

# Rancang Bangun Aplikasi MusicMoo dengan Metode MIR (Music Information Retrieval) pada Modul Mood, Genre Recognition, dan Tempo Estimation

Johanes Andre Ridoean, Riyanarto Sarno, Dwi Sunaryo

Departemen Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

*e-mail*:riyanarto@if.its.ac.id, dwi@if.its.ac.id

**Abstrak**—Saat ini, metode pemanggilan kembali informasi suatu musik atau yang sering disebut *Music Information Retrieval* (MIR) telah banyak diterapkan. Contohnya pada suatu aplikasi Shazam ataupun SounHound. Kedua aplikasi ini hanya menangani sebatas suatu lagu berjudul apakah ketika diperdengarkan. Untuk itu, tujuan penelitian ini adalah pengembangan lebih lanjut MIR yang lebih spesifik lagi, yaitu melakukan pemanggilan informasi lagu yang terkait kembali beserta detail lagu di antaranya adalah *mood*, *genre*, dan tempo lagu. Penelitian ini memakai ekstraksi fitur berbasis MPEG-7 yang oleh *library* Java bernama *MPEG7AudioEnc*. Hasil ekstraksi fitur ini berupa metadata dalam bentuk angka digital yang merepresentasikan karakteristik suatu sinyal pada tiap fiturnya. Setelah fitur didapatkan, tahap berikutnya adalah melakukan pengambilan suatu fitur sesuai dengan masing-masing modul dengan metode *Xquery* yang diimplementasikan oleh *library* Java bernama *BaseX*. Fitur yang diambil dipakai untuk proses pengolahan dengan *Discrete Wavelet Transform* (DWT) beserta level dekomposisi terbaik oleh *library* Python bernama *Pywt*. Setelah fitur-fitur diproses, maka dilakukan penggabungan fitur pada suatu *list* beserta penyamaan panjang fitur untuk proses klasifikasi. Tahap terakhir adalah melakukan klasifikasi dengan menggunakan *Support Vector Machine* (SVM). Terdiri dari 2 tahap yaitu tahap *training* dan *prediksi*. Hasil akurasi keberhasilan pada penelitian ini untuk modul *mood* 75%, *genre* 87,5% dan tempo 80%.

**Kata Kunci**—Analisis Audio, MIR, MPEG-7, SVM.

## I. PENDAHULUAN

DEWASA ini, industri media sudah berkembang sangat pesat, khususnya pada lagu. Lagu dapat ditemukan oleh masyarakat secara mudah. Mulai dari radio, *Compact Disc* (CD), internet, dan sumber-sumber lainnya.

Sebagai hasil dari ledakan terbaru dalam media tersebut, muncul juga suatu kebutuhan pokok untuk kalangan masyarakat agar bisa mengetahui informasi lagu yang lebih lengkap pada suatu lagu. Menjawab kebutuhan masyarakat, ditemukanlah metode *Music Information Retrieval* atau yang disingkat menjadi MIR. Menurut definisi Downie, MIR adalah usaha penelitian yang berusaha untuk mengembangkan inovatif musik berbasis konten, mencari skema, *interface* baru untuk membuat toko besar di dunia musik [1]. Definisi secara umum, MIR adalah metode pemanggilan informasi suatu musik agar dapat memberikan informasi lagu yang kompleks.

Pada penelitian ini, eksperimen yang dilakukan terkait dengan analisa dan implementasi MIR pada modul *mood*, *genre recognition*, dan tempo estimation. Tujuan dari penelitian ini adalah memperkaya detail informasi suatu lagu.

Konsep yang digunakan pada penelitian ini adalah menggunakan ekstraksi fitur berbasis MPEG-7 yang sudah menjadi standar dalam konten multimedia berdasarkan ISO/IEC 15938 [2][3]. Hasil ekstraksi fitur ini berupa metadata dalam format XML yang di dalamnya berisi angka-angka digital. Angka-angka digital ini merepresentasikan karakteristik suatu sinyal. Terdapat sejumlah 17 fitur yang dihasilkan dari ekstraksi fitur berbasis MPEG-7. Fitur-fitur yang didapatkan ini memungkinkan untuk melakukan identifikasi kemiripan suatu lagu termasuk *mood*, *genre*, dan *tempo* berdasarkan kemiripan karakteristik sinyal yang dimiliki.

Diskusi pada jurnal ini dibagai dalam struktur sebagai berikut: Bab II membahas materi dan metode yang digunakan dalam percobaan ini. Bab III membahas hasil dan diskusi percobaan yang dilakukan. Terakhir pada Bab IV membahas kesimpulan dari hasil yang didapatkan dalam penelitian ini.

