

AUTOMATIC CHORUS DETECTION FOR INDONESIAN MUSIC USING REFRAIN DETECTING METHOD (REFRAID)

Ichsan Permana Putra^{*1}, Elvia Budianita², Febi Yanto³, Yusra⁴

^{1,2,3,4}Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia

Email: ¹ichsan.permana.putra@students.uin-suska.ac.id, ²elvia.budianita@uin-suska.ac.id, ³febiyanto@uin-suska.ac.id, ⁴198401232015032001@uin-suska.ac.id

(Naskah masuk: 14 Maret 2022, Revisi: 3 April 2022, diterbitkan: 20 Agustus 2022)

Abstract

Music has become an important part of human life, chorus is part of the musical structure that makes some impression on music, people are generally very familiar with the chorus in music because the chorus is often repeated on music. Automatic chorus detection is a part of Music Information Retrieval which is considered important for building music analysis system with human-like patterns. Refrain Detecting Method (RefraiD) select the chorus by grouping various repeating parts of the music, evaluating the intensity level of the melody from each group, then selecting the group with the highest melodic intensity as the chorus. This paper intends to implement RefraiD in Indonesian pop and dangdut music by downloading 20 pop music videos and 20 dangdut music videos from YouTube then process it with Information retrieval using Python. The results of this paper indicate that the RefraiD method can be used to detect the chorus on Indonesian music with F measure of 91.8% for dangdut music and 91.5% for pop music.

Keywords: *Chorus, Dangdut Music, Music, Music Information Retrieval, Pop Music, Refrain Detecting Method*

PENDETEKSIAN CHORUS OTOMATIS MUSIK INDONESIA MENGGUNAKAN REFRAIN DETECTING METHOD (RefraiD)

Abstrak

Musik telah menjadi bagian penting dalam kehidupan manusia, chorus merupakan bagian dari struktur musik yang membuat musik bermakna, pecinta musik umumnya sangat hafal dengan bagian *chorus* dari sebuah musik karena *chorus* sering dilafalkan secara berulang. Pendeteksian chorus secara otomatis merupakan bagian *Music Information Retrieval* yang dianggap penting untuk membangun sistem analisa musik dengan pola layaknya manusia. Refrain Detecting Method (RefraiD) melakukan pemilihan *chorus* dengan cara mengelompokkan berbagai bagian musik yang berulang, mengevaluasi tingkat intensitas melodi dari tiap kelompok, lalu memilih kelompok dengan intensitas melodi tertinggi sebagai *chorus*. Penelitian ini bermaksud untuk mengimplementasikan RefraiD pada musik pop dan dangdut Indonesia dengan cara mengunduh video YouTube sebanyak 20 video musik bergenre pop dan 20 video musik bergenre dangdut kemudian dilakukan proses *Information retrieval* menggunakan Python. Hasil penelitian ini menunjukkan bahwa metode RefraiD dapat digunakan untuk mendeteksi chorus musik Indonesia dengan nilai *F measure* sebesar 91.8 % pada musik dangdut dan 91.5% pada musik pop.

Kata kunci: *Chorus, Musik, Musik Dangdut, Musik Pop, Music Information Retrieval, Refrain Detecting Method.*

1. PENDAHULUAN

Musik telah menjadi bagian penting dalam kehidupan manusia sebagai bentuk kebudayaan pada setiap kelompok masyarakat yang tidak dapat terpisahkan[1]. Setiap karya musik memiliki struktur musik yang saling berhubungan antara elemen dasar musik yang membuat musik bermakna[2]. Struktur musik terbagi dalam beberapa bagian, yaitu *intro*,

bridge, verse, chorus, reff, interlude, modulasi, *ending, coda* dan *outro*.

Chorus merupakan bagian tematik yang paling representatif, membangkitkan semangat, dan menonjol dalam struktur musik[3], [4]. Pecinta musik dalam mendengarkan ataupun bernyanyi, umumnya sangat hafal dengan bagian *chorus* dari sebuah musik, hal ini disebabkan karena bagian

chorus dalam sebuah musik sering dilafalkan secara berulang dan biasanya bagian *chorus* adalah makna dari sebuah musik[5]. Salah satu studi dalam menganalisis musik adalah *Music Information Retrieval*.

