IDENTIFIKASI CAMPURAN NADA PADA SUARA PIANO MENGGUNAKAN CODEBOOK

Ade Fruandta dan Agus Buono
Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor
Jl. Meranti Wing 20 Level 5 Darmaga Bogor 16680
Telp. (0251) 8625584, Faks. (0251) 8625584
E-mail: ade_g64070074@yahoo.com

ABSTRAK

Pada paper ini disajikan teknik pengenalan nada, baik sebagai nada tunggal maupun nada campuran dengan menggunakan mel-frequency cepstrum coefficients (MFCC) sebagai ekstraksi ciri dan pemodelan codebook untuk pengenal pola. Suara yang dpergunakan adalah suara piano dan dikenali 12 nada tunggal dan 66 nada campuran yang disample dengan 11 kHz pada durasi 1 detik. Pembuatan codebook dilakukan secara bertahap, yaitu codebook jumlah campuran dan codebook nada (tunggal dan campuran) yang dikembangkan dengan menggunakan teknik pengklasteran. Hasil percobaan menunjukkan bahwa jumlah codeword yang optimum adalah 20 dengan lebar frame 256 data, dengan akurasi 98.2%. Namun demikian, ada beberapa nada yang sulit dikenali, yaitu CC#, CD, CF, dan A#B yang memiliki akurasi masing-masing di bawah 50%. Untuk nada CC# lebih sering dikenali nada C, untuk nada CD lebih sering lebih sering dikenali dengan nada C#, untuk nada CF lebih sering dikenali dengan nada C# dan CF#, sedangkan untuk nada A#B lebih sering dikenali dengan nada A#. Kesalahan dalam pengenalan ini dikarenakan nada-nada tersebut berada dalam klaster yang sama sehingga jarak nada-nada tersebut saling berdekatan.

Kata Kunci: MFCC, Codebook, Codework, klastering, nada

1. PENDAHULAN

1.1 Latar Belakang

Seiring dengan berkembangnya keinginan manusia terhadap kemampuan komputer untuk membantu pekerjaannya, maka riset pemrosesan suara senantiasa makin meningkat. Hal ini salah satunya disebabkan banyaknya bidang terapan, mulai dari mesin pendikte, mesin konversi sinyal ke teks, mesin penjawab otomatis, hingga pengembangan interface manusia-komputer berbasis suara. Namun demikian, hasil penelitian yang telah ada hingga sekarang belum memberikan hasil yang memuaskan, (Buono, 2009). Oleh karena itu, riset bidang ini masih terus dan layak dilakukan.

Dalam bidang musik, telah dilakukan penelitian mengenai pengenalan *cord* dengan menggunakan teknik *Mel-Frequency Cepstral Coefficients* (MFCC) sebagai ekstraksi ciri dan *codebook* sebagai pengenal pola dengan akurasi mencapai 97%, (Wisnudisastra dan Buono, 2009). Dalam musik, hal yang tidak kalah pentingnya adalah mengetahui nada-nada pembentuk *cord* tersebut. Hal ini adalah hal yang biasa dilakukan oleh seorang *perfect pitch*. Oleh karena itu, pada penelitian ini akan dilakukan pemodelan suara untuk pengenalan nada-nada penyusun cord dengan teknik MFCC dan *codebook* yang dimodifikasi sebagai pengenal pola nada.

1.2 Tujuan

Penelitian ini bertujuan untuk meneliti kinerja dari algoritma *codebook* dalam mengidentifikasi campuran nada pada suara piano.

1.3 Ruang Lingkup

Adapun ruang lingkup dari penelitian ini antara lain:

a. Campuran nada yang akan dikenali hanya campuran nada pada satu octave dan maksimal dua campuran nada.
b. Suara yang dikenali hanya dimainkan dengan cara ditekan secara serentak.
c. Suara yang dikenali hanya suara piano pada *keyboard* Yamaha PSR 3000.

1.4 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan informasi mengenai kinerja *codebook* dalam mengidentifikasi campuran nada pada sebuah suara piano.

