

PEMISAHAN VERSE DAN REFF SECARA OTOMATIS PADA MUSIK MP3 MENGGUNAKAN KOLERASI ANTAR FRAME BERBASIS CIRI *DISCRETE COSINE TRANSFORM* (DCT)

(AUTOMATIC VERSE AND REFF SEPARATION ON MP3 MUSIC USING CORRELATION OF FRAME BASED ON DISCRETE COSINE TRANSFORM (DCT) FEATURE)

Reyfaldi Wahyu Pradana¹, Rita Magdalena², I Nyoman Apraz Ramatryana³

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik, Universitas Telkom

⁴Pusat Penelitian Elektronika dan Telekomunikasi, Lembaga Ilmu Pengetahuan Indonesia Bandung

¹reyfaldiwahyu@students.telkomuniversity.ac.id

Abstrak

Lagu merupakan suatu kesatuan musik yang terdiri atas susunan nada yang berurutan. Lagu juga merupakan seni yang melukiskan pemikiran dan perasaan manusia lewat keindahan suara. Terkadang kita menyanyikan sebuah lagu, tetapi tidak mengetahui verse dan reff nya. Penelitian ini tentang pencarian *verse* dan *reff* dengan inputan lagu yang membutuhkan potongan *verse* dan *reff* dari lagu untuk disimpan pada *database*. *Database* yang terdiri dari 25 potongan *verse* dan *reff* dari data lagu yang diproses secara manual. Proses tersebut membutuhkan waktu yang lama dikarenakan peneliti harus terlebih dahulu menentukan letak *verse* dan *reff* secara manual. Hal ini akan menjadi masalah bila jumlah *database verse* dan *reff* ditambahkan dengan data baru, sehingga diperlukan penelitian lebih lanjut untuk melakukan pemisahan *verse* dan *reff* secara *otomatis* dengan menganalisis sinyal dari *file* musik pada *mp3*. Pada penelitian ini akan merancang metode pemisahan letak *verse* dan *reff* menggunakan perhitungan korelasi antar *frame*. Sinyal audio dari *file mp3* diubah menjadi *frame-frame* kecil pada proses *framing*, selanjutnya *frame* tersebut masuk ke proses *windowing* sebelum ditransformasi menggunakan metode *Discrete Cosine Transform* (DCT). Hasil transformasi DCT kemudian akan dihitung nilai korelasi antara kumpulan *frame* untuk mencari pola kesamaan beberapa kumpulan *frame* tersebut. Dari hasil korelasi tersebut didapat pola korelasi yang sama pada kumpulan *frame* dan selanjutnya menentukan letak *verse* dan *reff*. Proses terakhir adalah pemotongan *verse* dan *reff* sesuai dengan letak yang telah ditentukan pada proses sebelumnya. Simulasi metode ini diaplikasikan pada *software* pemrograman *matlab*. Pada penelitian ini didapatkan nilai rata rata akurasi tertinggi pada genre *hip hop* dengan ukuran *frame* 1 yaitu sebesar 88,65% dengan rata rata waktu komputasi 111,2 detik.

Kata Kunci : *mp3, verse dan reff, Discrete Cosine Transform (DCT), korelasi.*

Abstract

Song is a musical arrangement of same sequence. Song is also an art that depicts human thoughts and feelings through the beauty of the sound. Sometimes we sing a song, but not knowing the verse and the refrain. This system is about to determine the position of the first verse and the reff to be added to database. Databases consist of 25 pieces of verse and reff from the songs that processed manually. However, this process take a long time because researcher must determine when the verse and the reff beginning begins and ends. This thing will become a problem if number of database of verse and reff being added with new data, so further research is required to perform the separation of verse and the refrain automatically by analyzing signals from music files on mp3. In this research will design the method of separation of verse and refrain using the calculation of the correlation between frames. The audio signal from an mp3 file is converted into a small frames on the process of framing, then frame will be entered to windowing processing before transformed using Discrete Cosine Transform (DCT). DCT transformation results will calculate the value of the correlation between frames to look for the pattern in common some collection of frames. From the correlation results obtained the same correlation patterns on the set of frame and then define the layout of the verse and the refrain. The last process is cutting of the verse and refrain in accordance with the location that was specified in the previous process. This method was applied in the simulation software matlab programming. In this research obtained average of highest accuracy from hip hop genre, frame size 1 is 88,65% with computation time 111,2 seconds.