Music Information Retrieval (MIR) adalah bagian dari *Information Retrieval* yang dimana MIR merupakan salah satu jenis bidang penelitian gabungan dari berbagai disiplin ilmu yang berhubungan dengan teknik untuk mencari dan mengambil struktur data pada musik[6], [7]. Pendeteksian *Chorus* otomatis adalah salah satu bagian penting dalam studi *Music Information Retrieval* (MIR)[8].

Pendeteksian otomatis bagian *chorus* sangat penting dalam membangun sebuah sistem analisa musik yang dapat mengenali musik layaknya manusia, sistem analisa musik tersebut berguna dalam berbagai pengaplikasian, contohnya pemotongan *chorus* otomatis di radio musik, pencarian musik yang mendengarkan *preview chorus* musik secara langsung, dan juga dapat meningkatkan efisiensi dan akurasi dalam pencarian musik berdasarkan potongan *keyword* yang ada dalam *chorus* sebuah musik[9]. Pemisahan *chorus* jika dilakukan secara manual membutuhkan waktu yang lama, jadi diperlukan metode untuk menentukan posisi *chorus* lalu memisahkannya secara otomatis merupakan solusi dari masalah tersebut[5], [10], [11]. Metode yang mampu melakukan pendeteksian otomatis bagian *chorus* salah satunya adalah *Refrain Detecting Method*.

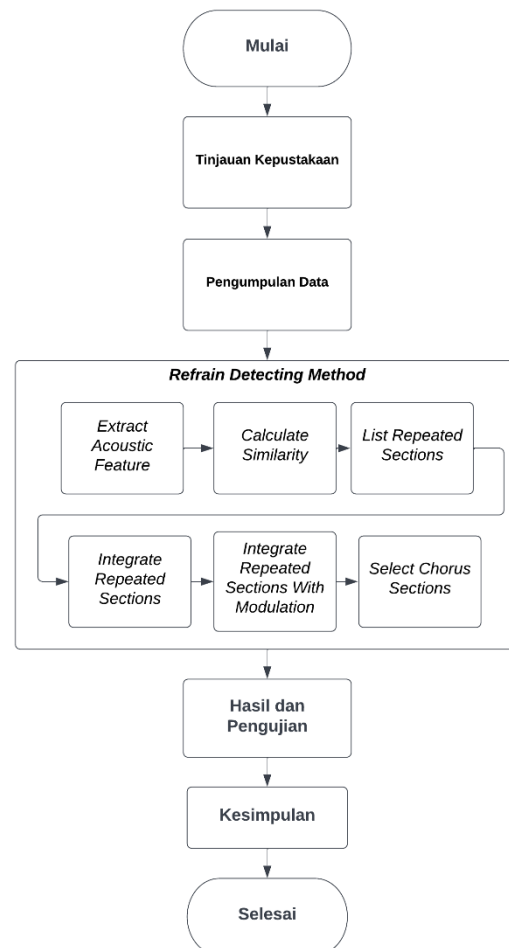
Refrain Detecting Method (RefraiD) merupakan suatu metode yang dapat memilah *chorus* dari suatu musik. RefraiD pertama kali dicetuskan oleh Masataka Goto pada tahun 2003. Konsep dari metode RefraiD adalah melakukan pemilahan *chorus* dengan cara mengelompokkan berbagai bagian musik yang berulang, mengevaluasi tingkat intensitas melodi dari tiap kelompok, lalu memilah kelompok dengan intensitas melodi tertinggi sebagai *chorus*[12].

Penelitian sebelumnya pernah dilakukan menggunakan metode RefraiD oleh Kento Watanabe dan Masataka Goto tahun 2020 yang meneliti penentuan *chorus* pada musik bahasa Inggris dan bahasa Jepang. Dari hasil penelitian tersebut, mendapatkan hasil akurasi *F-Measure* sebesar 78,1% dan 83,4%[9]. Penelitian lainnya juga dilakukan pada musik yang populer tahun 1990-an dengan hasil akurasi sebesar 80%[13].

Pada penelitian ini penulis menggunakan metode *Refrain Detecting Method* (RefraiD), untuk mendeteksi bagian *chorus* dalam musik Indonesia genre pop dan dangdut. Penelitian ini diharapkan mampu memberikan solusi dalam mendeteksi *chorus* secara otomatis dari sinyal audio digital musik Indonesia genre pop dan dangdut.

2. METODE PENELITIAN

Penelitian ini memiliki alur penelitian dengan tahapan didalam proses *Refrain Detecting Method* yaitu : *Extract Acoustic Feature*, *Calculate Similarity*, *List Repeated Sections*, *Integrate Repeated Sections*, *Integrate Repeated Sections With Modulation*, dan *Select Chorus Sections*. Langkah penelitian dapat dilihat secara lengkap pada gambar 1.



