Metoda Mel Frequency Cepstrum Coefficients (MFCC) untuk Mengenali Ucapan pada Bahasa Indonesia

Torkis Nasution
Jurusan Manajemen Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer AMIK Riau, Jl. Purwodadi Indah Km. 10,3 Panam – Pekanbaru 28299
torkisnasution@stmik-amik-riau.ac.id
torkisnasution@yahoo.com

Abstrak

Sampai saat ini belum ada suatu aplikasi yang dapat digunakan untuk mengubah ucapan dalam bahasa Indonesia menjadi tulisan yang memenuhi kaidah penulisan bahasa Indonesia. Kajian untuk mengubah ucapan menjadi tulisan, setakat ini baru berada pada pengubahan ucapan abjad untuk diterjemahkan menjadi huruf. Sementara, jika ucapan melalui bahasa Indonesia dapat di ubah ke dalam tulisan akan dapat menambah pola penyebaran informasi di kalangan akademis, pemerintahan dan masyarakat secara luas dan adaptif. Di dalam pertemuan ilmiah, non ilmiah, interogasi, dan pidato politik yang umumnya tidak menggunakan teks book sebagai media penyampai secara baku. Audien yang disasar oleh informasi yang diciptakan oleh pertemuan tersebut akan lebih merata, luas, dan seluruh strata. Walau suara dapat menjadi media penyampai informasi namun keberagaman kemasan yang dibuat dapat meningkatkan penetrasi informasi pada seluruh lapisan strata masyarakat. Konstruksi perangkat lunak dibuat dengan menggunakan metode MFCC (Mel Frequency Sepstrum Coefficients) feature extraction dan di dukung dengan K-Means clustering. MFCC feature extraction mengekstrak signal suara ke dalam beberapa vektor data. Hasil dari MFCC feature extraction mempunyai ukuran yang sangat besar, sehingga digunakanlah K-Means clustring untuk membuat beberapa vektor pusat sebagai wakil dari keseluruhan vektor data yang ada untuk digunakan dalam proses pengenalan sehingga mempersingkat waktu. Penelitian ini akan menghasilkan teknologi berupa aplikasi yang dapat di gunakan dengan baik serta diberi keleluasaan untuk dikembangkan pada seluruh bagian sehingga lebih adaptif dan inovatif.

1. Pendahuluan

Semakin hari aktivitas manusia semakin kompleks dan semakin tidak dapat dipisahkan dengan teknologi yang ada. Keberadaan teknologi diharapkan dapat memudahkan manusia dalam melaksanakan aktivitas-aktivitasnya. Oleh karena itu, teknologi yang ada di tuntut untuk semakin mudah, efektif dan efisien dalam penggunaannya. Perangkat lunak ini merupakan cikal bakal munculnya perangkat lunak pengenalan suara. Perangkat lunak pengenalan suara adalah suatu aplikasi yang memungkinkan manusia dalam menggunakan teknologi khususnya komputer, tidak perlu berhubungan secara langsung. Melainkan, cukup dengan memberikan perintah-perintah secara lisan kepada komputer selayaknya memberikan perintah kepada orang lain.

Selain itu dalam kehidupan sehari-hari kemampuan pendengaran manusia bervariasi antara satu dengan yang lainnya. Dalam suatu rapat yang berlangsung di ruangan yang luas dengan penataan audio yang kurang memadai memungkinkan peserta rapat dapat mendengar peserta rapat lainnya mengatakan sesuatu tetapi tidak dapat mendengar dengan jelas apa yang dikatakan atau siapa yang mengatakan apa.

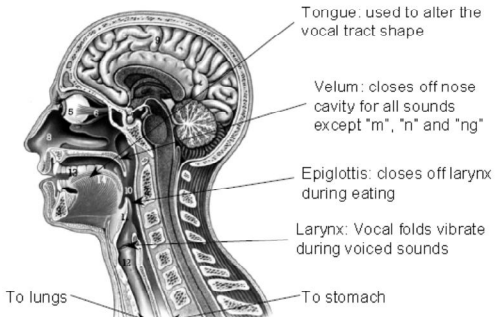
Dokumentasi tentang apa yang dibahas dalam suatu rapat sangatlah penting. Sarana yang diapakai sekarang ini untuk melakukan dokumentasi antara lain dengan merekam semua pembicaraan yang terjadi selama konferensi, kemudian mendengarkan kembali dan mencatat hal-hal yang dibicarakan. Ketika mendengarkan kembali apa yang telah direkam selama konferensi sering terdapat suara-suara lain (*noise*) yang menyulitkan untuk menangkap apa yang dikatakan oleh pembicara untuk ditulis kembali. Selain itu terdapat kesulitan

dalam mengali pembicaraan. Sehingga diharapkan perangkat lunak ini dapat menjadi cikal bakal dalam membantu melakukan hal-hal diatas dan menghemat waktu yang digunakan.

2. Proses Produksi Suara

Suara adalah sebuah signal yang merambat melalui media perantara. Suara dapat dihantarkan dengan media air, udara maupun benda padat. Dengan kata lain suara adalah gelombang yang merambat dengan frekuensi dan amplitudio tertentu. Suara yang dapat didengar oleh manusiaa berkisar antara 20 Hz sampai dengan 20 KHz, dimana Hz adalah satuan frekuensi yang aritnya banyaknya getaran per detik (cps/cycle per second). Perlengkapan produksi suara pada manusia terdapa pada gambar 1 yang secara garis besar terdiri dari jalur suara (vocal track) dan jalur hidung (nasal track). Jalur suara dimulai dari pita suara (vocal cords), celah suara (glottis) dan berakhir pada bibir. Jalur hidung dimulai dari bagian belakang langit-langit (velum) dan berakhir apda cuping Hidung (nostrils).