Keyword : *mp3, verse and reff, Discrete Cosine Transform (DCT), correlation.*

1. Pendahuluan

Pada tahun 2016 telah diteliti topik pencarian judul lagu dengan input suara senandung manusia atau *hamming* oleh Arintyo [1] dengan metode *Discrete Cosine Transform* (DCT) menghasilkan akurasi 80,06%, Agatha Rizka [2] dengan metode *Fast Fourier Transform* (FFT) menghasilkan akurasi 81,67%, Ignatius [3] dengan metode *Liner Predictive Coding* (LPC) menghasilkan akurasi 80%, dan Ganang [4] dengan metode *Mel Frequency Cepstrum Coefficient* (MFCC) menghasilkan akurasi 82,57%. Pada tahun 2017 dilakukan beberapa penelitian untuk pemisahan *verse* dan *reff* pada lagu oleh Obed [5] menggunakan metode *Fast Fourier Transform* (FFT) dengan akurasi 92%, Shimon [6] dengan metode *Liner Predictive Coding* (LPC) dengan akurasi 97%, Firmansyah [7]

menggunakan metode korelasi dengan akurasi 92%. Dengan demikian, pada penelitian ini telah dirancang metode pemisahan letak *verse* dan *reff* menggunakan perhitungan korelasi antar *frame*. Sinyal audio dari *file mp3* diubah menjadi *frame-frame* kecil pada proses *framing*, selanjutnya *frame* tersebut masuk ke proses *windowing* sebelum ditransformasi menggunakan metode *Discrete Cosine Transform* (DCT). Hasil transformasi DCT dihitung korelasi antara kumpulan *frame* untuk mencari pola kesamaan beberapa kumpulan *frame* tersebut. Dari hasil korelasi tersebut didapat pola korelasi yang sama pada kumpulan *frame* dan selanjutnya ditentukan letak *verse* dan *reff*. Proses terakhir adalah pemotongan *verse* dan *reff* sesuai dengan letak yang telah ditentukan pada proses sebelumnya. Simulasi metode ini diaplikasikan pada *software* pemrograman matlab. Pada penelitian ini didapatkan nilai rata rata akurasi tertinggi pada genre hip hop dengan ukuran *frame* 1 yaitu sebesar 88,65% dengan rata rata waktu komputasi 111,2 detik. Dirumuskan beberapa masalah di tugas akhir ini yaitu bagaimana merancang algoritma untuk pemisahan letak *verse* dan *reff* secara *otomatis* dengan mengaplikasikan korelasi antar *frame* yang telah ditransformasi DCT dan bagaimana performansi parameter DCT terhadap proses pemisahan letak. Batasan masalah dalam tugas akhir ini adalah simulasi yang dirancang menggunakan masukan lagu dengan lima genre yang berbeda, lagu yang diinput berformat *mp3*, input lagu tidak dipisahkan antara vokal dan non vokal pada sinyal audio musik *mp3* yang menjadi input dari penelitian, simulasi dari metode yang dilakukan menggunakan *software* matlab r2017b, *Reff* pada setiap lagu berjumlah tiga dan *verse* berjumlah dua, dan menggunakan ekstraksi ciri *Discrete Cosine Transform*. Tujuan dan manfaat dari tugas akhir ini adalah Merancang metode pemisahan letak *verse* dan *reff* secara *otomatis* menggunakan korelasi antar *frame* yang telah dilakukan transformasi DCT dan menganalisis pengaruh parameter DCT terhadap performansi pemisahan letak *verse* dan *reff* secara *otomatis*.

2. Landasan Teori

A. Konsep Dasar Suara

Suara adalah kompresi mekanikal atau gelombang longitudinal yang merambat melalui medium. Suara adalah kompresi mekanikal atau gelombang longitudinal yang merambat melalui medium. Kebanyakan dari suara merupakan gabungan beberapa sinyal, tetapi suara murni secara teoritis dapat dijelaskan dengan frekuensi yang diukur dalam Hertz (Hz) dan amplitudo atau tekanan suara dengan pengukuran dalam *desibel* [8].

Telinga manusia dapat mendengar bunyi dengan batas frekuensi dari 20 Hz sampai 20 kHz pada amplitude umum dengan berbagai variasi dalam kurva responnya. Suara diatas 20 kHz disebut ultrasonik dan dibawah 20 Hz disebut infrasonik.