Gambar 1. Alur Metodologi Penelitian

2.1. Tinjauan Kepustakaan

Tinjauan kepustakaan adalah tahap penelitian untuk mencari referensi teori tentang penelitian yang akan dilakukan. Referensi dapat bersumber dari buku, jurnal, dan artikel. Tinjauan kepustakaan adalah tahap pertama dalam pengerjaan penelitian. Tinjauan kepustakaan pada penelitian ini dilakukan dengan mencari sumber teori tentang musik yang dibutuhkan pada jurnal dan buku untuk dijadikan referensi.

2.2. Pengumpulan Data

Pada tahap ini penulis mengumpulkan musik dengan mengunduh video YouTube sebanyak 20 video musik bergenre pop dan 20 video musik

bergenre dangdut dengan kata kunci pencarian untuk musik dangdut dapat dilihat pada tabel 1.

Tabel 1. Daftar Kata Kunci Pencarian Musik Dangdut

No.	Kata Kunci
1	Lagu dangdut populer
2	Meggi Z
3	Via Vallen
4	Rhoma Irama
5	Nella Kharisma
6	Cita Citata

Kata kunci pencarian musik pada YouTube untuk musik pop dapat dilihat pada tabel 2.

Tabel 2. Daftar Kata Kunci Pencarian Musik Pop

No.	Kata Kunci
1	Lagu pop populer
2	Lagu pop tahun 2000 an
3	Lagu Armada
4	Lagu Geisha
5	Lagu Vierra
6	Lagu Peterpan

Data musik bergenre dangdut dapat dilihat pada tabel 3.

Tabel 3. Data Musik Dangdut

No.	Judul
1	Ayu Ting Ting - Sambalado
2	Bebizy - Cinta Tulalit
3	Camel Petir - CKC Cuma Kamu Cin
4	Duo Angrek - Cikini Gondangdia
5	Duo Serigala - Abang Goda
6	Meggi Z - Benang Biru
7	Meggi Z - Cinta Hitam
8	Meggi Z - Mata Air Cinta
9	Nella Kharisma - Cie Cie
10	Nella Kharisma - Sakit Dalam Bercinta
11	Regina - Cinta Basi
12	Rhoma Irama - Bujangan
13	Rhoma Irama - Judi
14	Sakitnya Tuh Disini - Cita Citata
15	Siti Badriah - Suamiku Kawin Lagi
16	Ucie Sucita - Cinta Tak Terbatas Waktu
17	Via Vallen - 5 Centi
18	Via Vallen - Cinta Kurang Gizi
19	Via Vallen - Sayang
20	Zaskia Gotik - Sudah Cukup Sudah

Data musik bergenre pop dapat dilihat pada tabel 4.

Tabel 4. Data Musik Pop

No.	Judul
1	Andmesh - Cinta Luar Biasa
2	Ari Lasso - Arti Cinta
3	Armada - Apa Kabar Sayang
4	Anneth - Mungkin Hari Ini Esok Atau Nanti
5	Armada - Harusnya Aku
6	Bondan - Ya Sudahlah
7	Cassandra - Cinta Terbaik
8	Geisha - Cinta Dan Benci
9	Geisha - Lumpuhkan Ingatanku
10	Geisha - Pergi Saja
11	Hijau Daun - Suara Ku Berharap
12	Irwansyah - Pecinta Wanita
13	Tulus - Hati Hati di Jalan
14	Noah - Tak Lagi Sama
15	Peterpan - Ada Apa Denganmu
16	Peterpan - Menghapus Jejakmu
17	Ungu - Disini Untukmu
18	Vagetoz - Kehadiranmu
19	Tiara Andini - Merasa Indah

2.3. Information Retrieval

Information retrieval atau sering disebut “temu kembali informasi” adalah ilmu yang mempelajari prosedur-prosedur dan metode-metode untuk menemukan kembali informasi yang tersimpan dari berbagai sumber. *Information Retrieval* (IR) berkaitan dengan stuktur, analisa, organisasi, penyimpanan, pencarian, dan penyebaran informasi.