2. TINJAUAN PUSTAKA

2.1 Pemrosesan Sinyal Suara

Sinyal suara merupakan gelombang yang tercipta dari tekanan udara yang berasal dari paru-paru yang berjalan melewati lintasan suara menuju mulut dan rongga hidung (Al-Akaidi, 2007). Pemrosesan suara itu sendiri merupakan teknik mentransformasi sinyal suara menjadi informasi yang berarti sesuai dengan yang diinginkan (Buono, 2009).

Sinyal secara umum dapat dikatagorikan sesuai dengan peubah bebas waktu:

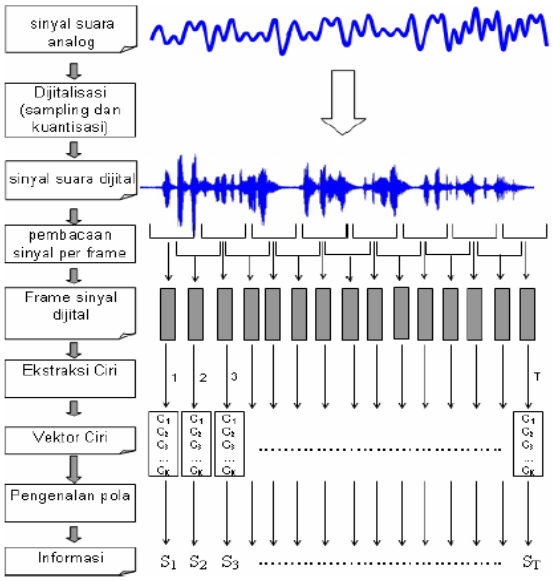
a. Sinyal waktu kontinyu: kuantitas sinyal terdefinisi pada setiap waktu dalam selang

 ISSN: 1907-5022

kontinyu. Sinyal waktu kontinyu disebut juga sinyal analog.

b. Sinyal waktu diskret: kuantitas sinyal terdefinisi pada waktu diskret tertentu, yang dalam hal ini jarak antar waktu tidak harus sama.

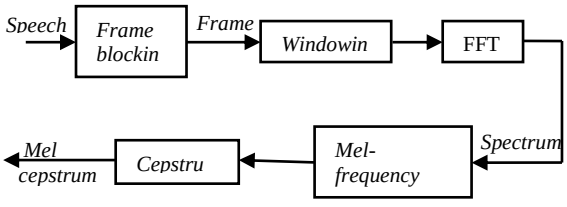
Secara umum proses transformasi tersebut terdiri atas digitalisasi sinyal analog, ekstraksi ciri dan diakhiri dengan pengenalan pola untukk klasifikasi, seperti yang terlihat pada Gambar 1.



Gambar 1**.** Tahapan transformasi sinyal suara menjadi Informasi (Jurafsky *dalam* Buono, 2009)

2.2 Ekstraksi Sinyal Suara

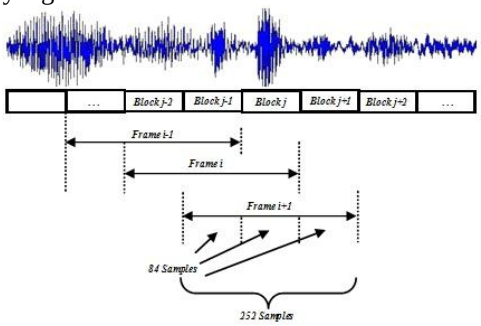
MFCC merupakan salah satu metode ekstraksi ciri dan cara yang paling sering digunakan pada berbagai bidang area pemrosessan suara, karena dianggap cukup baik dalam merepresentasikan ciri sebuah sinyal. Cara kerja MFCC didasarkan pada perbedaan frekuensi yang dapat ditangkap oleh telinga manusia sehingga mampu merepresentasikan ciri sinyal suara sebagaimana manusia merepresentasikannya. Blok diagram proses MFCC dapat dilihat pada Gambar 2 (Do, 1994).