Proses menghasilkan suara, dimulai dari udara masuk ke paru-paru melalui pernafasan, kemudian melalui trakea, udara masuk ke batang tenggorok dimana dalam batang tengorok ini terdapat pita suara. Pita suara ini kemudian bergetar dengn frekuensi tertenu karena adanya aliran udara tersebut sehingga dihasilkan suara. Suara yang dihasilkan ini berbeda-beda bergantung pada poisis lidah, bibir, mulut, langit-langit pada saat itu.



Gambar 1. Unsur pembentuk suara

Suara yang dihasilkan terbagi menjadi tiga bagian yaitu voiced sound, unvoiced sound, serta plosive sound. Voiced sound terjadi jika pita suara bergetar dengan frekuensi 50Hz sampai 250Hz. Contoh voiced sound adalah bunyi pada kata “ah”, “oh” unvoiced sopund terjadi jika pita suara tidak bergetar sama sekali. Contoh unvoice sound ialah bunyi pada kata “shh”. Sedangkan plosive sound terjadi jika pita suara tertutup sesaat dan kemudian tiba-tiba membuka. Contohnya plosive sound ialah bunyi “Beh” pada kata benar “pahpah” pada kata pasar.

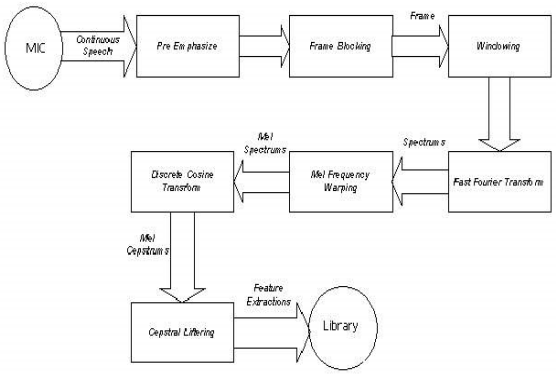
3. Metoda MFCC

MFCC (Mel Frequency Cepstrum Coefficients), merupakan salah satu metode yang banyak digunakan dalam bidang speech technology baik speaker recognition maupun speech recognition. Metode ini digunakan untuk melakukan feature extraction, sebuah proeses yang mengkonversikan signal suara menjadi beberapa parameter. Keunggulan dari metode ini adalah :

a. Mampu untuk menangkap karakteristik suara yang sangat penting bagi pengenalan suara. Atau dengan kata lain mampu menangkap informasi-informasi penting yang terkandung dalam signal suara.
b. Menghasilkan data seminimal mungkin, tanpa menghilangkan informasi-informasi penting yang ada.
c. Mengadaptasi organ pendengaran manusia dalam melakukan persepesi terhadap signal suara.

Perhitungan yang dilakukan dalam MFCC[7] menggunakan dasar dari perhitungan short-term analysis. Hal ini dilakukan mengingat signal suara yang bersifat quasi stationeary. Pengujuian yang dilakukan untuk periode waktu yang cukup pendek (sekotar 10 sampai 30 milidetik) akan menunjukkan karakteristik signal suara yang stationary. Tetapi bila dilakukan dalam periode waktu yang lebih panjang karakteristik signal suara akan terus berubah sesuai dengan kata yang diucapkan.

MFCC feature extraction sebenarnrnya merupakan adaptasi dari sistem pendengaran manusia dimana signal suara akan difilter secara linier untuk frekuensi rendah (dibawah 1000Hz) dan secara logitmik untuk frekuensi tinggi (diatas 1000 Hz). Berikut ini adalah blok diagram untuk MFCC.



Gambar 2. Blok diagram MFCC

4. Konversi Analog Menjadi Digital

Signal-signal yang natural pada umumnya, seperti signal suara merupakan signal continue dimana memiliki nilai yang tidak terbatas. Sedangkan pada komputer, semua signal yang dapat dipreoses oleh komputer hanyalah signal discrete atau sering dikenal dengan istilah digital signal. Agar signal natural dapat diproses oleh komputer, harus dapat mengubah data signal continue menjadi discrete. Hal itu dapat dilakukan melalui 3 proses, diantaranya adalah proses sampling data, proses kuantiti, dan preose pengkodean.

Proses sampling adalah suatu proses untuk mengambil data signal contiue untuk setiap periode tertentu. Dalam melakukan proses sampling data, berlaku aturan Nquist, yaitu bahwa frekuensi sampling (sampling rate) minimal harus dua kali lebih tinggi dari frekuensi maksimum yang ada akan di sampling. Jika signal sampling kurang dari dua kali frekuensi maksimum signal yang akan di sampling maka akan timbul efek aliasing. Aliasing adalah suatu efek dimana signal yang dihasilkan memiliki frekuensi yang berbeda dengan signal aslinya.

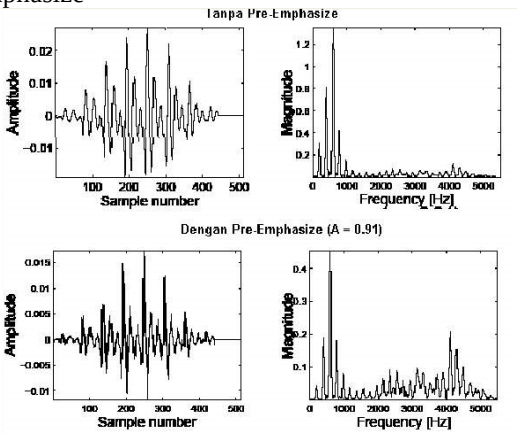
Proses kuantisasi adalah proses untuk membulatkan nilai data ke dalam bilangan-bilangan tertentu yang telah ditentukan terlebih dahulu. Semakin banyak level yang dipakai maka semakin akurat pula data signal yang simpan tetapi akan menghasilkan ukuran data yang besar dan proses yang lama. Proses pengkodean adalah proses pemberian kode untuk tiap-tiap data signal yang telah terkuantisasi berdasarkan label yang ditempati. Berikut adalah gambaran contoh proses pembentukan signal digital .