Sistem IR dirancang untuk membuat kumpulan informasi yang tersedia untuk banyak pengguna [13]. *Information retrieval* bertujuan untuk mempelajari dan memahami proses untuk merancang, membangun, dan menguji pengambilan sistem yang dapat memfasilitasi komunikasi yang efektif dari informasi yang diinginkan antara manusia pengguna dan generator manusia. Secara tradisional informasi mengambil bentuk teks, menyirat bahwa IR identik dengan dokumen. Namun, sekarang IR telah memperluas ruang lingkup lingkungan multimedia yang berkaitan dengan penyimpanan dan pengambilan gambar, grafik, suara, komponen perangkat lunak, dokumen dan kantor [14].

2.4. Music Information Retrieval

Music Information Retrieval (MIR) adalah bidang penelitian yang berhubungan dengan teknik pengolahan data [6]. MIR dapat digunakan untuk kepentingan komersial, musikologi, teori musik serta pendidikan musik. Penggunaan MIR dalam kontribusi dalam penelitian musik adalah untuk mencari contoh musik dan menganalisis menemukan teknik komposisi yang sesuai dari kumpulan musik [15]. MIR adalah salah satu bidang yang sedang banyak digunakan dalam ilmu komputer. Sebagai bidang keilmuan, MIR mempelajari hal yang tentang pencarian informasi pada media musik dan hubungan metadata pada file musik dalam suatu basis data yang berhubungan. Selain melakukan analisa metadata dalam musik, penelitian MIR juga dilakukan dengan penerapan algoritma data mining untuk pengelompokan data musik[16].

2.5. Python

Python adalah bahasa pemrograman yang simpel dan mudah dibaca. Python memiliki *library* yang besar dengan berbagai fungsionalitas, karena besarnya komunitas pengguna Python di seluruh dunia. Salah satu fitur utama Python adalah kedinamisan bahasa pemrograman dengan dilengkapi manajemen memori yang bekerja dengan baik. Salah satu kelebihan Python dapat digunakan pada berbagai perangkat lunak di berbagai platform. Python dapat dipergunakan secara bebas dalam berbagai hal, bahkan untuk MIR[17].

2.6. Refrain Detecting Method

Refrain Detecting Method (RefraiD) adalah suatu metode dapat mendeteksi *chorus* dari sebuah musik. RefraiD diciptakan pada tahun 2003 oleh Masataka Goto. Konsep dari metode RefraiD adalah melakukan pendeteksian *chorus* dengan cara menyatukan berbagai bagian musik yang berulang dalam satu grup, lalu mengevaluasi tingkat intensitas melodi dari tiap grup, lalu menentukan bahwa grup dengan intensitas melodi terbesar sebagai *chorus* [12].

2.7. Extract Acoustic Feature

Metode RefraiD dimulai dengan melakukan perhitungan *chroma vector* untuk mengekstrak fitur akustik dari sinyal audio. *Chroma vector* merepresentasikan distribusi magnitudo pada *chroma* yang diubah menjadi 12 kelas nada dalam satu oktaf (C, C#, D, D#, E, F, F#, G, G#, A, A#, dan B). Persamaan *chroma vector* dapat dilihat pada persamaan (1)[18].

$$v_c(t) = \sum_{h=0}^{oct_h} \int_{L}^{\infty} BPF_{c,h}(f) \psi_p(f, t) df \quad (1)$$

Dimana $BPF_{c,h}(f)$ adalah adalah *filter bandpass* yang melewatkan sinyal pada frekuensi skala log $F_{c,h}$ (*cent*) dari kelas nada c (*chroma*) pada posisi oktaf h (*height*). *filter bandpass* adalah *filter* yang memungkinkan sinyal antara dua frekuensi tertentu lewat. Rumus untuk mencari $F_{c,h}$ didapat dari persamaan (2)[18].

$$F_{c,h} = 1200h + 100(c - 1) \quad (2)$$

Dari persamaan (2) frekuensi dalam *hertz* f_{Hz} diganti dengan frekuensi dalam *cent* f_{cent} yang menyebabkan adanya 100 *cent* menjadi seminada *temper* dan 1200 menjadi sebuah oktaf. Pencarian frekuensi dalam bentuk *cent* dapat dilihat pada persamaan (3)[18].

$$f_{cent} = 1200 \log_2 \frac{f_{Hz}}{440 \times 2^{\frac{3}{12} - 5}} \quad (3)$$

$BPF_{c,h}(f)$ didefinisikan dengan jendela *Hanning* untuk pemrosesan sinyal, sebagai fungsi menghaluskan nilai, yang dapat dilihat pada persamaan 4[18].