Gambar 2**.** Blok diagram proses MFCC

- *Frame blocking*

Pada proses ini, sinyal suara disegmentasi menjadi beberapa *frame* yang saling tumpang tindih (*overlap*), hal ini dilakukan agar tidak ada sedikitpun sinyal yang hilang (*deletion*). Panjang *frame* biasanya memiliki panjang 10-30 ms atau 256-1024 data. Proses ini akan berlanjut sampai seluruh sinyal sudah masuk ke dalam satu atau lebih *frame* seperti yang diilustrasikan dalam Gambar 3.



Gambar 3**.** Ilustrasi *frame blocking* pada sinyal suara

- *Windowing*

Sinyal analog yang sudah diubah menjadi sinyal digital dibaca *frame* demi *frame* dan pada setiap *frame*-nya dilakukan *windowing* dengan fungsi *window* tertentu. Proses *windowing* bertujuan untuk meminimalisasi ketidakberlanjutan sinyal pada awal dan akhir setiap *frame* (Do, 1994). Dengan pertimbangan kesederhanaan formula dan nilai kinerja *window*, maka penggunaan *window* Hamming cukup beralasan (Buono, 2009).

Jika kita definisikan *window* sebagai $w(n)$, $0 \le n \le N-1$, dimana N adalah jumlah sampel pada setiap *frame*-nya, maka hasil dari *windowing* adalah sinyal:

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

dimana $w(n)$ biasanya menggunakan *window* Hamming yang memiliki bentuk:

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

- *Fast Fourier Transform* (FFT)

FFT adalah algoritma cepat untuk mengimplementasi discrete fourier transform (DFT). FFT ini mengubah masing-masing frame N sampel dari domain waktu menjadi domain frekuensi yang didefinisikan sebagai berikut:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j2\pi kn/N},\ k = 0,1,2,\ldots,N-1$$

hasil rangkaian $\{X_k\}$ direpresentasikan sebagai berikut:

a. Frekuensi positif $0 \le f \le \frac{F_S}{2}$ yang merepresentasikan nilai $0 \le n \le \frac{N}{2} - 1$,

b. Frekuensi negatif $\frac{-F_S}{2} < f < 0$ yang merepresentasikan nilai $\frac{N}{2} + 1 \le n \le N-1$.

Disini, F_S berarti *frequency sampling*. Hasil dari tahapan ini biasanya disebut dengan *spectrum* atau *periodogram*.

- *Mel-Frequency Wrapping*

Persepsi sistem pendengaran manusia terhadap frekuensi sinyal suara tidak dapat diukur dalam skala linear. Untuk setiap nada dengan frekuensi aktual, *f*, diukur dalam Hz, sebuah *subjective pitch* diukur dalam sebuah skala yang disebut 'mel'. Skala *mel-frequency* ialah sebuah frekuensi rendah yang bersifat linear di bawah 1000 Hz dan sebuah frekuensi tinggi yang bersifat logaritmik di atas 1000 Hz. Persamaan berikut menunjukkan hubungan skala mel dengan frekuensi dalam Hz:

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

- *Cepstrum*

Langkah terakhir yaitu mengubah spektrum *log mel* menjadi domain waktu. Hasil ini disebut *mel frequency cepstrum coefficient* (MFCC). *Cepstral* dari *spectrum* suara merepresentasikan sifat-sifat spektral lokal sinyal untuk analisis *frame* yang diketahui. Koefisien *mel spectrum* merupakan sebuah nilai riil sehingga kita dapat mengkonversinya ke dalam dominan waktu menggunakan *Discrete Cosine Transform* (DCT). Selanjutnya kita dapat menghitung MFCC sebagai $\hat{c}_n$, sebagai

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

dimana $\hat{S}_0$, $k = 0, 2, ..., K-1$ dan $n = 0, 1, ..., K-1$.

2.3 *Codebook*

Codebook adalah sekumpulan titik (vektor) yang mewakili distribusi suara dari individu maupun objek tertentu dalam ruang suara. Titik-titik pada *codebook* disebut *codeword*. *Codebook* merupakan cetakan yang dihasilkan suara setelah melalui proses *training*. Dalam pengenalan suara, masing-masing suara yang akan dikenali harus dibuatkan *codebook*-nya.