## II. MATERI DAN METODE

Pada modul *mood*, audio fitur yang dipakai untuk proses adalah *Audio Power* dan *Audio Harmonicity*. Kedua fitur ini dipilih dan dikombinasikan karena untuk *mood* suatu lagu, dipengaruhi oleh suatu daya / naik turunnya amplitudo serta keselarasan harmonis nada [4]. Maka dipilihlah fitur *Audio Power* dan *Harmonicity*. Sedangkan Pada modul *genre* dan tempo, audio fitur yang dipakai untuk proses adalah *Audio Spectrum Centroid*, *Audio Spectrum Spread*, dan *Audio Spectrum Flatness* [5]. Fitur ini dipakai karena suatu lagu yang mempengaruhi *genre* dan tempo adalah informasi kejernihan suara (ASC), penyimpangan spektrum dari sinyal asli (ASS), dan kerataan properti suatu kekuatan spektrum (ASF). Maka untuk *data train* adalah gabungan ketiga sinyal yaitu ASC, ASS, dan ASF.

### A. Audio Power [2]

*Audio Power* (AP) adalah fitur yang menggambarkan temporal daya sesaat dari sinyal audio. Koefisiennya dari kuadrat rata-rata nilai gelombang sn dalam *non-overlapping frame*. Tujuannya adalah untuk membandingkan suatu label

sinyal. Persamaan (1) adalah cara mendapatkan nilai AP.

$$AP(l) = \frac{1}{N_{hop}} \sum_{n=0}^{N_{hop}-1} |s(n + lN_{hop})|^2 \quad (0 \leq l \leq L-1) \quad (1)$$

di mana  $L$  adalah total jumlah frame waktu,  $s(n)$  merupakan rata-rata *square waveform*,  $l$  merupakan indeks frame dan  $N_{hop}$  adalah sampel antara *successive non-overlapping*.

Tabel 1. Penentuan Decompose Level Wavelet

Decomposition Level (L)	Frequency range (Hz)
1	256-512
2	128-256
3	64-128
4	32-64
5	16-32
6	8-16
7	4-8
8	2-4
9	1-2
10	0.5-1
11	0.25-0.5
12	0.125-0.25
13	0.0625-0.125

**B. Audio Power [2]**

*Audio Power* (AP) adalah fitur yang menggambarkan temporal daya sesaat dari sinyal audio. Koefisiennya dari kuadrat rata-rata nilai gelombang  $s(n)$  dalam *non-overlapping frame*. Tujuannya adalah untuk membandingkan suatu label sinyal. Persamaan (1) adalah cara mendapatkan nilai AP.

$$AP(l) = \frac{1}{N_{hop}} \sum_{n=0}^{N_{hop}-1} |s(n + lN_{hop})|^2 \quad (0 \leq l \leq L-1) \quad (1)$$

di mana  $L$  adalah total jumlah frame waktu,  $s(n)$  merupakan rata-rata *square waveform*,  $l$  merupakan indeks frame dan  $N_{hop}$  adalah sampel antara *successive non-overlapping*.

**C. Audio Harmonicity [2]**

*Audio Harmonicity* adalah fitur yang mendeskripsikan 2 properti sinyal harmonik dari spektrum. Properti yang pertama adalah *harmonic ratio* yaitu rasio daya harmonik dari total daya sedangkan properti yang kedua adalah *upper limit harmonicity* yaitu frekuensi spektrum yang tidak dapat dianggap harmonis. Tujuannya untuk membedakan suara harmonik (alat musik contohnya) dan suara non-harmonik (*noise*, pidato tidak jelas, dsb).

**D. Audio Spectrum Centroid [2]**

*Audio Spectrum Centroid* (ASC) merepresentasikan sebagai karakteristik dari sebuah spektrum. ASC bisa juga menunjukkan pusat dari sebuah spektrum. Secara perseptual,

ASC memiliki hubungan kuat antara kejernihan suara. Persamaan (2) adalah cara mendapatkan nilai ASC.

$$ASC = \frac{\sum_{k=0}^{N_{FT}-1} \log_2 \left( \frac{f(k)}{1000} \right) P'(k)}{\sum_{k=0}^{N_{FT}-1} P'(k)} \quad (2)$$

di mana setiap frekuensi  $f(k')$  adalah frekuensi yang sesuai dengan indeks  $k'$ ,  $P'(k')$  adalah *power* spektrum yang

Tabel 2. Perbedaan Antara Emosi dan Mood

Emosi	Mood
Durasi singkat.	Durasi lama.
Spesifik mengarah kepada suatu hal.	Biasanya lebih umum.
Biasanya disertai dengan beragam ekspresi wajah.	Biasanya tidak disertai dengan ekspresi wajah.