$$BPF_{c,h}(f) = \frac{1}{2} \left(1 - \cos \frac{2\pi(f - (F_{c,h} - 100))}{200} \right) \quad (4)$$

Tahapan *Extract Acoustic Feature* ditampilkan dalam *pseudocode* di bawah.

```
extract_acoustic_feature
y, sr ← librosa.load(input_file)
song_length_sec ← y.shape[0] / float(sr)
S ← numpy.abs(librosa.stft(y, n_fft))^2
chroma ← librosa.feature.chroma_stft(S, sr)
return chroma, y, sr, song_length_sec
```

2.8. Calculate Similarity

Kemiripan *chroma vector* dari sinyal audio dapat dilihat pada persamaan (5)[18].

$$r(t, l) = 1 - \frac{\left| \frac{\bar{v}(t)}{\text{Max}_c V_c(t)} - \frac{\bar{v}(t-l)}{\text{Max}_c V_c(t-l)} \right|}{\sqrt{12}} \quad (5)$$

Dimana l ($0 \leq l \leq t$) merupakan *delay*. Penyebut $\sqrt{12}$ merupakan panjang dari garis diagonal dari 12 *hypercube* dimensi dengan panjang tepi 1, $r(t, l)$ yang memenuhi $0 \leq r(t, l) \leq 1$. Tahapan *Calculate Similarity* ditampilkan dalam *pseudocode* di bawah.

```
calculate_similarity
broadcast_x ← numpy.expand_dims(chroma, 2)
broadcast_y ← numpy.swapaxes(
    numpy.expand_dims(
        chroma, 2), 1, 2)
time_time_matrix ← 1 - (numpy.linalg.norm(
    (broadcast_x - broadcast_y), 0) / sqrt(12))
return time_time_matrix
```

2.9. List Repeated Sections

Bagian-bagian berulang dibuat ke dalam suatu *list* yang nantinya akan diintegrasikan untuk mencari *chorus*. Perulangan tersebut di dapatkan dari $r(t, l)$ yang didapatkan dari langkah sebelumnya. Kemungkinan kemiripan segmen baris yang dievaluasikan pada waktu saat ini dapat dihitung dengan persamaan (6)[18].

$$R_{all}(t, l) = \frac{1}{t-l+1} \sum_{\tau=l}^t r(\tau, l) \quad (6)$$

Selanjutnya, tiap $R_{all}(t, l)$ akan dicari diferensiasinya dengan tujuan untuk menghilangkan *noise* dengan persamaan (7)[18].

$$\sum_{\omega=-K_{size}}^{K_{size}} \omega R_{all}(t, l + \omega) \quad (7)$$

Pada tahapan ini merupakan representasi untuk menghitung pada tahapan selanjutnya, yaitu perhitungan menggunakan matriks.

2.10. Integrate Repeated Sections

Karena setiap ruas baris menunjukkan hanya sepasang bagian yang berulang, maka perlu untuk mengatur ke dalam kelompok segmen garis yang memiliki bagian yang sama. Pada langkah ini, juga dilakukan pendeteksian ulang untuk mencari segmen yang hilang atau tersembunyi dari baris yang berulang. Pada tahap ini, pencarian dilakukan dengan persamaan (8)[18].

$$R_{[Tsi, Tei]}(l) = \frac{1}{Tei - Tsi + 1} \sum_{\tau=Tsi}^{Tei} r(\tau, l) \quad (8)$$

Lalu untuk mendapatkan detail, dibutuhkan perhitungan dengan persamaan (9)[18].

$$\sum_{w=-K'_{size}}^{K'_{size}} \omega R_{[Tei, Tsi]}(l + \omega) \quad (9)$$

Tahapan *Integrate Repeated Sections* ditampilkan dalam *pseudocode* di bawah.

```

integrate_repeated_sections
num_samples ← chroma.shape[1]
broadcast_x ← numpy.repeat
                (numpy.expand_dims
                 (chroma,2),num_samples+1, 2)
circulant_y ← numpy.tile
                (chroma, (1,num_samples
                +1)).reshape(12, num_samples,
                num_samples + 1)
time_lag_similarity ← 1 - (numpy.linalg
                .norm((broadcast_x -
                circulant_y),0)/
                sqrt(12))
time_lag_similarity ← numpy.rot
                90(time_lag_similarity,
                y,1,(0, 1))
return time_lag_similarity
[:num_samples,:num_samples]
    
```