B. Frame

Frame adalah proses yang digunakan sebagai pembagian sinyal audio menjadi beberapa *frame*. Pada perancang sistem ini menggunakan beberapa nilai *frame* diantaranya yaitu dengan nilai 1 second, 0,5 second, dan 0,25 second. Semakin besar *frame*, maka semakin cepat waktu komputasinya dikarenakan ciri yang dicocokkan oleh sistem menjadi lebih sedikit. Satu *frame* terdiri dari beberapa *sample*, tergantung tiap berapa detik suara yang akan di sampling dan berapa besar frekuensi samplingnya_[1]. Sinyal hasil dari *framing* adalah sinyal terpotong yang diskontinu_[1]. Berikut persamaan *frame*:

$$\text{Jumlah Frame} = ((L - n)/M) \quad (1)$$

C. Windowing

Dalam melakukan pemrosesan sinyal, maka dari input yang dimasukan akan terbentuk sinyal yang magnitudenya bervariasi pada awal maupun akhir *frame*. Hal tersebut menghambat pemrosesan sinyal dan menghasilkan keluaran yang kurang akurat. Untuk itu perlu diaplikasikan suatu window penghalus pada setiap *frame* dengan melakukan overlapping antara suatu *frame* dengan *frame* yang lain sehingga dapat dibangkitkan suatu *feature* yang lebih halus sepanjang durasi waktu tersebut. Fungsi window yang digunakan disini adalah :

1 Window Hamming

$$W(n) = 0.54 - 0.46 \cos \frac{2\pi n}{N-1} \quad (2)$$

2 Window Hanning

$$W(n) = 0,5 (1 - \cos \frac{2\pi n}{N-1}) \quad (3)$$

3 Window Rectangular

$$W(n) = 1 \quad (4)$$

D. Discrete Cosine Transform

Discrete Cosinus Transform (DCT) mengungkapkan urutan titik data yang terbatas dalam hal jumlah fungsi kosinus yang beresilasi pada frekuensi yang berbeda. DCT penting untuk berbagai aplikasi dalam sains dan teknik, dari kompresi *lossy* audio (misalnya MP3) dan gambar (misalnya JPEG) (di mana komponen frekuensi tinggi kecil dapat dibuang), ke metode spektral untuk solusi numerik dari persamaan diferensial parsial. Penggunaan kosinus daripada fungsi sinus sangat penting untuk kompresi, karena ternyata lebih sedikit fungsi kosinus diperlukan untuk mendekati sinyal yang khas, sedangkan untuk persamaan diferensial, kosinus mengekspresikan pilihan tertentu dari kondisi batas_[1]. *Discrete Cosine Transform* (DCT) biasa digunakan untuk mengubah sebuah sinyal menjadi komponen frekuensi dasarnya_[9]. Persamaan DCT dapat dilihat dibawah ini :

$$X(k) = C(k) \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x(n) \cos\left(\frac{(2n+1)k\pi}{2N}\right), \quad (5)$$

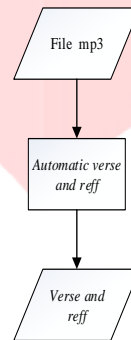
dengan $k = 0, \dots, k - 1$

$$\text{Dimana } C(k) = \begin{cases} 2^{-\frac{1}{2}} & , \text{ untuk } k = 0 \\ 1 & , \text{ untuk } k \text{ lainnya} \end{cases} \quad (6)$$

2. Perancangan dan Simulasi

A. Diagram Sistem

Sistem perangkat lunak yang akan dirancang pada tugas akhir ini adalah Matlab. Tahapan yang akan dilakukan yaitu : input audio digital, *Automatic verse and reff*, dan *verse and reff*

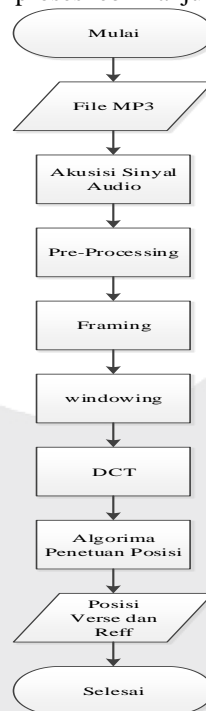


Gambar 2.1 Diagram Blok Sistem

Automatic verse and reff diatas digunakan untuk mengubah file MP3 menjadi potongan *verse dan reff*.