5. Pre-emphasize Filtering

Pre-emphasize merupakan salah satu jenis filter yang sering digunakan sebelum sebuah signal diproses lebih lanjut. Filter ini mempertahankan frekuensi-frekuensi tinggi pada sebuah spektrum, yang umumnya ter-eliminisasi pada saat proses produksi suara, tujuan dari pre-emhasize filter ini adalah :

a. Mengurangi noise ratio pada signal, sehingga dapat meingkatkan kualitas signal.
b. Menyeimbangkan spektrum dari voiced sound.

Pada saat memproduksi voiced sound, glotis manusia menghasilkan sekitar -12dB octave slope. Namun ketika energi akustik tersebut dikeluarkan melalui bibir, terjadi peningkatan sebesar +6dB. Sehingga signal yang terekam oleh microphone adalah sekitar -6dB octave slope, tanpa pre-emphasize



Gambar 3. Contoh dari pre-emphasize pada sebuah frame

Perhatikan perbedaan pada frekuensi domain akibat diimplementasikannya pre-emphasize filter. Pada gambar 3 tampak bahwa distribusi energi pada setiap frekuensi menjadi lebih seimbang setelah diimplementasikan pre-emphasize filter. Bentuk yang paling umum digunakan dalam pre-emphasize filter adalah sebagai berikut :

$$H(z) = 1 - \propto z^{-1}$$

Dimana $0.9 \leq \alpha \leq 1.0$, dan $\alpha \epsilon$ R. formula diatas dapat diimplmementasikan sebagai first order differentiator, sebagai berikut :

$$y[n] = s[n] - a\, s[n-1] \qquad 3.2$$

Y[n] = Signal hasil pre-emphasize filter
S[n] = Signal sebelum pre-emphasize filter

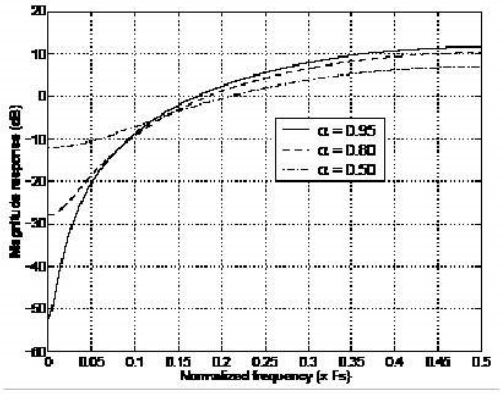
Pada umumnya nilai α yang paling sering digunakan adalah antara 0.9 sampai 1.0. Respon frekwensi dari filter tersebut adalah :

$$H(e^{jw}) = 1 - ae^{-jw} \qquad 3.3$$
$$= 1 - \alpha(\cos w - j \sin w)$$

Sehingga, squared magnitude response dapat dihitung dengan persamaan berikut ini :

$$|H(e^{jw})|^2 = (1 - a\cos w)^2 + a^2 \sin^2 w \qquad 3.4$$
$$= 1 - 2a\cos w + a^2 \cos^2 w + a^2 \sin^2 w$$
$$= 1 - 2a\ \cos w + a^2(\cos^2 w + \sin^2 w)$$
$$= 1 - 2a\cos w + a^2$$

Magnitude response (dB scale) untuk nilai α yang berda dapat dilihat pada gambar 4



Gambar 4. Magnitude response dari pre-emphasize filter untuk nilai α yang berbeda

5.1. Frame Blocking

Karena signal suara terus mengalami perubahan akibat adanya pergeseran artikulasi dari organ produksi vokal, signal harus diproses secara short segments (short frame). Panjang frame yang biasanya digunakan untuk pemrosesan signal adalah antara 10-30 milidetik. Panjang frame yang digunakan, sangat mempengaruhi keberhasilan dalam analisa spektral. Di satu sisi, ukuran dari frame harus sepanjang mungkin untuk dapat menunjukkan resolusi frekuensi yang baik. Tetapi di lain sisi, ukuran frame juga harus cukup pendek untuk dapat menunjukkan resolusi waktu yang baik.

Proses framing ini dilakukan terus sampai seluruh signal dapat terproses. Selain itu, proses ini umumnya dilakukan secara overlapping untuk setiap frame-nya. Panjang daerah overlap yang umum digunakan adalah kurang lebih 30% sampai 50% dari panjang frame.

