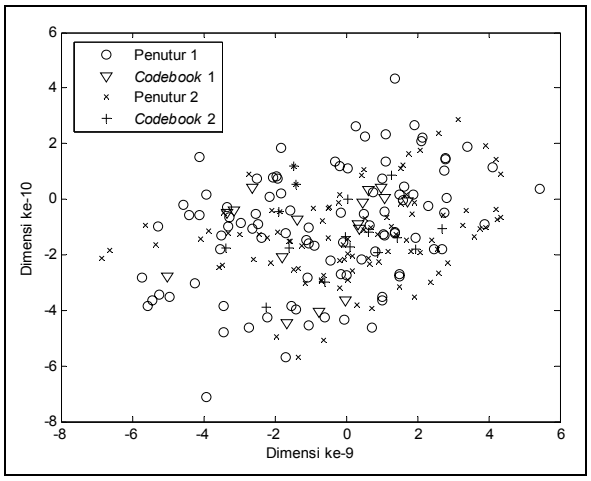
Kinerja sistem pengenalan penutur tergantung pada besarnya data *test* dan data *train*. dalam rekaman, peralatan yang digunakan dalam proses rekaman suara dan jumlah penutur yang sedikit dalam kelompok pelatihan dan pengetesan. Laju kebenaran untuk seluruh sistem identifikasi sistem adalah 100% dengan proses rekaman yang sangat cermat dan pengaturan parameter-parameter yang relevan, seperti ukuran *codebook,* jumlah iterasi dan lain-lain. Uji vektor kuantisasi menggunakan koefisien mel frekuensi 12, filter bank 20 dan 16 centroid.

Pelatihan dilakukan dengan 10 penutur dengan mengucapkan kata /kampus/, masing-masing penutur berbeda, dengan proses *mel frequency wrapping* dan vektor kuantisasi. Gambar 17 dan Gambar 18 menunjukkan plot dua dimensi yang dilatih dengan vektor-vektor MFCC untuk dua penutur dengan dimensi ke-9 dan ke-10 dengan 12 koefisien *mel frequency* dan perbandingan dua dimensi untuk plot *codebook* yang dibangkitkan oleh dua penutur menggunakan dua dimensi yang sama dengan ukuran codebook 16. Masing-masing *codeword* merepresentasikan hubungan kluster dengan titik-titik data MFCC dan secara akurat merepresentasikan karakteristik suara masing-masing penutur.

Dalam vektor kuantisasi, jarak ecluidean dihitung antara kata dan *codebook* yang tak dikenal, kemudian nilai terendah dari jarak tersebut diidentifikasi sebagai suara penutur yang benar. Hasil untuk ukuran *codebook* 16 untuk 10 penutur adalah 32.889, 38.263, 41.579, 41.004, 50.192, 35.520, 47.696, 47.403, 56.719, dan 29.044.



Gambar 17. Sebaran vektor-vektor akustik dari dua penutur dalam proses mfcc 12 dan *filter* bank 20, dan 16 centroid (*N*=256, *M*=100).



Gambar 18. Sebaran vektor-vektor akustik dari dua penutur dan *codebook*-nya, mfcc 12, *filter* bank 20, dan 16 centroid (*N*=256,*M*=100).

Tabel 1. Hubungan Jarak Euclidean antar Penutur untuk Empat Penutur

	Penutur 1	**Penutur 2**	**Penutur 3**	**Penutur 4**
Penutur 1	5.0609	14.8146	15.0358	13.0911
Penutur 2	17.4570	5.7034	12.8706	13.5058
Penutur 3	16.6471	13.2394	5.9544	12.5522
Penutur 4	14.4681	13.2555	11.7277	5.7707