B. Alur Kerja Sistem

Penentuan posisi menggunakan metode *Discrete Cosine Transform* untuk ekstraksi ciri, korelasi dua dimensi untuk mendapatkan koefisien korelasi 2-D. Proses dimulai dengan memilih *file mp3* dan selanjutnya didapatkan data sinyal audio. Lalu dilakukan pemotongan sinyal audio menjadi potongan kecil yang disebut *frame* dan proses ini disebut *framing*. Langkah berikutnya adalah melakukan proses *windowing* pada setiap bagian sinyal yang telah dibuat sebelumnya. Hal ini dilakukan untuk meminimalkan diskontinuitas pada bagian awal dan akhir sinyal. Selanjutnya dilakukan proses transformasi DCT untuk mendapatkan nilai koefisien DCT. Koefisien DCT disini merupakan ciri yang membedakan satu *frame* dengan *frame* yang lain. Dan proses terakhir penentuan posisi *verse dan reff* dengan menggunakan proses korelasi 2-D. proses lebih lanjut akan dijelaskan pada gambar 3.2.



Gambar 3.2 Alur kerja sistem

4. Pengujian Sistem

Pada bab ini dilakukan beberapa pengujian terhadap sistem yang telah dirancang. Setelah dilakukan pengujian maka hasil pengujian tersebut dianalisis dan disimpulkan hasilnya.

Untuk mengetahui performansi sistem yang telah dirancang, maka dilakukan pengujian terhadap sistem dengan beberapa skenario pengujian yaitu:

- 1 Pengujian dan analisis pengaruh ukuran *frame* terhadap koefisien korelasi 2 dimensi
- 2 Pengujian dan analisis pengaruh jenis window terhadap koefisien korelasi
- 3 Pengujian dan analisis sistem penentuan *verse* dan *reff* dengan referensi
- 4 Pengujian dan analisis sistem penentuan *verse* dan *reff* tanpa referensi (*otomatis*)

A. Skenario 1 pengujian dan analisis pengaruh ukuran *frame* terhadap koefisien korelasi 2 dimensi

Pada skenario ini dilakukan pengujian terhadap ukuran *frame* yaitu 250 ms, 500 ms, dan 1000 ms.

Tabel 4. 1 Tabel nilai koefisien korelasi 2D terhadap *frame*

| Ukuran <i>Frame</i> (s) | Verse 2 | Reff 2 | Reff 3 |
|----------------------------|-----------------------|-----------------------|-----------------------|
| | rata-rata nilai KK-2D | rata-rata nilai KK-2D | rata-rata nilai KK-2D |
| 0,25 | 0,14 | 0,29 | 0,21 |
| 0,5 | 0,15 | 0,32 | 0,23 |
| 1 | 0,15 | 0,34 | 0,25 |

Tabel 4.1 Tabel top rank posisi terhadap *frame*

| Ukuran <i>Frame</i> (s) | Verse 2 | Reff 2 | Reff 3 |
|----------------------------|----------------|-----------------|-----------------|
| | ranking posisi | rangking posisi | rangking posisi |
| 0,25 | 256,56 | 386,88 | 329,36 |
| 0,5 | 170,76 | 179,44 | 187,12 |
| 1 | 27,52 | 14,92 | 19,44 |

Tabel 4.2 Tabel waktu proses penentuan *verse* dan *reff* terhadap *frame*

| Ukuran <i>Frame</i> (s) | waktu proses (s) | |
|----------------------------|------------------|-------|
| | verse | reff |
| 0,25 | 11,32 | 11,02 |
| 0,5 | 5,72 | 5,56 |
| 1 | 3,14 | 3,02 |

Pada pengujian ini, ukuran *frame* 1000 ms menghasilkan nilai yang paling baik dengan nilai koefisien korelasi 2D terlihat pada tabel 4.1 dimana semakin besar nilainya maka hasil pengujian akan semakin baik. Pada tabel 4.2 ranking posisi didapat dari urutan nilai korelasi yang termasuk dalam *reff* atau *verse*, yang membuat rankingnya bergeser menjadi lebih besar karena adanya nilai korelasi yang angkanya sama. Dan ranking posisi yang paling baik didapatkan oleh *frame* 1000 ms dimana semakin kecil nilai pada rank posisi maka hasil pengujian akan semakin baik. Waktu proses penentuan *verse* dan *reff* paling baik juga didapatkan pada ukuran *frame* 1000 ms terlihat pada tabel 4.3. Dari skenario ini akurasi terbaik didapat pada genre Hiphop yaitu 100%.