5.2. Windowing

Proses framing dapat menyebabkan terjadinya kebocoran spektral (spectral leakage) atau aliasing. Efek ini dapat terjadi karena rendahnya jumlah sampling rate, ataupun karena proses frame blocking dimana menyebabkan signal menjadi discontinue. Untuk mengurangi kemungkinan terjadinya kebocoran spektral, maka hsil dari proses framing harus melewati proses windowing. Ada banyak fungsi window, w(n), speerti yang ditunjukkan pada Tabel 2.1. sebuah fungsi window yang baik harus menyempit pada bagian main lobe, dan melebar pada bagian side lobe-nya. Rumus 2.5. menunjukkan reprensetnsai fungsi window terhadap signal suara yang diinputkan.

$$x(n) = \mathrm{x}_1(n)w(n) \qquad (2.5)$$
$$n = 0, 1, \dots, \qquad N = 1$$

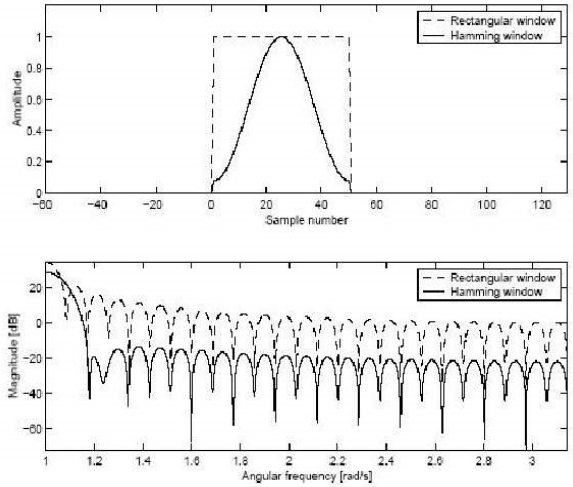
x(n)= Nilai sampel signal
x_i= Nilai sampel dari Frame signal ke i
w(n) = Fungsi window
N=Frame size, merupakan kelipatan 2

Setiap fungsi windows mempunyai karakteristik masing-masing diantara berbagai fungsi window tersebut. Blackman window menghasilkan sidelobe level yang paling tinggi (kurang lebih -58dB), tetapi fungsi ini juga menghasilkan noise paling besar (kurang lebih 1.73 BINS). Oleh karena itu fungsi ini jarang sekali digunakan baik untuk speaker recoginition, maupun *speech recognition*.

Fungsi windows rectangle adalah fungsi window yang paling mudah untuk diaplikasikan. Fungsi ini menghasilkan noise yang paling rendah yaitu sekitar 100 BUIINS. Tetapi sayangnya fungsi ini memberikan sidewloba level yang paling rendah. Sidelobe level yang rendah tersebut menyebabkan besarnya kebocoran spektral yang terjadi dalam proses feature extraction.

Fungsi window yang paling sering digunakan dalam aplikasi speaker recognition adalah hamming windows. Fungsi window ini menghasilkan sidelobe level yang tidak terlalu tinggi (kurang lebih -43db), selain itu noise yang dihasilkan pun tidak terlalu besar (kurang lebih 1.36 BINS).

Gambar 5 menunujukkan bentuk k gelombang dalam time domain dan magnitude dari hamming windows dan retangular windows

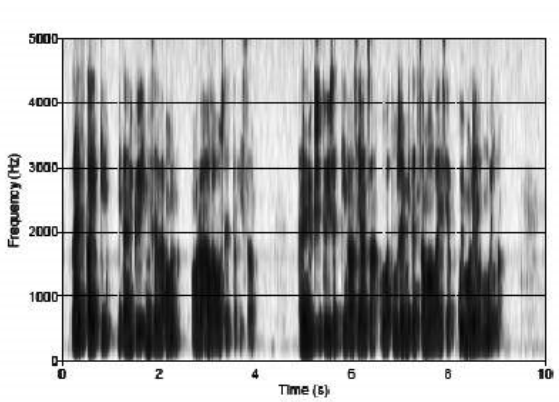


Gambar 5. bentuk gelombang dari hamming dan rectangular windows serta dengan magnitude hasil dari proses FFT.

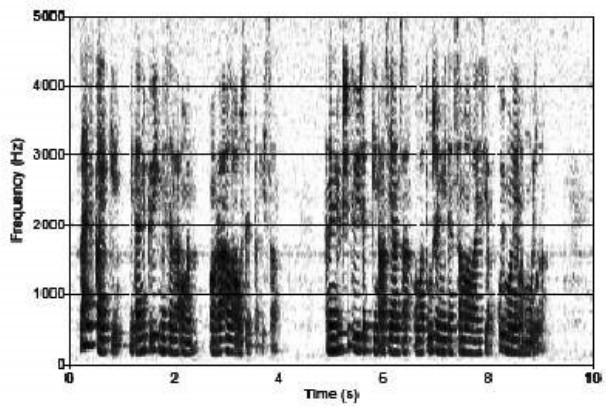
Dari contoh gambar diatas dapat diketahui bahwa kebocoran spektral lebih sedikit terjadi pada hamming windows dari pada rectanguler windows.

5.3. Analisis fourer

Analisis fourier adalah sebuah metode yang memungkinkan untuk melakukan analisa terhadap spectral properties dari signal yang diinputkan. Representasi dari spectral properties sering disebut sebagai spectrogram. Dalam spectrogram terdapat hubungan yang sangat erat antara waktu dan frekuensi. Hubungan antara frekuensi dan waktu adalah hubungan berbanding terbalik. Bila resoulsi waktu yang ditinggikan, makan resolusi frekuensi yang dihasilkan akan semkain rendah. Kondisi seperti ini akan menghasilkan Narrowband spectrogram. Sedangkan wideband spectgrogram adalah kebalikan dari narrowband spectrograma.