2.11. Integrate Repeated Sections With Modulation

Bagian yang sudah diintegrasikan sebelumnya akan dideteksi kembali untuk mendapatkan perubahan kunci nada. Modulasi akan dilakukan pada matriks S 12x12 (10) berikut.

$$S = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 0 \end{pmatrix} \quad (10)$$

Perhitungan yang digunakan pada tiap isi matriks tersebut dapat dilihat pada persamaan (11)[18].

$$r_{\zeta}(t, l) = 1 - \frac{s^{\zeta} \vec{v}(t) \cdot \vec{v}(t-l)}{\text{Max}_{c} v_c(t) \cdot \text{Max}_{c} v_c(t-l)} \quad (11)$$

Tahapan *Integrate Repeated Sections With Modulation* ditampilkan dalam *pseudocode* di bawah.

```

integrate_repeated_sections_with_modulation
input :
    denoised_time_lag, rows, threshold,
    min_length_samples
num_samples ← denoised_time_lag.shape[0]
line_segments ← []
cur_segment_start ← None
for each row ∈ rows
do IF row < min_length_samples
for each col ∈ range(row, num_samples)
do IF denoised_time_lag
    [row, col]>threshold
IF cur_segment_start is None
    cur_segment_start ← col
END IF
else
IF (cur_segment_start is not
    None) and
    (col-cur_segment_start)>
    min_length_samples
    line_segments.append(Line
    (cur_segment_start, col, row))
END IF
    cur_segment_start ← None
END IF
return line_segments
    
```

2.12. Select Chorus Sections

Langkah terakhir, memilih bagian mana yang merupakan *chorus* dari hasil pemisahan sebelumnya. Ukuran *chorus* didapatkan dengan persamaan (12)[18].

$$v_i = \left(\sum_{j=1}^{M_i+1} \lambda_{ij} \right) \log \frac{Tei-Tsi+1}{D_{len}} \quad (12)$$

Kelompok dengan ukuran v_i terbesar akan dipilih sebagai *chorus*. Tahapan *Select Chorus Sections* ditampilkan dalam *pseudocode* di bawah.

```

select_chorus_sections
lines_to_sort ← []
for each line ∈ line_scores
do lines_to_sort.append((line,line_scores
    [line], line.end - line.start))
lines_to_sort.sort(key=lambda x: (x[1],
x[2]), reverse=True)
best_tuple ← lines_to_sort[0]
return best_tuple[0]
    
```

Keterangan seluruh simbol dapat dilihat pada tabel 5[18].

Tabel 5. Keterangan Simbol

Simbol	Keterangan
t	Waktu (langkah waktu diskrit 80 ms)
$\psi_p(f, t)$	Spektrum magnitudo dalam frekuensi berskala $\log f$
$\vec{v}(t)$	12 dimensi <i>chroma vector</i>
$v_c(t)$	Elemen dari $\vec{v}(t)$ ($c=1,2,\dots,12$)
$r(t, l)$	Kesamaan antara $\vec{v}(t)$ dan $\vec{v}(t-l)$
$[T1, T2]$	Bagian diantara waktu T1 dan T2
$R_{all}(t, l)$	Peluang yang mengandung segmen baris pada lag l
$R[Tsi, Tei]^{(l)}$	Kemungkinan mengandung segmen garis di lag l dalam $[Tsi, Tei]$
ζ	Perbedaan nada dari perubahan kunci ($\zeta = 0, 1, \dots, 11$)
S	12 ke 12 Pergeseran matriks
$R\zeta(t, l)$	Kesamaan antara $S^{\zeta} \vec{v}(t)$ dan $\vec{v}(t-l)$
$[Ps_{ij}, Pe_{ij}]$	Membuka bagian berulang yang sesuai dengan γ_{ij}
$\times ij$	Kemungkinan bagian $[Tsi, Tei]$ dan $[Ps_{ij}, Pe_{ij}]$ diulang
v_i	Ukuran <i>chorus</i> (kemungkinan menjadi bagian <i>chorus</i> pada setiap grup i)
K_{size}	4
K'_{size}	35
D_{len}	1.4

3. HASIL DAN PEMBAHASAN

Pada bagian ini diuraikan hasil dari penelitian beserta pengujian yang telah dilakukan. Selain itu, disampaikan juga mengenai pembahasan dari penelitian maupun pengujian yang telah dilakukan.