Tabel 3. Pembagian Estimasi Tempo

Satuan BPM	Estimasi Tempo
<100	Lambat
100-135	Sedang
>135	Cepat

Tabel 4. Panjang Minimal Tiap Fitur

Fitur	Panjang minimal
Audio Power	4.498
Audio Harmonicity	4.493
Audio Spectrum Centroid	51
Audio Spectrum Spread	51
Audio Spectrum Flatness	107.904

diekstrak pada *frame* ke- $l$ , dan  $N_{FT}$  adalah ukuran dari *Fast Fourier Transform*.

**E. Audio Spectrum Spread Type [2]**

*Audio Spectrum Spread* didefinisikan sebagai momen sentral kedua spektrum *log*-frekuensi. Fitur ini diekstraksi dengan mengambil *Root Mean Square* (RMS) dari penyimpangan spektrum dari *Audio Spectrum Centroid*. Persamaan (3) adalah cara mendapatkan nilai ASS.

$$ASS = \sqrt{\frac{\sum_{k=0}^{N_{FT}-1} \left[ \log_2 \left( \frac{f(k)}{1000} \right) - ASC \right]^2 P'(k)}{\sum_{k=0}^{N_{FT}-1} P'(k)}} \quad (3)$$

di mana setiap frekuensi  $f(k')$  adalah frekuensi yang sesuai dengan indeks  $k'$ ,  $P'(k')$  adalah *power* spektrum yang diekstrak pada *frame* ke- $l$ .

**F. Audio Spectrum Flatness Type [2]**

*Audio Spectrum Flatness* didefinisikan sebagai cerminan kerataan properti suatu kekuatan spektrum. Persamaan (4) adalah cara mendapatkan nilai ASF.

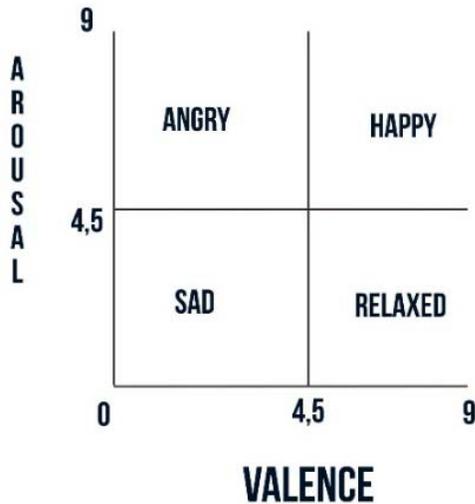
$$ASF = \frac{1}{N_{FT} - \log_2 + 1} \frac{\prod_{k=0}^{N_{FT}-1} B(k)}{\sqrt{\prod_{k=0}^{N_{FT}-1} B(k)}} \quad (1 \leq b \leq B) \quad (4)$$

G. Fast Fourier Transform

Fast Fourier Transform (FFT) adalah metode untuk mengubah sinyal dari *time domain* menjadi *frequency domain* [6]. Tujuannya adalah untuk mencari suatu informasi penting dalam *frequency domain* untuk tahap analisis [7][8].

H. Discrete Wavelet Transform

Discrete Wavelet Transform (DWT) adalah metode wavelet yang digunakan untuk melakukan dekomposisi pada wavelet



Gambar 1. Bagan Russel's Circumplex

sampai level N [6]. Tujuannya adalah untuk mengurangi *noise* pada sinyal dan memperkuat informasi di dalam sinyal tersebut dengan mempertahankan keutuhan informasi data [9] [10].

Ada banyak metode wavelet, namun dalam pengerjaan Tugas Akhir ini, metode wavelet yang digunakan adalah *bior2.8* dengan mengambil nilai *approximation coefficients*.

Untuk menentukan level dekomposisi ini tidak boleh sembarangan karena ketika level dekomposisi tinggi belum tentu yang baik, sebaliknya malah merusak sinyal sehingga menghilangkan informasi yang terkandung sinyal asli. Maka perlu dilakukan pemilihan level dekomposisi wavelet yang terbaik [11].

Langkah pertama adalah melakukan perhitungan pada Rujuk ke (5) yang dilakukan pada Matlab.

$$[max\ value, index\ max] = \max(abs(FFT(S - mean(S)))) \quad (5)$$

Maka akan didapatkan hasil *max value* dan *max index*. Kedua hasil ini digunakan untuk mencari *Fh*. *Fh* adalah *Frequency Range* pada Tabel 1. Untuk mendapatkan *Fh*, lakukan perhitungan pada Rujuk ke (6).