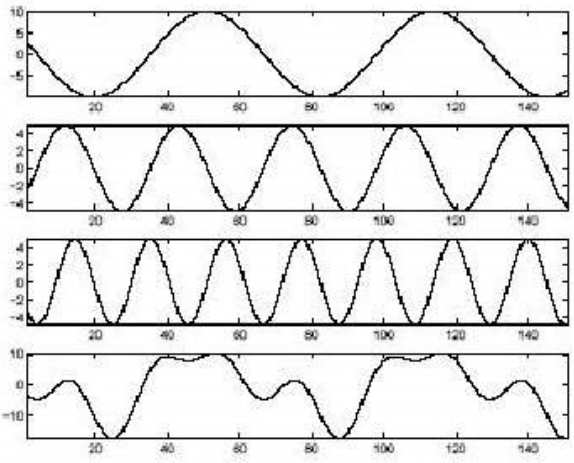


Gambar 6. Contoh dari Wideband Spectogram



Gambar 7. Contoh dari Narrowband spectogram

Inti dari transformasi fourer adalah menguraikan signal ke dalam komponen-komponen bentuk sinus yang berbeda-beda frekuensinya. Gambar 8 menunjukkan tiga gelombang sinus dan superposisinya. Signal semula yang periodik dapat diuraikan menjadi bebarapa komponen bentuk sinus dengan frekuensi yang berbeda. Jika signal semula tidak periodik maka transofrmasi fourer-nya merupakan fungsi frekuensi yang continue, artinya merupakan penjumlahan bentuk sinus dari segala frekuensi. Jadi dapat disimpulkan bahwa tranformasi fourier merpukan representasi frekuensi domian dari suatau signal. Representasi ini mengandung informasi yagn tepat sama dengan kandungan informasi dari signal semula.



Gambar 8. Tiga gelombang sinusoidal dan superposisinya

5.4. Discrete Fourier Transform (DTF)

DFT merupakan perluasan dari transformasi fourier yang belaku untuk signal signal diskrit dengan panjang yang terhingga. Semua signal periodik terbentuk dari gabungan singnal-signal sinusoidal yang menjadi satu yang dalam perumusannya dapat ditulis menjadi :

$$S[k] = \sum_{N=0}^{N-1} s[n] e^{-j2\pi nk/N}, 0 \leq k \leq N-1 \quad (2.6)$$

N = Jumlah sampel yang akan di proses (*N*єN)
S(n) = Nilai sampel signal

k = variabel frewensi discrete, diaman akan bernilai (*k*=*N*/2, Kє*N*)

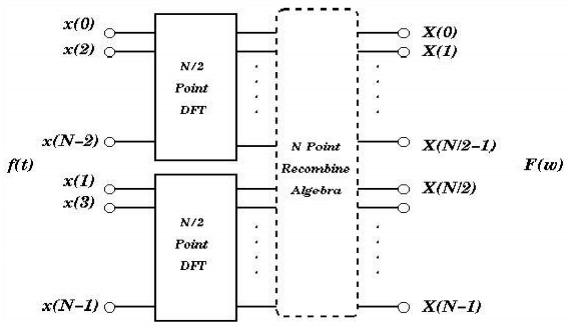
Dengan rumus diatas, suatu signal suara dalam domain waktu dapat dicari frekuensi pembentuknya. Hal inilah tujuan dari penggunaan analisa Fourier pada data suara, yaitu untuk mengubah data dari domain waktu menjadi data spektrum di domain frekuensi. Untuk pemrosesan signal suara, hal ini sangatlah menguntungkan karena data pada domain frekuensi dapat dipreose dengan lebih mudah dibandingkan data pada domain waktu, karena pada domain frekuensi keras lemahnya suara tidak segera berpengaruh.

Untuk mendapatkan spektrum dari sebuah signal dengan DFT diperlukan N buah sample data berurutan pada domain waktu, yaitu data x[m] sampai dengan x[m+N-1]. Data tersebut dimasukkan dalam fungsi DFT maka akan menghasilkan N buah data. Namun karena hasil dari DFT adalah simteris, maka hanya N/2 data yang diambil sebagai spektrum.

5.6. Fast Fourer Transform (FFT)

Perhitungan DFT secara langsung dalam komputerisasi dapat menyebabkan proses perhitungan yang sangat lama. Hal ini disebabkan karena DFT, dibutuhkan N^2 perkalian bilangan kompoleks. Karena itu dibutuhkan car alain untuk menghitung DFT dengan cepat. Hal itu dapat dilakukan dengan menggunakan algoritma Fas Fourier Transform (CFFT) dimana FFT menghilangkan proses perhitungan yang kembar dalam DFT. Algoritma CFFT hanya membutuhkan N $\log_2$ N perkalian kompleks. Berikut ini menunjukkan perbandingan kecapepatan antara FFT Ddan DFT.

Jumlah sample signal yang akan diinputkan ke dalam algoritma ini harus merupakan kelipatan dua (2^M) algoritma Fas Fourer Transform dimulai dengan membagi signal menjadi dua bagian, dimaan bagian pertama berisi nilai signal suara pda indeks waktu genap, dan sebagian yang lain berisi nilai signal suara pada indeks waktu ganjil. Visualisasi dari proeses ini dapat dilihat pada gambar 9. setelah itu, akan dikaukan analisis fourier (recombine algebra) untuk setiap bagiannya. Proses pembagian signal suara tersebut dilakukan sampai didapatkan dua seri nilai signal suara

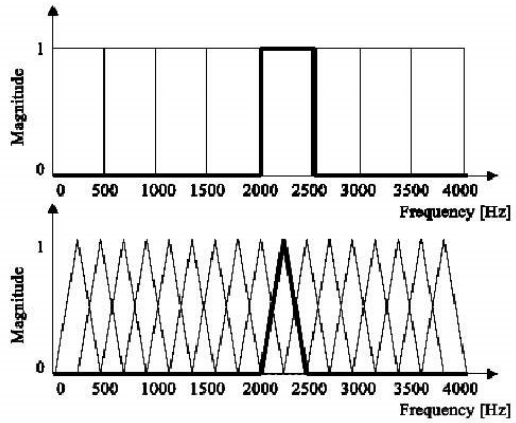


Gambar 9 Pembagian signal suara menadi dua kelompok

Algoritma recombine (DFT) melakuka N perkalian kompoleks, dan dengan metode pembagian seperti ini, maka terdapat $\log_2(N)$ langkah perkaliah kompleks. Hal ini berarti jumlah perkalian komoleks bekurang dari N^2 (pada DFT) menjadi N $\log_2(N)$. Hasil dari proses FFT ini adalah simetris antara index 0 – (N/2 -1) dan (N/2) – (N-1). Oleh karena itu, umum,nya hanya blok pertama saja yang aaaaaaaaakan digunakan dalam proses-proses selanjutnya (lihat halaman)