Akan tetapi, hal ini sukar untuk memberikan nilai ambang untuk jarak pada penutur lain yang mencoba mengakses *database* dan penutur yang ada dalam *database* pelatihan berdasarkan urutan pengetesan, sebagai ilustrasi dalam Tabel 1. menunjukkan nilai-nilai test vektor kuantisasi artinya bahwa penutur 1 mempunyai jarak yang kecil dengan penutur 1 dibandingkan dengan penutur-penutur lainnya, maka penutur 1 cocok dengan penutur 1. Sedangkan dalam Tabel 2. menunjukkan jarak Euclidean terkecil untuk sepuluh penutur. Karena nilai ambang bervariasi untuk masing-masing kata, sehingga jika nilai ambang ditentukan, maka banyak nilai-nilai lain menjadi terlalu tinggi atau terlalu rendah. Sehingga keterbatasan sistem ini adalah orang-orang yang belum dilatih dalam sistem ini masih dapat lolos dengan melalui algoritma vektor kuantisasi. Keterbatasan ini tidak diharapkan. Untuk penutur yang dilatih dalam *database* untuk mengakses sistem ini, vektor kuantisasi membantu untuk mengolah *password* penutur dalam *database* dan memperbaiki keamanan dari keseluruhan sistem dan melayani keseluruhan sistem keamanan.

Tabel 2. Jarak Euclidean Terkecil untuk Masing-Masing Penutur

	Jarak *euclidean*
Penutur 1	5.0609
Penutur 2	5.7034
Penutur 3	5.9544
Penutur 4	5.7707
Penutur 5	6.5507
Penutur 6	5.4490
Penutur 7	6.4143
Penutur 8	6.2931
Penutur 9	6.7853
Penutur 10	5.0141

KESIMPULAN

Penelitian ini untuk membuat sistem pengenalan penutur. Ekstraksi ciri wicara dari penutur tak dikenal dan dibandingkan dengan ekstraksi ciri dari penutur yang ada dalam *database*. Ekstraksi ciri menggunakan *mel frequency wraping* yaitu dengan MFCC. Fungsi mel cepstrum digunakan untuk menghitung sinyal mel. Penutur dimodelkan dengan menggunakan VQ. *Codebook* VQ dibangkitkan uleh kluster dari pelatihan vektor-vektor ciri dari masing-masing penutur dan disimpan dalam *database*. Dalam metode ini, algoritma *K-mean* digunakan untuk kluster. Dalam tahap pengenalan penutur, distorsi diukur berdasarkan minimisasi jarak Euclidean yang digunakan saat mencocokkan (*matching*) penutur tak dikenal dengan *database* penutur. Dengan MFCC dan VQ pengenalan penutur dapat digunakan untuk identifikasi penutur.

UCAPAN TERIMAKASIH

Terima kasih ditujukan pada Fakultas Teknik Universitas Brawijaya yang telah membiayai penelitian ini melalui anggaran DPP tahun 2007.

DAFTAR PUSTAKA

[1] Sigurdsson S, Petersen K.B dan Schiøler TL, "*Mel Frequency Cepstral Coefficients: An Evaluation of Robustness of MP3 Encoded Music"*, University of Victoria, 2006.

[2] Kuldip K.P dan Bishnu S.A, "*Frequency-Related Representation of Speech*", EUROSPEECH Seminar 2003 - Geneva

[3] Irino T, Minami Y, Nakatani T, Tsuzaki M, dan Tagawa H, "*Evaluation of a Speech Recognition/ Generation Method Based on HMM and Straight*", Presented at ICSLP2002 Denver, Colorado

[4] Rabiner L.R dan Juang B. H, "*Fundamentals of Speech Recognition*", Prentice-Hall, Englewood Cliffs, N.J., 1993.

[5] Furui S, "*An overview of speaker recognition technology*", ESCA Workshop on Automatic Speaker Recognition, Identification and Verification, pp. 1-9, 1994.

[6] Xu, Tan, Dalsgaard dan Lindberg ,"*Exploitation of spectral variance to improve robustness in speech recognition*", Electronic Letters, 2nd March 2006 Vol. 42 No. 5

[7] Ludeman, L.C, "*Fundamentals of Digital Signal Processing*", Happer & Row Publishers, New york,1986

[8] Song F.K, Rosenberg dan Juang B.H , "*A vector quantisation approach to speaker recognition*", AT&T Technical Journal, Vol. 66-2, pp. 14-26, March 1987.

[9] Furui, S, "*Digital Speech Processing, Synthesis, and Recognition*", Marcel Dekker Inc.New York, 1989