$$Fh = index\ max * Fs/L \quad (6)$$

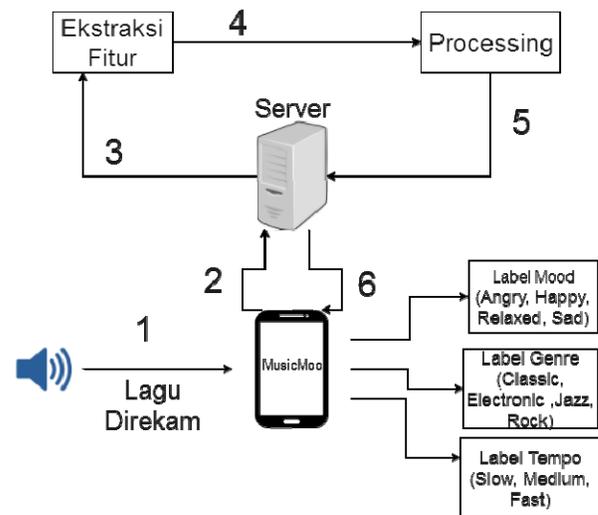
Di mana *Fs* adalah frekuensi sampling yaitu 1024, dan *L* adalah panjang sinyal. Hasil dari *Fh* dapat dilihat dari Tabel 2.1 sesuai dengan *frequency range*. Aturan untuk menentukan tingkat dekomposisi pada Tabel 1 dapat dinyatakan oleh Rujuk ke (7) [6]:

$$\frac{f_s}{2^N + 1} \leq f_{center} \leq \frac{f_s}{2^N} \quad (7)$$

di mana  $f_s$  adalah *sampling frequency*,  $f_{center}$  adalah *dominant frequency*, dan *N* adalah level dekomposisi. Maka didapatkan level *decompose wavelet* terbaik.

I. Xquery

Xquery adalah bahasa untuk meng-query/ pemanggilan data di dalam suatu *database* dalam bentuk file XML [12]. Pada Tugas Akhir ini, Xquery digunakan untuk mengambil suatu fitur pada XML yang dihasilkan lewat ekstraksi fitur MPEG-7. Untuk mengimplementasikan Xquery, menggunakan *library* Java yang bernama *BaseX* [13].



Gambar 2. Deskripsi Umum Sistem

J. Dataset

Dataset adalah kumpulan satu data yang dilakukan untuk uji coba. Pada uji coba yang dilakukan untuk penelitian ini diambil dari *1000 Songs for Emotional Analysis of Music* [14][15]. Dari dataset ini sudah terdapat nilai *valence* dan *arousal* sebagai penentuan *mood*, *genre* sebuah lagu, dan BPM lagu yang pasti dimiliki semua lagu.

K. Mood

Mood adalah keadaan emosional yang bersifat sementara, bisa beberapa menit bahkan beberapa minggu. Mood juga bisa diartikan tanggapan kita terhadap suatu rangsangan yang terjadi. Mood berbeda dengan emosi. Emosi adalah perasaan intens yang diarahkan pada seseorang atau sesuatu. Sedangkan mood adalah perasaan yang tumbuh kurang intens yang dikarenakan kekurangan suatu stimulus. Secara umum, perbedaannya antara mood dan emosi dipetakan menjadi berikut pada Tabel 2 [16].

Pada musik, yang dipakai pada uji coba kali ini adalah yang mood, dikarenakan musik adalah stimulus yang bagus untuk meningkatkan mood seseorang.

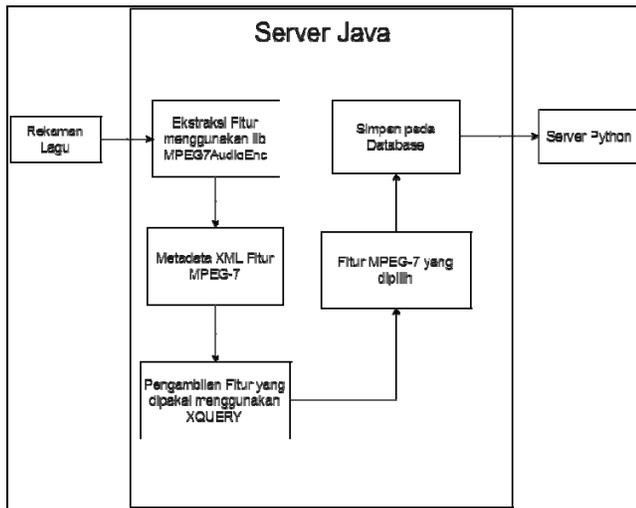
L. Genre

Genre musik adalah pengelompokan musik sesuai dengan kemiripan satu sama lain. Sebuah genre dapat juga didefinisikan oleh teknik musik, gaya, dan konteks musik.

Terdapat banyak macam variasi *genre* pada suatu musik [17]. Dikarenakan sebuah *genre* pada suatu lagu terlalu banyak, maka pada percobaan ini dibatasi hanya sebatas *genre* umum suatu lagu yaitu *classic*, *electronic*, *jazz*, dan *rock* dimana di *dataset* juga sudah tersedia.