B. Skenario 2 pengujian dan analisis pengaruh jenis window terhadap koefisien korelasi 2 dimensi

Skenario kedua ini akan dilakukan dengan menambahkan *windowing* dengan 3 jenis *windowing* yaitu *rectangular*, *hanning*, dan *humming*.

Tabel 4.3 Tabel nilai koefisien korelasi 2D terhadap *windowing*

| Jenis <i>Windowing</i> | Verse 2 | Reff 2 | Reff 3 |
|------------------------|-----------------------|-----------------------|-----------------------|
| | rata-rata nilai KK-2D | rata-rata nilai KK-2D | rata-rata nilai KK-2D |
| <i>Rectangular</i> | 0,15 | 0,34 | 0,25 |
| <i>hanning</i> | 0,15 | 0,33 | 0,23 |
| <i>humming</i> | 0,15 | 0,33 | 0,23 |

Tabel 4.4 Tabel top rank posisi terhadap *windowing*

| Jenis <i>Windowing</i> | Verse 2 | Reff 2 | Reff 3 |
|---------------------------|--------------------------|--------------------------|--------------------------|
| | rata-rata ranking posisi | rata-rata ranking posisi | rata-rata ranking posisi |
| <i>Rectangular</i> | 27,52 | 14,92 | 19,44 |
| <i>Hanning</i> | 38,48 | 21,8 | 25,48 |
| <i>Humming</i> | 37,4 | 20,88 | 25,12 |

Tabel 4.5 Tabel waktu proses penentuan *verse* dan *reff* terhadap *windowing*

| Jenis <i>Windowing</i> | waktu proses (s) | |
|------------------------|------------------|------|
| | verse | reff |
| <i>Rectangular</i> | 3,27 | 3,20 |
| <i>Hanning</i> | 3,16 | 3,09 |
| <i>Hamming</i> | 3,16 | 3,07 |

Pada pengujian ini *rectangular windowing* menghasilkan nilai yang paling baik dengan nilai koefisien korelasi 2D terlihat pada tabel 4.4 dimana semakin besar nilainya maka hasil pengujian akan semakin baik. Pada tabel 4.5 ranking posisi didapat dari urutan nilai korelasi yang termasuk dalam *reff* atau *verse*, yang membuat rankingnya bergeser menjadi lebih besar karena adanya nilai korelasi yang angkanya sama. Dan ranking posisi yang paling baik didapatkan oleh *rectangular windowing* dimana semakin kecil nilai rank posisi maka hasil pengujian akan semakin baik. Waktu proses pada penentuan *verse* dan *reff* paling baik didapatkan pada *hamming windowing* terlihat pada tabel 4.6. Dari skenario ini akurasi terbaik didapat pada genre Hiphop yaitu 100%.

C. Skenario 3 pengujian dan analisis sistem penentuan *verse* dan *reff* dengan referensi

Dalam skenario ini dilakukan pengujian dan analisis *verse* dan *reff* dengan referensi. Dalam pengujian ini digunakan data 25 lagu yang dikelompokkan menjadi 5 *genre* yaitu pop, EDM, funk, rock, dan hip-hop.

a. Pengujian terhadap lagu dengan genre EDM

Tabel 4.6 Hasil Data Lagu EDM manual

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| EDM | 1000 s | 94,74% | 3,08 |
| | | 93,33% | 2,93 |
| | | 0,00% | 2,52 |
| | | 100,00% | 3,45 |
| | | 70,83% | 3,19 |

Pada pengujian untuk jenis lagu EDM didapat nilai akurasi terbaik pada angka 100% dengan waktu komputasi 3,45 detik. Sedangkan akurasi terendah sebesar 0% dengan waktu komputasi 2,52 detik.

b. Pengujian terhadap lagu dengan genre Funk

Tabel 4.7 Hasil Data Lagu Funk manual

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi(detik) |
|------------|---------------------|---------|------------------------|
| FUNK | 1000 s | 94,44% | 2,62 |
| | | 0% | 4,23 |
| | | 21,62% | 5,16 |
| | | 38,71% | 4,31 |
| | | 87,50% | 3,68 |