3.1. Data

Dalam penulisan ini, penulis menggunakan data musik berupa *File* wav yang merupakan hasil konversi dari video musik yang diunduh pada situs YouTube. Data yang didapatkan berupa *File* sebanyak 40 musik, terbagi atas 20 musik dengan genre dangdut Indonesia dan 20 musik dengan genre

pop Indonesia. Data ini akan di proses menggunakan *Refrain Detecting Method* untuk dicari bagian *chorus* nantinya. Genre musik dapat dilihat dalam tabel 6.

Tabel 6. Analisa Genre

No.	Genre	Jumlah
1	Dangdut	20
2	POP	20

3.2. Implementasi RefraiD

Pada pengimplementasian RefraiD terdapat enam tahapan seperti yang sudah dijelaskan pada metode penelitian. Pada bagian ini penulis menggunakan contoh musik Via Vallen – Sayang untuk dihitung menggunakan rumus sesuai tahapan RefraiD.

1. Extract Acoustic Feature

Hasil perhitungan *Extract Acoustic Feature* dapat dilihat pada tabel 7.

Tabel 7. *Extract Acoustic Feature*

0	1	1678
0,664829	0,640658	0
0,633532	0,647052	0
0,664716	0,794877	0
1	1	0
0,549177	0,547655	0
0,228103	0,282975	0
0,340828	0,306389	0
0,425789	0,377329	0
0,347734	0,348459	0
0,320815	0,360615	0
0,475339	0,429344	0
0,586962	0,518503	0

2. Calculate Similarity

Hasil perhitungan *Calculate Similarity* dapat dilihat pada tabel 8.

Tabel 8. *Calculate Similarity*

0	1	1678
1	0,947844	0,44243
0,947844	1	0,438614
0,925535	0,968569	0,419273
0,921465	0,968219	0,431145
0,533591	0,537023	0,684024
....
0,44243	0,438614	1
0,44243	0,438614	1
0,44243	0,438614	1
0,44243	0,438614	1

3. List Repeated Sections

Hasil perhitungan *List Repeated Sections* dapat dilihat pada tabel 9.

Tabel 9. *List Repeated Sections*

0	1	...	1676	1677
1	1	...	1	1
0,44243031	0,94784361	...	1	1
7	1	...	1	1
0,44243031	0,43861401	...	1	1
7	1	...	1	1
.....
0,92146486	0,53702259	...	0,43861401	0,41927272
	1	...	1	1
0,92553496	0,96821880	...	0,44243031	0,43861401
4	3	...	7	1
0,94784361	0,96856874	...	1	0,44243031
1	2	...	1	7

4. Integrate Repeated Sections

Hasil perhitungan *Integrate Repeated Sections* dapat dilihat pada tabel 10.

Tabel 10. *Integrate Repeated Sections*

0	1	1677	1678
0	0	0	0
0	0	0	0
0	0	0	0
.....
0	0	0,072549	0,072972
0	0	0,051467	0,05177
0	0	0,026956	0,027116

5. Integrate Repeated Sections With Modulation

Hasil perhitungan *Integrate Repeated Sections With Modulation* dapat dilihat pada tabel 11.

Tabel 11. *Integrate Repeated Sections With Modulation*

0
Line (517 658 443)
Line (822 1105 443)
Line (1260 1536 443)
Line (961 1108 886)
Line (1254 1537 886)
Line (1401 1533 1329)

6. Select Chorus Sections

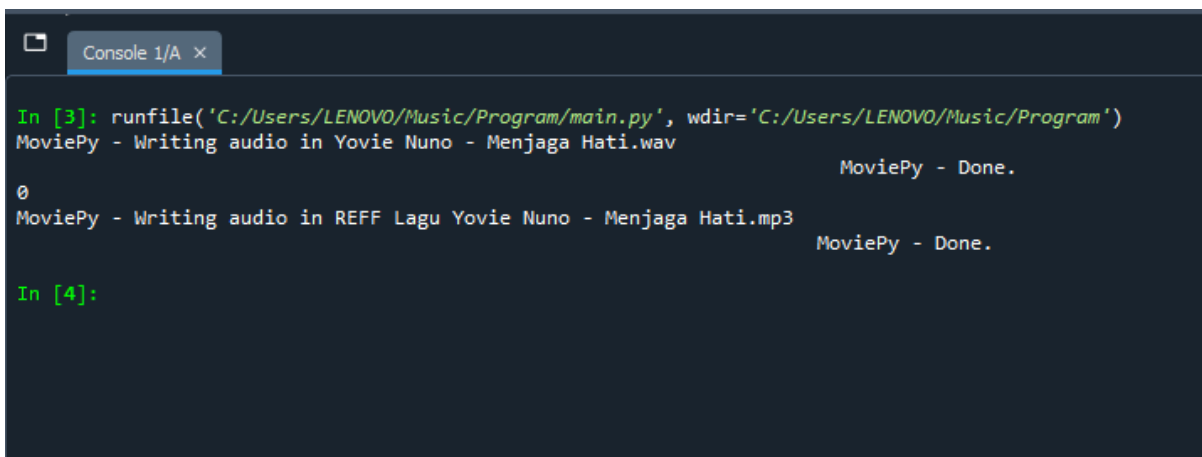
Hasil perhitungan *Select Chorus Sections* dapat dilihat pada tabel 12.