#### M. Tempo

Tempo musik adalah ukuran kecepatan dalam birama lagu. Ukuran kecepatan bisa diukur dengan alat bernama *metronome* dan *keyboard*. Secara umum, tempo dibagi



Gambar 3. Tahapan Ekstraksi Fitur Pada Server Java

menjadi 3 yaitu lambat, sedang, dan cepat. Tempo dibagi berdasarkan kecepatan nada [18]. Tabel3 adalah pemetaan pembagian tempo berdasarkan *Beats Per Minutes* (BPM).

Untuk mendapatkan BPM suatu lagu, penelitian ini menggunakan *software* yang bernama MixMeister BPM Analyzer yang dapat diunduh secara gratis [19].

#### M. Russel's Circumplex models

*Russel's Circumplex models* adalah model yang digunakan untuk memetakan emosi dari dua kombinasi linear: *arousal* dan *valence*[20]. Pada uji coba kali ini, akan dipetakan berdasarkan skor *arusal* dan *valence* sesuai dengan Gambar 1.

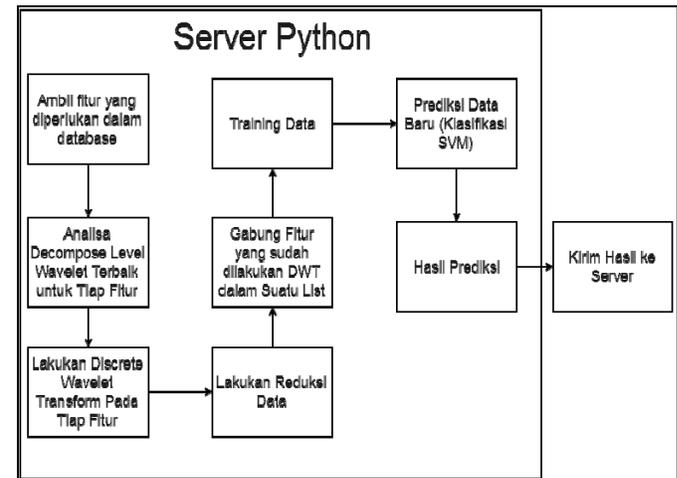
#### N. Support Vector Machine

*Support Vector Machine* (SVM) adalah metode *machine learning* yang bekerja dengan mencari *hyperlane* terbaik (fungsi klasifier) yang memisahkan beberapa label yang berbeda [1]. *Hyperlane* yang optimal dapat ditemukan dengan cara mengukur *margin/distance* antara *hyperlane* dengan data yang paling dekat pada masing-masing labelnya.

SVM telah digunakan dalam berbagai eksperimen seperti klasifikasi identifikasi pembicara, pengenalan obyek, deteksi wajah dan klasifikasi vokal. SVM dapat mengklasifikasikan data multi-dimensi adalah yang pada dasarnya menentukan perbatasan antara dua kelas atau lebih.

Contoh pelatihan *training* data pada SVM menentukan parameter dari keputusan untuk mengklasifikasikan fungsi dari dua atau lebih kelas dan memaksimalkan margin selama fase pembelajaran. Setelah tahap pembelajaran, klasifikasi pola yang tidak diketahui dapat diperkirakan

Pada penelitian ini, dilakukan juga sesi *training* dan sesi *testing* untuk mencoba klasifikasi. Sesi *training* merupakan sesi melakukan pembelajaran terhadap variasi data-data yang dimiliki oleh suatu label sedangkan sesi *testing* merupakan proses uji coba prediksi pada suatu data yang baru. Metode SVM dipilih karena percobaan sebelumnya klasifikasi SVM menggunakan fitur MPEG-7 dan hasilnya baik [21]. Untuk melakukan implementasi SVM ini, digunakan *library sklearn* dari bahasa pemrograman Python menggunakan *default kernel* yaitu *rbf*.



Gambar 4. Tahapan Processing Pada Server Python

### III. HASIL DAN DISKUSI

Aplikasi yang dibuat pada penelitian ini adalah aplikasi perangkat bergerak. Untuk alur jalannya sistem digambarkan pada Gambar 2. Penjelasan keterangan pada penomoran Gambar 2 adalah sebagai berikut:

1. Sebuah lagu direkam melalui aplikasi perangkat bergerak.
2. Lagu rekaman di-*upload* ke server.
3. Lagu rekaman dilakukan ekstraksi fitur dan pengambilan fitur oleh server Java.
4. Fitur yang diambil dilakukan *processing* pada server Python (melakukan DWT, reduksi data dan klasifikasi).
5. Hasil akan dikirim kembali pada server.
6. Server akan menampilkan hasil pada aplikasi perangkat bergerak