Pada pengujian untuk jenis lagu Funk didapat nilai akurasi terbaik pada angka 94,44% dengan waktu komputasi 2,62 detik. Sedangkan akurasi terendah sebesar 0% dengan waktu komputasi 4,23 detik.

c. Pengujian terhadap lagu dengan *genre* Hip Hop

Tabel 4.8 Hasil Data Lagu Hip Hop manual

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| HIP HOP | 1000 s | 92,86% | 2,495 |
| | | 0% | 4,395 |
| | | 0% | 3,724 |
| | | 0% | 3,226 |
| | | 66,67% | 4,314 |

Pada pengujian untuk jenis lagu Hip Hop didapat nilai akurasi terbaik pada angka 92,86% dengan waktu komputasi 2,495 detik. Sedangkan akurasi terendah sebesar 0% dengan waktu komputasi 4,395 detik.

d. Pengujian terhadap lagu dengan *genre* Pop

Tabel 4.9 Hasil Data Lagu Pop manual

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| POP | 1000 s | 95,83% | 3,71 |
| | | 68,75% | 3,84 |
| | | 62,50% | 3,96 |
| | | 100% | 3,65 |
| | | 95,83% | 3,84 |

Pada pengujian untuk jenis lagu Pop didapat nilai akurasi terbaik pada angka 100% dengan waktu komputasi 3,65 detik. Sedangkan akurasi terendah sebesar 62,50% dengan waktu komputasi 3,96 detik.

e. Pengujian terhadap lagu dengan *genre* Rock

Tabel 4.10 Hasil Data Lagu Rock manual

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| ROCK | 1000 s | 100% | 4,86 |
| | | 0% | 4,84 |
| | | 62,50% | 4,90 |
| | | 88,46% | 4,74 |
| | | 0% | 3,51 |

Pada pengujian untuk jenis lagu Rock didapat nilai akurasi terbaik pada angka 100% dengan waktu komputasi 4,86 detik. Sedangkan akurasi terendah sebesar 0% dengan waktu komputasi 4,84 detik.

D. Skenario 4 pengujian dan analisis sistem penentuan verse dan reff tanpa referensi (otomatis)

Dalam skenario ini dilakukan pengujian dan analisis sistem penentuan verse dan reff tanpa referensi. Dalam pengujian digunakan data 25 lagu yang dikelompokkan menjadi 5 genre yaitu pop, EDM, funk, rock, dan hip-hop. Dari hasil pengujian, dilakukan analisis akurasi menggunakan persamaan rumus (3.3).

a. Pengujian terhadap lagu dengan genre EDM

Tabel 4.11 Hasil Data Lagu EDM otomatis

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi(detik) |
|------------|---------------------|---------|------------------------|
| EDM | 1000 s | 84,22% | 85 |
| | | 87,50% | 98 |
| | | 86,67% | 93 |
| | | 92,86% | 97 |
| | | 61,12% | 89 |

Pada pengujian untuk jenis lagu EDM didapat nilai akurasi terbaik pada angka 92,86% dengan waktu komputasi 97 detik. Sedangkan akurasi terendah sebesar 61,12% dengan waktu komputasi 89 detik.

b. Pengujian terhadap lagu dengan genre FUNK

Tabel 4.12 Hasil Data Lagu Funk otomatis

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| FUNK | 1000 s | 77,78% | 89 |
| | | 72,98% | 115 |
| | | 95,84% | 135 |
| | | 50% | 132 |
| | | 77,42% | 103 |

Pada pengujian untuk jenis lagu FUNK didapat nilai akurasi terbaik pada angka 95,84% dengan waktu komputasi 135 detik. Sedangkan akurasi terendah sebesar 50% dengan waktu komputasi 132 detik.

c. Pengujian terhadap lagu dengan genre HIP HOP

Tabel 4.13 Hasil Data Lagu Hip Hop otomatis

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| HIP HOP | 1000 s | 90% | 114 |
| | | 91,67% | 97 |
| | | 89,66% | 112 |
| | | 85,72% | 136 |
| | | 86,37% | 97 |

Pada pengujian untuk jenis lagu HIP HOP didapat nilai akurasi terbaik pada angka 91,67% dengan waktu komputasi 97 detik. Sedangkan akurasi terendah sebesar 85,72% dengan waktu komputasi 136 detik.

d. Pengujian terhadap lagu dengan genre POP

Tabel 4.14 Hasil Data Lagu Pop otomatis

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi (detik) |
|------------|---------------------|---------|-------------------------|
| POP | 1000 s | 81,25% | 88 |
| | | 81,25% | 97 |
| | | 80,96% | 124 |
| | | 87,50% | 96 |
| | | 58,34% | 122 |