Codebook dibentuk dengan cara membentuk *cluster* semua vektor ciri yang dijadikan sebagai *training set* dengan menggunakan *clustering algorithm*. Algoritma *clustering* yang akan dipakai adalah algoritma *K-means*. Langkah pertama yang dilakukan oleh algoritma ini adalah menentukan *Kinitial centroid*, di mana *K* adalah parameter spesifik yang ditentukan *user*, yang merupakan jumlah *cluster* yang diinginkan.Setiap titik atau objek kemudian ditempatkan pada *centroid* terdekat, dan kumpulan titik atau objek pada tiap *centroid* disebut *cluster*.*Centroid* pada setiap *cluster* kemudian akan berubah berdasarkan setiap objek yang ada pada *cluster*. Kemudian langkah penempatan objek dan perubahan *centroid* diulangi sampai tidak ada objek yang berpindah *cluster*.

Prinsip dasar dalam penggunaan *codebook* adalah setiap suara yang masuk akan dihitung jaraknya kesetiap *codebook* yang telah dibuat. Kemudian jarak setiap sinyal suara ke *codebook* dihitung sebagai jumlah jarak setiap frame sinyal suara tersebut ke setiap *codeword* yang ada pada *codebook*. Kemudian dipilih *codeword* dengan jarak minimum. Setelah itu setiap sinyal suara yang masuk akan diidentifikasi berdasarkan jumlah dari jarak minimum tersebut. Perhitungan jarak dilakukan dengan menggunakan jarak *euclid* yang didefinisikan sebagai berikut:

$$d_{euclidian}(x, y) = \sum_{i=1}^{D} (x_i - y_i)^2$$

dimana *x* dan *y* adalah vektor yang ada sepanjang D.

Jika dalam sinyal suara input *O* terdapat T *frame* dan $codeword_k$ merupakan masing-masing *codeword* yang ada pada *codebook* maka jarak sinyal input dengan *codebook* dapat dirumuskan:

$$jarak(O, codebook) = \sum_{t=1}^{T} \min[\, d(O_b codeword_k)]$$

3. METODE PENELITIAN

3.1 Kerangka Pemikiran

Penelitian ini dikembangkan dengan metode yang terdiri dari beberapa tahap yaitu: (1) pengambilan data, (2) *preprocessing*, (3) pemodeln *codebook*, (4) evaluasi. Alur metode ini dapat dilihat pada Gambar 4.

3.2 Pengambilan Data

Suara yang akan digunakan dalam penelitian ini adalah suara grand piano yang terdapat di *keyboard* Yamaha PSR 3000. Nada yang diambil sebanyak 12 nada tunggal yang terdiri dari C, C#, D, D#, E, F, F#, G, G#, A, A#, dan B yang masing-masing akan diulang sebanyak 15 kali. Nada campuran diambil sebanyak 66 nada campuran yang masing-masing akan diulang sebanyak 15 kali.