#### A. Ekstraksi Fitur

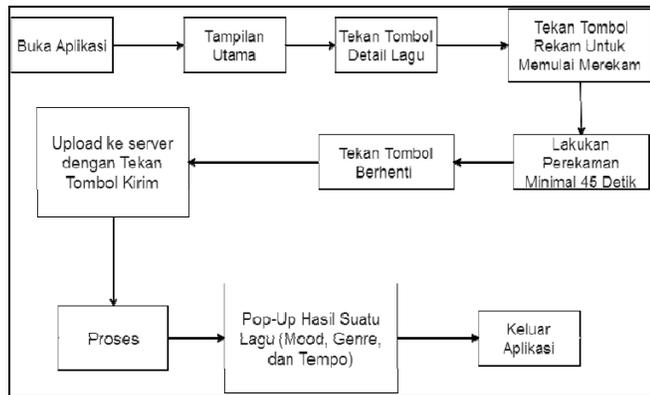
Untuk detail tahapan proses yang terjadi di server Java akan digambarkan pada Gambar 3. Suatu lagu akan dilakukan ekstraksi fitur sehingga menghasilkan metadata XML berisi fitur-fitur MPEG-7. Setelah metadata dihasilkan, maka dilakukan pengambilan fitur oleh *Xquery* untuk mengambil fitur-fitur yang akan dipakai untuk disimpan dalam *database*. Jika sudah maka tugas pada server Java selesai dan dilanjutkan tahap *processing* yang ditangani oleh server Python.

#### B. Processing

Pada tahap ini, fitur yang dipilih akan masuk ke tahap *processing*. Fitur yang dipilih tadi akan dilakukan *Discrete Wavelet Transform* (DWT) dengan menggunakan tipe *wavelet bior 2.8*. Tujuannya adalah untuk menghilangkan *noise* yang terdapat pada sinyal. Implementasi yang dilakukan untuk

melakukan DWT ini menggunakan *library* Python yang bernama *Pywt*. Untuk detail tahapan proses yang terjadi di server Python akan digambarkan pada Gambar 4.

Kemudian dilanjutkan tahapan yaitu reduksi data. Reduksi data yang dimaksud adalah menyamakan panjang sinyal agar menjadi seragam untuk proses klasifikasi. Hal ini dilakukan karena ekstraksi fitur yang dihasilkan mempertimbangkan milidetik juga.



Gambar 5. Alur Penggunaan Aplikasi

Tabel 5. Perincian Data Training

Perincian Data Training		
Modul	Label	Jumlah Data Training
Mood	Angry	65
	Happy	65
	Relaxed	65
	Sad	65
Genre	Classic	65
	Electronic	65
	Jazz	65
	Rock	65
Tempo	Slow	65
	Medium	65
	Fast	65

Kami telah melakukan analisis pada 296 lagu dan mengambil panjang minimal pada setiap fitur yang telah dilakukan DWT. Tabel 3 adalah hasil analisis panjang sinyal minimal untuk tiap fitur akan dipaparkan pada Tabel 4.

Maka panjang sinyal fitur setiap musik yang diambil sebanyak angka-angka tersebut, sisanya bisa diabaikan. Proses ini dijamin tidak akan merusak sinyal karena dari data awal panjang sinyal seragam semua yaitu 45 detik dan hanya membuang milidetik saja.

Setelah proses ini, akan masuk pada tahap berikutnya yaitu menggabungkan fitur dalam suatu *list*. Pada tahap ini, fitur-fitur yang sudah dilakukan DWT akan digabungkan menjadi satu untuk menjadi data yang siap diklasifikasikan.

Untuk modul *mood*, gabungan fitur yang dibentuk adalah *Audio Power* dan *Audio Harmonicity*, Sedangkan untuk modul *genre* dan *tempo*, gabungan fitur yang dibentuk adalah kombinasi dari *Audio Spectrum Centroid*, *Audio Spectrum Spread*, dan *Audio Spectrum Flatness*. Gabungan-gabungan fitur ini sudah dibagi sama panjang fitur per kolom untuk

melakukan klasifikasi, sehingga keaslian informasi sinyal dijamin tidak akan rusak/berbeda.

Terakhir, adalah proses *training data*. Sebelum tahap klasifikasi menggunakan SVM, ada proses yang namanya melatih data. Tujuannya adalah agar mesin dapat mengetahui beragam karakteristik sinyal yang dimaksud sesuai dengan label tertentu.