6. Mel Frequency Warping

Mel Frequency Warping umumnya dilakukan dengan menggunakan Filterbank. Filterbank adalah salah satu bentuk filter yang dilakukan dengan tujuan untuk mengetahui ukuran energi dari frequency band tertentu dalam signal suara. Filter bank dapat diterapkan baik pada domain waktu maupun domain frekuensi, tetapi untuk keperluan MFCC, filter harus diterapkan dalam domain frekuensi. Gambar 10 menunjukkan dua jenis filterbank magnitude. Dalam kedua kasus pada gambar 10 filter yang dilakukan adalah secara linier terhadap frekuensi 0-45 kHz.



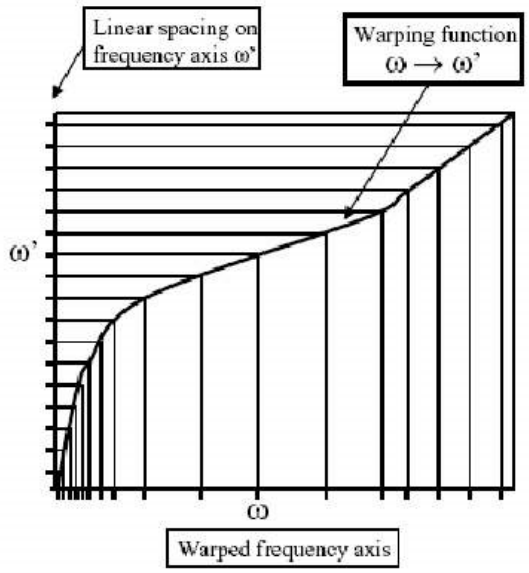
Gambar 10. Magnitude dari rectangular dan triangular filterbank

Filterbank menggunakan representasi konvolusi dalam melakukan filter terhadap signal. Konvolusi dapat dilakukan dengan melakaukan multiplikasi antara spektrum singal dengan koefisien filterbank. Berikut ini adalah rumus yang diguanakn dalam perhitungan filterbanks.

$$Y[i] = \sum_{j=1}^{N} S[j] H_i[j]$$ 3.7

N = Jumlah magnitude spectrum ($N \in N$)
S[j] = Magnitude spectrum pada frekwensi j
$H_i[j]$ = Koefisien filterbank pada frekwensi j($1 \leq i \leq M$)
M = Jumlah channel dalam filterbank

Persepsi manusia terhadap frekuensi dari signal suara tidak mengikuti liear scale. Frekuensi yang sebenarnya (dalam Hz) dalam sebuah signal akan diukur manusia secara subyektif dengan menggunakan mel scale. Mel frequency scale adalah linear frekunsi sscale pada frekuensi dibawah 1000 Hz, dan merupakan logarithmic scale pada frekuensi diatas 1000Hz



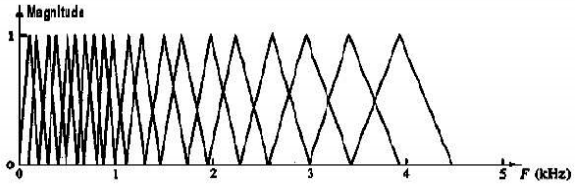
Gambar 11. Prinsip Frekuensi Warping

Berikut ini adalah formula untk menghitung mel scale

$$mel(f) = 2595 \times \log_{10}(1 + f/700)$$ 3.8

Mel(f) = Fungsi Mel Scale
f = Frekwensi

Gambar 12. menunjukkan trianular filterbank dengan menggunakan mel scale. Bila diperhatikan lebih jauh, tampa bahwa untuk frekwensi 1 kHz kebawah poleh filternya terdistribusikan secara linear, da diatas 1 kHz akan terdistribusi secara logarithmic



Gambar 12 Triangular filterbank dengan mel scale

Dalam aplikasi speaker recognition dan speech recognition, jumlah channel filterbank yang digunakan akan sangat berpengaruh terhadap keberhasilan aplikasi tersebut. Semakian besar jumlah channel yang digunakan akan semakin meningkatkan keberhasilan aplikasi, tetapi akan

membutuhkan waktu lebih besar untuk melakukan proses tersebut, begitu pula sebaliknya.

7. Discrete Cosine Transform (DCT)

DCT merupakan langkah terkahir dari proses utama MFCC feature extraction. Konsep dasar dari DCT adalah mendekorelasikan mel spectrum sehingga menghasilkan representasi yang baik dari properti spektral okal. Pada dasarnya konsep dari DCT sama dengan inverse fourier transform. Namun hasil dari DCT mendekati PCA (Principle Component Analysis). PCA adalah metode statistik klasik yang diguankan secara luas dalam analisa data dan dan kompresi. Hal inilah yang menyebabkan seringkali DCT mengantikan inverse fourier transofm dalam poses MFCC Feature Extraction. Berikut ini adalah formula yang digunakan untk menghitung DCT.

$$r_n = \sum_{k=1}^{k} (\log S_k) \cos\left[n\left(k-\frac{1}{2}\right)\frac{\pi}{K}\right] \quad 2.9$$

Sk= Keluaran dari proses filterbank pada index k
K= Jumlah koefisien yang diharapkan

Koefisien ke nol dari DCT pada umumnya akan dihilangkan, walaupun sebenarnya mengindikasikan energi dari frame signal tersebut. Hal ini dilakukan karena, berdasarkan penelitian-penelitian yang pernah dilakukan, koefisien ke nol ini tidak reliable terhadap speaker recognition. Tetapi koefisien ke nol dari DCT sangatlah berguna dalam menentukan end point dari suatu suku kata maupun kata. Hal ini disebabkan karena pada end point dari suatu suku kata maupun kata mempunyai energi yang lebih rendah daripada point-point.