3.3 *Preprocessing*

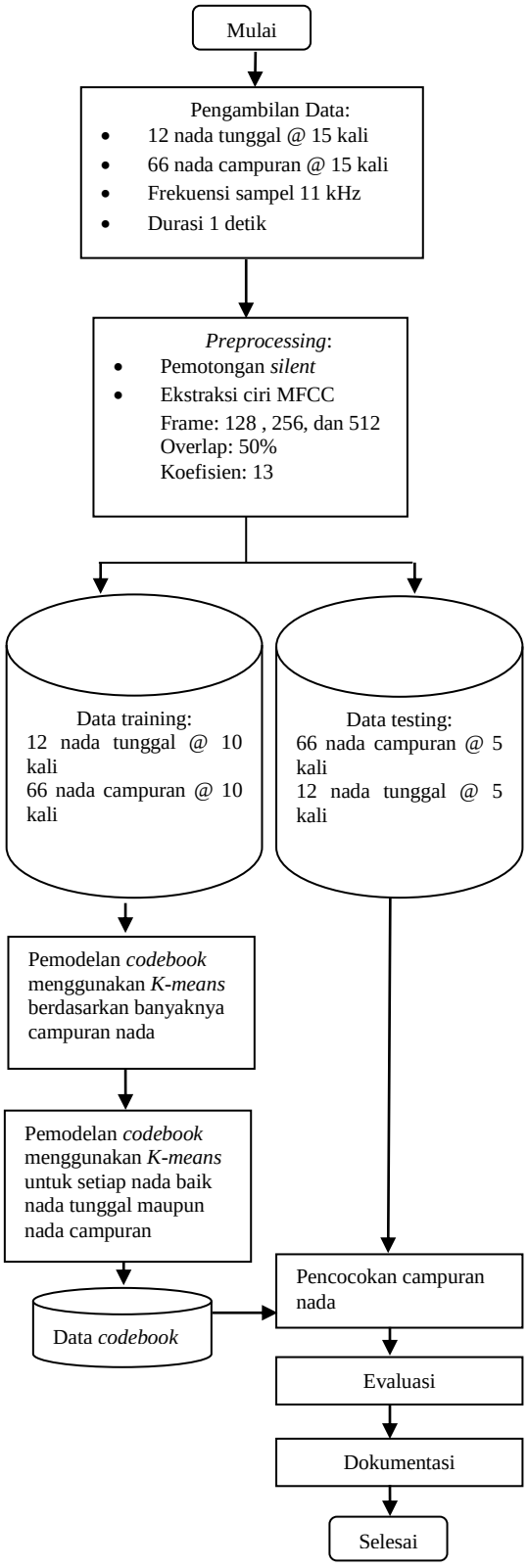
Pada tahap ini, suara yang telah direkam akan dilakukan proses pemotongan *silent*. Pemotongan *silent* ini diharapkan dapat menfokuskan sinyal yang akan diteliti.

Setelah melalui tahap pemotongan *silent* sinyal akan diekstraksi ciri pada setiap nada dengan menggunakan MFCC. Pada penelitian ini akan diteliti dengan lebar frame 128, 256, dan 512, overlap sebesar 50%, dan jumlah *cepstral coefficient* setiap frame sebanyak 13 koefisien.

3.4 Pemodelan *Codebook*

Pada tahap ini akan dibuat *codebook* dari 120 suara nada tunggal dan 660 suara nada campuran yang telah melalui *preprocessing*. Tiap suara tersebut akan di *clustering* dengan menggunakan *K-means* sehingga dihasilkan *cluster-cluster* yang berisi vektor-vektor nada yang berdekatan. Setiap *cluster* tersebut kemudian dibuatkan *codebook*-nya. Masing-masing *codebook* yang dibuat memiliki jumlah *k cluster* 5, 10, 15 , dan 20. Setiap *codebook* yang telah dibuat akan di *clustering* kembali berdasarkan banyaknya campuran nada dengan

menggunakan *K-means* sehingga dihasilkan *cluster-cluster* yang berisi vektor-vektor yang mencirikan banyaknya campuran nada.



Gambar 4. Diagram alur proses identifikasi campuran nada

3.5 Evaluasi

Evaluasi sistem ini melihat akurasi identifikasi campuran nada pada suara piano. Data yang digunakan adalah 1490 suara yang terdiri dari 286 nada campuran yang masing-masing lima suara dan 12 nada tunggal yang masing-masing lima suara. Untuk perhitungan tingkat akurasi identifikasi campuran nada dilakukan dengan membandingkan jumlah *output* yang benar diidentifikasi oleh sistem dengan jumlah seluruh data yang diuji. Persentase tingkat akurasi dihitung dengan fungsi berikut:

$$akurasi = \frac{\sum nada\ yang\ benar}{\sum nada\ yang\ diuji} \times 100\%$$

4. HASIL DAN PEMBAHASAN

4.1 *Preprocessing*

Dari data yang telah direkam yaitu 780 data *training* dan 390 data *testing*, terlebih dahulu dilakukan pemotongan *silent* yang akan diteruskan dengan ekstraksi ciri menggunakan MFCC. Dalam pemakaiannya terdapat lima parameter yang harus digunakan yaitu suara, *sampling rate*, *frame*, *overlap*, dan *cepstral coefficient*. Pemilihan nilai untuk *sampling rate*, *overlap*, dan *cepstral coefficient* berturut-turut adalah 11000 Hz, 50%, dan 13. Untuk nilai *frame* akan diuji dengan nilai 128, 256, dan 512. Proses ekstraksi ini akan dilakukan terhadap semua data. MFCC mengubah data menjadi sebuah matriks yang berisikan vektor-vektor yang menunjukkan ciri *spectral* dari data tersebut.