Tabel 6. Perincian Data Testing

Rincian Jumlah Data Testing			
Modul	Label	Jumlah Data Testing	Total
Mood	Angry	5	35
	Happy	10	
	Relaxed	10	
	Sad	10	
Genre	Classic	6	24
	Electronic	6	
	Jazz	6	
	Rock	6	
Tempo	Slow	10	30
	Medium	10	
	Fast	10	

Tabel 7. Hasil Percobaan Modul Mood

Aktual	Pengujian			
	Angry	Happy	Relaxed	Sad
Angry	4	1	0	0
Happy	0	10	0	0
Relaxed	0	3	2	5
Sad	0	0	0	10

Implementasi SVM dilakukan menggunakan *library* Python yang bernama *sklearn*. Pada uji coba kali ini adalah melakukan *training data* untuk tiap modul (*mood*, *genre*, dan *tempo*) sebanyak 65 data per label. Tabel 5 adalah perincian jumlah lagu yang digunakan untuk *data training* tiap modul.

Setelah mesin mempelajari karakteristik-karakteristik sinyal sesuai yang diberikan, maka prediksi data baru pun dapat dilakukan. Sehingga karakteristik lagu rekaman tadi dapat diprediksi memiliki *mood*, *genre*, ataupun *tempo* apa sesuai dengan karakteristik sinyal yang diberikan. Hasil prediksi akan dibawa kembali ke server untuk ditampilkan pada aplikasi perangkat bergerak. Gambar 5 adalah penjelasan alur penggunaan aplikasi dalam bentuk *workflow*. Alur proses ini membantu memperlihatkan langkah demi langkah yang akan dilakukan pengguna sehingga terlihat jelas antara *input*, *proses*, dan *output*.

### C. Hasil Percobaan

Data yang dilakukan untuk uji coba pada masing-masing modul pada penelitian ini diperlihatkan pada Tabel 6. Akurasi dihitung menggunakan Rujuk ke (8) yang merupakan

perhitungan dengan cara *Positive Predictive Value* (PPV) sebagai berikut.

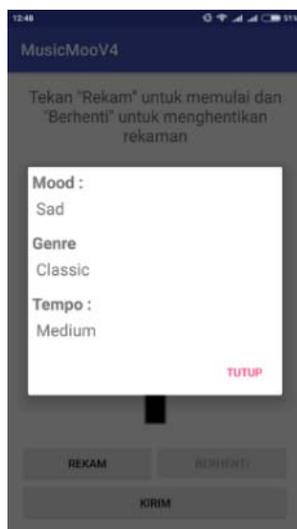
$$PPV = \frac{TP}{TP + FP} \times 100\%$$

Tabel 8. Hasil Percobaan Modul Tempo

Aktual	Pengujian		
	Fast	Medium	Slow
Fast	5	5	0
Medium	0	10	0
Slow	0	1	9

Tabel 9. Hasil Pecobaan Modul Genre

Aktual	Pengujian			
	Classic	Electronic	Jazz	Rock
Classic	6	0	0	0
Electronic	0	6	0	0
Jazz	1	1	4	0
Rock	0	1	0	5



Gambar 6. Contoh Tampilan Output Pada Aplikasi Perangkat Bergerak

Di mana *TP* adalah *True Positif* yaitu jumlah data yang diprediksi benar sesuai dengan labelnya dan *FP* adalah *False Positif* yaitu jumlah data yang diprediksi dikatakan benar namun kenyataannya adalah salah. *TP + FP* bisa juga dibilang jumlah data keseluruhan yang dipakai dalam pengujian.

Secara berurutan, Tabel 7, Tabel 8, dan Tabel 9 adalah hasil uji coba pada modul *mood*, *tempo*, dan *genre*. Dengan menggunakan Rujuk ke (8), maka akurasi yang didapatkan untuk modul *mood* 75%, *tempo* 80%, dan *genre* 87,5 %. Gambar 6 adalah contoh hasil tampilan *output* berupa *pop-up* yang dilihat oleh pengguna pada aplikasi perangkat bergerak. Pada pojok kanan bawah terdapat tulisan *tutup* untuk menutup *pop-up* yang dihasilkan.

#### IV. KESIMPULAN

Pada modul *mood*, audio fitur yang mempengaruhi adalah *Audio Power* dan *Audio Harmonicity*. Pada modul *genre*, audio fitur yang berpengaruh adalah fitur *Audio Spectrum Centroid*, *Audio Spectrum Spread*, dan *Audio Spectrum Flatness*. Terakhir, pada modul *tempo*, audio fitur yang berpengaruh adalah fitur *Audio Spectrum Centroid*, *Audio Spectrum Spread*, dan *Audio Spectrum Flatness*. Fitur yang berpengaruh pada tempo sama seperti dengan *genre*. Tingkat akurasi untuk masing-masing modul sudah baik di mana *mood* 75%, *genre* 87,5%, dan *tempo* 80%.