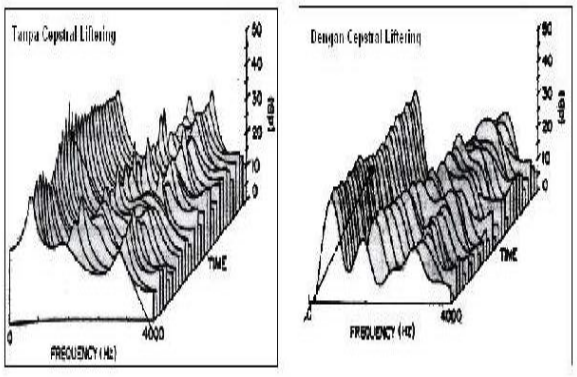
7.1. Cepstral Liftering

Cepstral, hasil dari proses utama MFCC feature extraction, memiliki beberapa kelemahan. Low-order dari cepstral coefficients sangat sensisitif terhadap spectral slope, sedangkan bagian hig-order-nya sangat sensitif terhadap noise. Oleh karen itu, maka cepstral liftering menjadi salah satu standar teknik yang diterapkan untuk meminimalisasi sensitifitas tersebut. Cepstral liftering dapat dilakukan dengan mengimplementasikan fungsi windows terhadap cepstral feature

$$w[n] = \left\{1 + \frac{L}{sin}\, sin\left(\frac{n\pi}{L}\right)\right\} \quad \begin{matrix} n=1,2,\ldots L \\ elsewhere \end{matrix} \quad 2.10$$

L = Jumlah cepstral coefficients
n = index dari cepstral coefficient

Cepstral liftering menghaluskan spektrum hasil dari main processor sehingga dapat digunakan lebih baik untuk pattern mathing. Gambar berikut ini menunjukkan perbadingan spektrum dengan dan tanpa cepstral lifteringt.



Gambar 13. Perbandingan spectrum dengan dan tanpa cepstral liftering

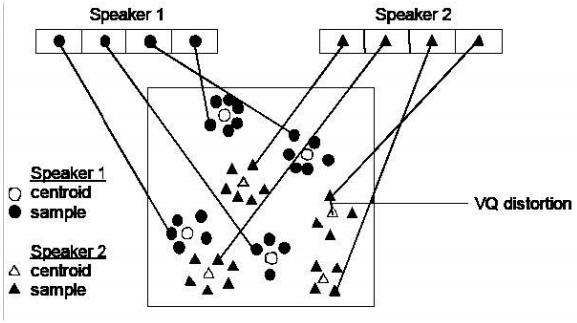
Berdasarkan gambar 13, dapat dilihat bahwa pola spektrum setelah dilakukan cepstral liftering lebih halus daripada spektrum yang tidak melalui tahap cepstral liftering. Dalam beberapa jrnal dikatakan bahwa cepstral liftering dapat meningkatkan akurasi dari aplikasi pengenalan suara, baik speaker recognition, maupun speec recognition

7.2. K-Means Clustring

Clustering merupakan faktor yang paling fundamental dalam *pattern recognition*. Masalah utama dari clustering adalah mendapatkan beberapa nilai fektor pusat yang dapat mewakili keseleruhan vektor dari hasil feature extration. K-means clustering adalah salah satu metode yang digunakan untuk mempartisi vektor hasil feature extraction ke dalam k vektor pusat.

K-Means Clustering [5] adalah proses memetakan vektor-vektor yang berada pada lingkup wilayah yang luas besar menjadi sejumlah tertentu (k) vektor. Wilayah yang terwakili oleh vektor pusat hasil dari proses kuantitasasi disebut sebagai cluster. Sebuah vektor pusat hasil dari proses kuantitasasi dikenal sebagi codewords. Sedangkan kumpulan dari vektor pusat dikenal sebagai codebooks. Gambar berikut ini

menunjukkan ilustrasi dari formasi hasil K-Means clustering.



Keuntungan dari implementasikannya K-Means Clustering dalam merepresentasikan speech spectral ventor adalah :

a. Mengurangi storage memory yang digunakan untuk analisis informasi spektral.
b. Mengurangi perhhitungan yang digunakan untuk mennnentukan mkemiripan dari ventor spektral.

Kelemahan dari penggunaan K-Means Clustering codebooks dalam merepresentasikan speech spectral vectors adalah :

a. Tiimbulnya spektral distori, hal ini terjadi karena vektor yang dianalisa bukanlah vektor asli, tetapi sudah mengalim proses kuantisasi.
b. Storage yang digunakan untk menyimpan codebooks vektor sering kali menjadi masalah . untuk jumlah codebooks yang besar, membutuhkan storage yang cukup besar juga.