4.2 Pemodelan *Codebook*

Pada proses pembuatan *codebook*, data yang digunakan adalah data *training* yang berupa vektor ciri dari suara piano yang telah direkam dan melewati *praprocessing*. Terdapat dua jenis model *codebook* yang akan dimodelkan yaitu *codebook* tiap nada baik tunggal maupun campuran dan *codebook* berdasarkan banyaknya campuran nada.

a. Pemodelan *codebook* tiap nada

Terdapat 12 jenis nada tunggal dan 66 jenis nada campuran yang masing-masing berjumlah 10 suara yang akan dimodelkan *codebook*-nya. Tiap jenis nada akan melalui proses *clustering* dengan *K-means*. Nilai *K* yang akan diuji adalah 5, 10, 15, dan 20 untuk tiap *frame* yang diuji yang masing-masing *frame* adalah 128, 256, dan 512. Setelah melalui proses *clustering* akan didapatkan vektor-vektor *centroid* yang mencirikan masing-masing jenis nada.

b. Pemodelan *codebook* berdasarkan banyaknya campuran nada

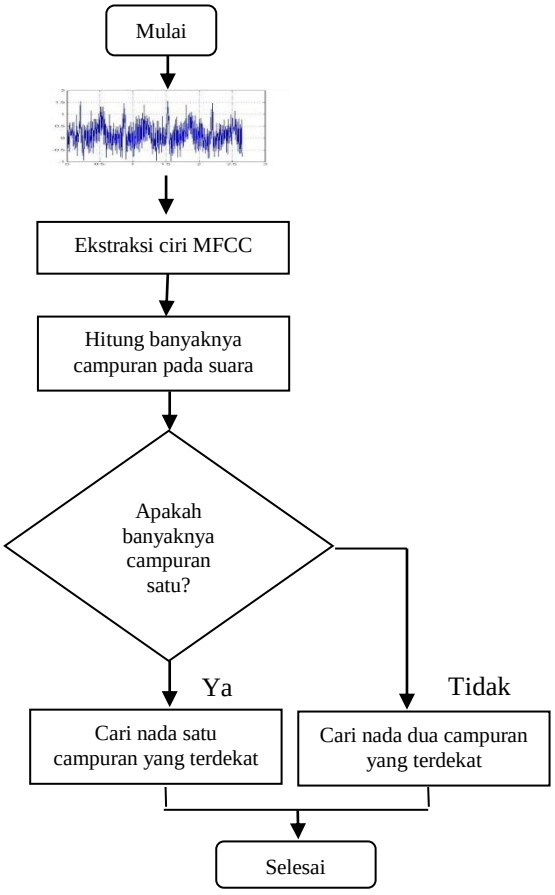
Setelah mendapatkan model *codebook* untuk tiap nada, maka hasil tersebut akan diguanakan dalam pemodelan *codebook* berdasarkan banyaknya campuran nada. Model *codebook* sebelumnya akan dipisahkan berdasarkan banyaknya campuran nada, dalam hal ini satu campuran dan dua campuran.

 ISSN: 1907-5022

Setelah dipisahkan maka akan di proses dengan *clustering* menggunakan *K-means*. Nilai *K* yang akan diuji adalah 5, 10, 15, dan 20 untuk tiap *frame* yang diuji yang masing-masing *frame* adalah 128, 256, dan 512. Setelah melalui proses *clustering* akan didapatkan vektor-vektor *centroid* yang mencirikan masing-masing banyaknya campuran yang ada.