#### DAFTAR PUSTAKA

- [1] F. Wiering, "Can Humans Benefit from Music Information Retrieval?," in *Proceedings of the 4th International Conference on Adaptive Multimedia Retrieval: User, Context, and Feedback*, Berlin, Heidelberg, 2007, pp. 82–94.
- [2] H.-G. Kim, N. Moreau, and T. Sikora, *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval*. John Wiley & Sons, 2005.
- [3] ISO/IEC (2001), "Information Technology — Multimedia Content Description Interface — Part 4: Audio," *FDIS 15938-42001E*, June.
- [4] Z. W. Ras and A. Wiczorkowska, *Advances in Music Information Retrieval*, 1st ed. Springer Publishing Company, Incorporated, 2010.
- [5] S. Li, H. Li, and L. Ma, "Music Genre Classification Based on MPEG-7 Audio Features," in *Proceedings of the Second International Conference on Internet Multimedia Computing and Service*, New York, NY, USA, 2010, pp. 185–188.
- [6] R. X. Gao and R. Yan, *Wavelets: Theory and Applications for Manufacturing*, 2011 edition. New York; London: Springer, 2010.
- [7] B. T. Nugraha, R. Sarno, D. A. Asfani, T. Igasaki, and M. N. Munawar, "Classification of driver fatigue state based on EEG using Emotiv EPOC+," *J. Theor. Appl. Inf. Technol.*, vol. 86, no. 3, pp. 347–359, Apr. 2016.
- [8] R. Sarno, M. N. Munawar, B. T. Nugraha, R. Sarno, M. N. Munawar, and B. T. Nugraha, "Real-Time Electroencephalography-Based Emotion Recognition System," *Int. Rev. Comput. Softw. IRECOS*, vol. 11, no. 5, pp. 456–465, May 2016.
- [9] R. Sarno, B. T. Nugraha, M. N. Munawar, R. Sarno, B. T. Nugraha, and M. N. Munawar, "Real Time Fatigue-Driver Detection from Electroencephalography Using Emotiv EPOC+," *Int. Rev. Comput. Softw. IRECOS*, vol. 11, no. 3, pp. 214–223, Mar. 2016.
- [10] M. N. Munawar, R. Sarno, D. A. Asfani, T. Igasaki, and B. T. Nugraha, "Significant preprocessing method in EEG-Based emotions classification," *J. Theor. Appl. Inf. Technol.*, vol. 87, no. 2, pp. 176–190, May 2016.
- [11] D. R. Wijaya, R. Sarno, and E. Zulaika, "Information Quality Ratio as a novel metric for mother wavelet selection," *Chemom. Intell. Lab. Syst.*, vol. 160, pp. 59–71, Jan. 2017.
- [12] M. Gruhne, R. Tous, J. Delgado, M. Doeller, and H. Kosch, "Introduction of an Mpeg-7 Query Format."
- [13] B. Team, "The XML Database," 27-May-2015. [Online]. Available: <http://basex.org/home/>. [Accessed: 15-Jan-2017].
- [14] M. Soleymani, M. N. Caro, E. M. Schmidt, C. Ya Sha, and Y.-H. Yang, "Emotion in Music Database - MediaEval 2013 - aka 1000 songs." [Online]. Available: <http://cvml.unige.ch/databases/emoMusic/>. [Accessed: 19-Dec-2016].
- [15] M. Soleymani, M. N. Caro, E. M. Schmidt, C.-Y. Sha, and Y.-H. Yang, "1000 Songs for Emotional Analysis of Music," *Proc. ACM Int. Multimed. Conf. Exhib.*, vol. 6, no. 1, pp. 1–14, 2015.
- [16] D. Hume, *Emotion and Moods*. Organizational behavior, 2012.
- [17] "About the Music Genres List Site," *Music Genres List*. [Online]. Available: <http://www.musicgenreslist.com/about-music-genre-site/>. [Accessed: 15-Jan-2017].
- [18] Y.-Y. Chang and Y.-C. Lin, "Music Tempo (Speed) Classification," 2005.
- [19] "MixMeister BPM Analyzer," *Softonic*. [Online]. Available: <https://mixmeister-bpm-analyzer.en.softonic.com/>. [Accessed: 15-Jan-2017].
- [20] M. Nardelli, G. Valenza, A. Greco, A. Lanata, and E. P. Scilingo, "Recognizing Emotions Induced by Affective Sounds through Heart Rate

Variability," *IEEE Trans. Affect. Comput.*, vol. 6, no. 4, pp. 385–394, Oct. 2015.

- [21] C.-H. Lin *et al.*, "SVM-Based Sound Classification Based on MPEG-7 Audio LLDs and Related Enhanced Features," in *Convergence and Hybrid Information Technology*, 2012, pp. 536–543.