7.3. Euclidean Distance

Euclidean Distance[1] adalah sebuah metode yang digunakan untuk mengukur jarak (distance). Euclidean Distance sebenarnya merupakan generalisasi dari teorema phytagoras. Berikut ini adalah contoh perhitungan dengan menggunakan Euclidean Distance. Jika terdapat dua buah titik pada sebuah bidang dua dimensi (R^2), u=(x_1,y_1) dan v=(x_2,y_2), maka untuk mengukur jarak dari kedua buah titik tersebut dapat digunakan persamaan phytagoras

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

3.11X1, x2= Koordinat sumbu x dari sebuah titik.
Y1,y2 = koordinat sumb y dari sebuah titik

Jari tersebut menyebabkan sebuah metric pada R^2, yang disebug sebagai Euclidean metric pada R^2. Bila terdapat dua buah vektor dengan n dimensi, a=(a_1,a_2,….,a_n) dan b=(b_1,b_2,…..,b_n) maka formula phytagoras 3.11, dapt digeneralsiasikan ke dalam n dimensi (R^n) menjadi

$$d = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_n - b_n)^2} \qquad 3.12$$

Perhatikan kemiripan dari dua buah formula di atas, formula 2.11 dan 2.12. Formula Euclidean Distance pada R^n, dikenal sebagai Euclidean Space. Berikut ini adalah bentuk umum dari Euclidean Distance

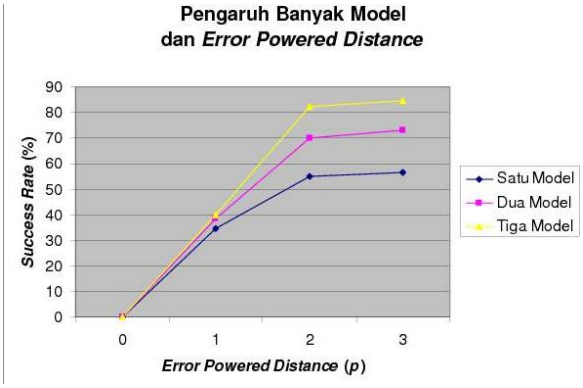
$$d = \left(\sum_{n-1}^{N} (X_n - Y_n)^P\right)^{1/p} \qquad 3.13$$

N= Jumlah dimensi vektor (*N*ϵN)
x, *y* = vektor
P = norm (*P* ϵ R)

Nilai p yang paling sering digunakan adalah *p*=1 dan *p*=2, atau yang sering disebut 1-norm distance dan 2-norm distance

8. Pengujian

Pengujian pertama dilakukan untuk mengetahui pengaruh banyaknya model dan error powered distance terhadap tingkat kesuksesan program. Pengujian akan dilakukan terhadap 25 orang, dengan komposisi 15 orang pria, dan 10 orang wanita. Jumlah kata yang diujikan adalah lima kata, dimana untuk masing-masing kata dilakukan dua kali perekeman. Kata-kata yang digunakan dalam pengujian ini adalah: informatika, pekanbaru, medan, padang, test. Sehingga terdapat 250 kata yang diujikant terhadap aplikasi ini



Gambar 14. Pengaruh Banyak Model dan Error Powered Distance

9. Kesimpulan

Berdasarkan pengujian yang telah dilakukan maka dapat disimpulkan bahwa:

1. Semakin banyak pola kata-kata yang di training-kan terhadap perangkat lunak yang dibuat, akan semakin meningkatkan kemampuan perangkat lunak dalam mengenali pembicara.
2. Jumlah dimensi vektor hasil MFCC feature extraction juga memiliki peran peting dalam meningkatkan persentase keberhasilan dalam mengenali pembicara.
3. Penentuan nilai parameter-parameter yang digunakan dalam MFCC feature extraction maupun K-Means clustering sangat berpengaruh terhadap tingkat keberhasilan yang dapat dicapai oleh aplikasi.
4. Semakin tinggi tingkat keberhasilan yang diharapkan, maka waktu proses yang dibutuhkan semakin lama.
5. Berdasarkan hasil pengujian yang dilakukan terhadap 101 orang, dimana masing-masing mengucapkan 32 kata yagn telah ditentukan sebelumnya (lihat lampiran 1), range presentase keberhasilan yang dapat dicapai oleh aplikasi ini adalah 60% - 90%, dan rata-rata tingkat keberhasilan dari keseluruhan pengujian adalah 83.5%

Daftar Pustaka

[1] Antonio M. Peinado, Jos´ eC.Segura, 2006, *Speech Recognition Over Digital Channels Robustness and Standards*, , John Wiley & Sons, Ltd, West Sussex

[2] David Damm, Harald Grohganz, Frank Kurth, Sebastian Ewert, and Michael Clausen, 2011, *SyncTS: Automatic synchronization of speech and text documents*, AES 42ND INTERNATIONAL CONFERENCE, Ilmenau, Germany, 2011 July 22–24, page 1 – 10

[3] Helenca Duxans i Barrobes, 2006, *Voice Conversion applied to Text-to-Speech systems*, Universitat Politecnica de Catalunya

[4] Ricardo Ribeiro, David Martins de Matos, 2008, *Mixed-Source Multi-Document Speech-to-Text Summarization*, Coling 2008: Proceedings of the workshop on Multi-source Multilingual Information Extraction and Summarization, Manchester, August 2008, pages 33–40

[5] Sanjiv K. Bhatia, 2004, *Adaptive K-Means Clustering*, American Association for Artificial Intelligence (www.aaai.org). page 74-79

[6] Yusuke Nakashima, Zhipeng Zhang, Nobuhiko Naka, *Efficient Speech-recognition Error Correction for More Usable Speech-to-text Input*, NTT DOCOMO Technical Journal Vol. 11 No. 2, page 30-38

[7] Wu Chou, Biing-Hwang Juang, 2003, *Pattern Recognition in Speech and Language Processing*, CRC Press LLC, New Jersey