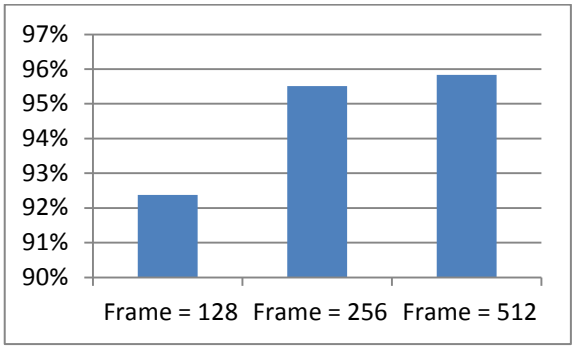
4.3 Pengujian

Pengujian akan dilakukan dengan data *testing* yang telah direkam dengan *sampling rate* 11000 Hz. Banyaknya data *testing* ini adalah 12 nada tunggal dan 66 nada campuran yang masing-masing nada memiliki lima suara. Data *testing* ini akan melewati *preprocessing* untuk pemebersihan data dan pencirian data. Setelah melalui *preprocessing* masing-masing suara akan dikenali berdasarkan model *codebook* yang telah dibuat sebelumnya dengan mencari jarak yang terdekat dengan model. Untuk tahap awal suara yang akan dikenali berdasarkan banyaknya campuran pada suara tersebut. Setelah diketahui banyaknya campuran pada suara tersebut maka akan dikenali jenis nadanya berdasarkan banyaknya campuran nada. Alur pengenalan campuran nada dapat dilihat pada Gambar 5.



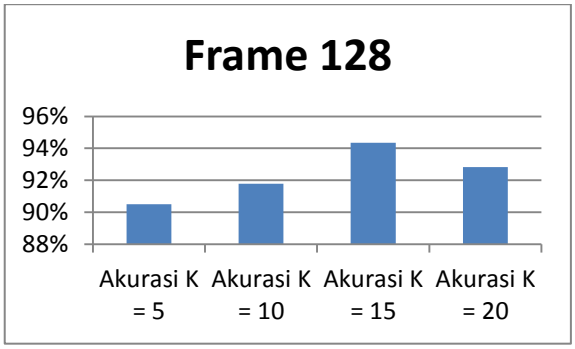
Gambar 5. Alur pengenalan campuran nada

Pengujian dilakukan dengan *frame* yang berbeda-beda. Nilai *frame* yang diuji adalah 128, 256, dan 512. Pada *frame* 128 didapatkan akurasi sebesar 92%, saat *frame* 256 didapatkan akurasi 96% dan saat *frame* 512 didapatkan akurasi 96%. Grafik akurasi untuk setiap *frame*-nya dapat dilihat pada Gambar 6.



Gambar 6. Akurasi untuk setiap nilai *frame*

Untuk setiap *frame* memiliki nilai akurasi berdasarkan nilai *K* yang berbeda-beda. Nilai *K* yang digunakan adalah 5, 10, 15, dan 20. Untuk hasil akurasi *frame* 128 dapat dilihat pada Gambar 7. Dari Gambar 7 dapat dilihat bahwa akurasi yang paling besar saat *K* bernilai 15 yang memiliki akurasi 94%.

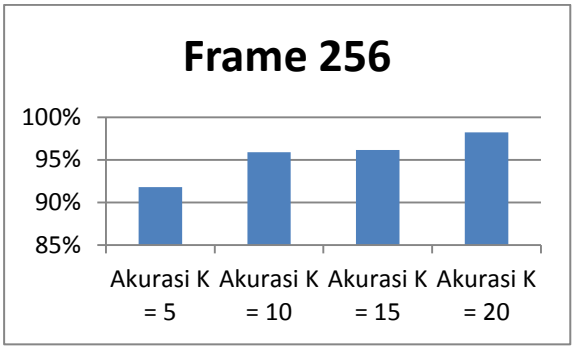


Gambar 7. Akurasi untuk setiap nilai *K* pada *frame* 128

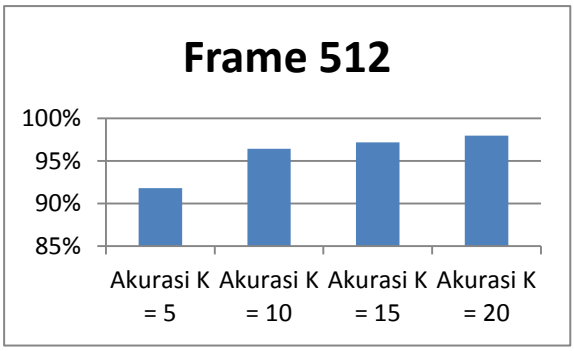
Untuk hasil akurasi *frame* 256 dapat dilihat pada Gambar 8. Dari Gambar 8 dapat dilihat bahwa akurasi paling besar saat *K* bernilai 20 yang memiliki akurasi 98%.