Pada pengujian untuk jenis lagu POP didapat nilai akurasi terbaik pada angka 87,50% dengan waktu komputasi 96 detik. Sedangkan akurasi terendah sebesar 58,34% dengan waktu komputasi 122 detik.

e. Pengujian terhadap lagu dengan genre ROCK

Tabel 4.15 Hasil Data Lagu Rock otomatis

| Jenis Lagu | Ukuran <i>frame</i> | Akurasi | Waktu Komputasi(detik) |
|------------|---------------------|---------|------------------------|
| ROCK | 1000 s | 57,7% | 151 |
| | | 97,06% | 105 |
| | | 33,33% | 212 |
| | | 91,66% | 136 |
| | | 0% | 95 |

Pada pengujian untuk jenis lagu ROCK didapat nilai akurasi terbaik pada angka 97.06% dengan waktu komputasi 105 detik. Sedangkan akurasi terendah sebesar 0% dengan waktu komputasi 95 detik.

5. Kesimpulan dan Saran

Berdasarkan hasil pengujian sistem dan analisis sistem dalam Tugas Akhir ini, dapat diambil kesimpulan Penentuan letak *verse* dan *reff* menggunakan metode DCT dapat berjalan dengan baik namun pada data dengan tempo yang cenderung sama hasilnya masih belum optimal karena sinyal keluaran sulit dibedakan. Didapatkan rata rata akurasi yang baik pada genre hip hop sebesar 88,65% dengan waktu komputasi rata rata 111,2 detik, dan rata rata selisih waktu 2,6 detik. Parameter yang mempengaruhi tingkat akurasi sistem adalah *framing* dengan ukuran *frame* 1000 ms dan *windowing* dengan jenis *rectangular windowing*. Adapun saran yang dapat digunakan untuk penelitian Tugas Akhir selanjutnya, yaitu Pengembangan metode ekstraksi ciri lain yang dapat meningkatkan akurasi yang lebih baik. Pengembangan cara kerja sistem yang mempunyai waktu komputasi lebih baik. Pengembangan terhadap lagu dengan genre yang berbeda. Pengembangan pencarian bagian-bagian lagu untuk semua genre lagu.

Daftar Pustaka

- [1] Archamadi, Arintyo. Analisis dan Simulasi Identifikasi Judul Lagu dari Senandung Manusia Menggunakan Ekstraksi Ciri *Discrete Cosine Transform*. Bandung: Universitas Telkom, 2016.
- [2] Rizka Prassiwi, Agatha. Analisis dan Simulasi Klasifikasi Judul Lagu dari Senandung Manusia Menggunakan Ekstraksi Ciri *Fast Fourier Transform*. Bandung: Universitas Telkom, 2016.
- [3] Yoslan Kurniawan, Ignatius. Analisis dan Simulasi Identifikasi Judul Lagu melalui Senandung Manusia Menggunakan Ekstraksi Ciri *Linier Predictive Coding*. Bandung: Universitas Telkom, 2016.
- [4] Wicaksono, Ganang. Analisis dan Simulasi Klasifikasi Judul Lagu dari Senandung Manusia Menggunakan Ekstraksi Ciri *Mel Frequency Cepstral Coeficient*. Bandung: Universitas Telkom, 2016.
- [5] Gabriel Filemon, Obed. Perancangan dan Simulasi Pemisahan *Reff* Lagu dengan Metode *Fast Fourier Transform*. Bandung: Universitas Telkom, 2017.
- [6] Anterio Armando, Shimon. Analisis dan Simulasi Pencarian *Verse dan reff* Lagu pada Musik Digital dengan Metode *Linear Predictive Coding*. Bandung: Universitas Telkom, 2017.
- [7] Patriandhika, Firmansyah. Simulasi dan Analisa Pencarian *Verse dan reff* Lagu pada Musik Digital dengan Metode Korelasi. Bandung: Universitas Telkom, 2017.
- [8] Mcloughlin, Ian. *Applied Speech and Audio Processing with Matlab Examples*. United States: Cambridge University Press. 2009.
- [9] Andrew B. Watson. Image Compression Using the Discrete Cosine Transform. *Mathematical Journal*, 4(1), 1994, pp. 81-88.