Untuk hasil akurasi *frame* 512 dapat dilihat pada Gambar 9. Dari Gambar 9 dapat dilihat bahwa akurasi paling besar saat *K* bernilai 20 yang memiliki akurasi 98%.

 ISSN: 1907-5022



Gambar 8. Akurasi untuk setiap nilai *K* pada *frame* 256



Gambar 9. Akurasi untuk setiap nilai *K* pada *frame* 512

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini telah berhasil dalam mengimplementasikan metode *codebook* dalam mengenali banyaknya campuran dari sebuah suara. Total nilai rataan tertinggi dihasilkan saat *frame* 256 dan 512 yang masing-masing 96%. Pada *frame* 256 akurasi tertinggi didapatkan pada nilai *K* = 20 sebesar 98,2051%. Sedangkan untuk *frame* 512 akurasi tertinggi didapatkan pada nilai *K* = 20 sebesar 97,9487%.

Namun dari nilai-nilai akurasi tersebut terdapat nada-nada yang sulit dikenali atau memiliki nilai akurasi yang rendah yaitu CC#, CD, CF, dan A#B yang memiliki akurasi masing-masing di bawah 50%. Untuk nada CC# lebih sering dikenali nada C, untuk nada CD lebih sering lebih sering dikenali dengan nada C#, untuk nada CF lebih sering dikenali dengan nada C# dan CF#, sedangkan untuk nada A#B lebih sering dikenali dengan nada A#. Kesalahan dalam pengenalan ini dikaranakan nada-nada tersebut berada dalam *cluster* yang sama sehingga jarak nada-nada tersebut berdekatan yang menyebabkan sulit untuk dikenali.

5.2 Saran

Penelitian ini masih sangat sederhana sehingga memungkinkan untuk dikembangkan lebih lanjut. Saran-saran yang dapat diberikan untuk pengembangan lebih lanjut adalah:

a. Pada penelitian ini nada yang dimodelkan hanya berada pada satu octave sehingga jika dimasukkan dengan nada yang sama namun dengan octave yang berbeda maka akan salah dikenali. Sehingga disarankan untuk memodelkan semua octave pada piano untuk dimodelkan.
b. Banyaknya campuran pada penelitian ini hanya dua campuran, sehingga disarankan untuk memodelkan semua kemungkinan campuran yang ada.
c. Nada campuran yang dapat dikenali hanya berada pada satu octave, jika terdapat campuran yang masing-masing berbeda octave maka tidak dapat dikenali. Sehingga disarankan untuk memodelkan campuran nada yang masing-masing berbeda octave.

PUSTAKA

Al-Kaidi M. (2007). *Fractal Speech Processing*. Cambridge University Press.

Buono, A. (2009). *Representasi Nilai HOS dan Model MFCC sebagai Ekstraksi Ciri pada Sistem Identifikasi Pembicara di Lingkungan Ber-noise Menggunakan HMM*. [Disertasi]. Depok: Program Studi Ilmu Komputer, Universitas Indonesia.

Do MN. (1994). DSP Mini-Project: An Automatic Speaker Recognition System.

Jurafsky D, Martin JH. (2007). *Speech and Language Processing An Introduction to Natural Language Processing, Computational Linguistic, and Speech Recognition*. New Jersey: Prentice Hall.

Wisnudisastra, E dan A. Buono. (2009). *Pengenalan Cord pada Alat Musik Gitar Menggunakan Codebook dengan Teknik Ekstraksi Ciri MFCC*. Jurnal Ilmiah Ilmu Komputer, Departemen Ilmu Komputer IPB.