Tabel 12. *Select Chorus Sections*

Line (1401 1533 1329)

3.3. Implementasi Sistem

Proses pendeteksian *chorus* yang diimplementasikan menggunakan Python, pada metode *Refrain Detecting Method* dapat dilihat pada gambar 2.



Gambar 2. Hasil Run Sistem

3.4. Pengujian F Measure

Umumnya untuk mengukur ketepatan akurasi pendeteksian menggunakan proporsi frekuensi yang tepat diklasifikasikan dengan total sampel yang ada[19]. Pada penelitian ini pendeteksian *chorus* digunakan proporsi waktu chorus musik yang tepat dideteksi (PTCTB) dengan total chorus musik yang benar (PTCB). Akurasi pada penelitian ini menggunakan *F Measure*. *F Measure* merupakan tahapan untuk menentukan kinerja penentuan *chorus* menggunakan *Refrain Detecting Method* dapat dilihat pada persamaan (13) dimana sebelumnya akan dilakukan perhitungan nilai *Precision* menggunakan persamaan (14) serta dilakukan perhitungan *Recall* menggunakan persamaan (15).

$$F Measure = \frac{2RP}{R+P} \tag{13}$$

$$P = \frac{PTCTB}{PTCT} \tag{14}$$

$$R = \frac{PTCTB}{PTCB} \tag{15}$$

Persamaan (13),(14) dan persamaan (15) dijelaskan dengan keterangan pada tabel 13.

Tabel 13. Keterangan Persamaan

Singkatan	Keterangan
PTCTB	Panjang total bagian chorus yang terdeteksi dengan benar
PTCB	Panjang total bagian chorus yang benar
PTCT	Panjang total bagian chorus yang terdeteksi

3.5. Pencarian Chorus Pada Musik Dangdut

Hasil pencarian *chorus* , *F Measure* , *Precision* dan *Recall* pada musik dangdut menggunakan *Refrain Detecting Method* dapat dilihat pada tabel 14.

Tabel 14. Hasil Pencarian Chorus Pada Musik Dangdut

No.	Nama File	PTCTB (Detik)	PTCT (Detik)	PTCB (Detik)	R	P	F Measure
1	Ayu Ting Ting - Sambalado.wav	43	54	45	0,955556	0,796296	0,868687
2	Bebizy - Cinta Tulalit.wav	47	64	47	1	0,734375	0,846847
3	Camel Petir - CKC Cuma Kamu Cin.wav	39	77	39	1	0,506494	0,672414
4	Duo Anggrek - Cikini Gondangdia.wav	26	26	30	0,866667	1	0,928571
5	Duo Serigala - Abang Goda.wav	26	27	26	1	0,962963	0,981132
6	Meggi Z - Benang Biru.wav	120	120	120	1	1	1
7	Meggi Z - Cinta Hitam.wav	149	149	149	1	1	1
8	Meggi Z - Mata Air Cinta.wav	101	101	105	0,961905	1	0,980583
9	Nella Kharisma - Cie Cie.wav	54	55	58	0,931034	0,981818	0,955752
10	Nella Kharisma - Sakit Dalam Bercinta.wav	47	47	48	0,979167	1	0,989474
11	Regina - Cinta Basi .wav	38	38	39	0,974359	1	0,987013
12	Rhoma Irama - Bujangan.wav	22	24	23	0,956522	0,916667	0,93617
13	Rhoma Irama - Judi.wav	29	29	32	0,90625	1	0,95082
14	Sakitnya Tuh Disini - Cita Citata.wav	45	62	45	1	0,725806	0,841121
15	Siti Badriah - Suamiku Kawin Lagi.wav	43	46	43	1	0,934783	0,966292
16	Ucie Sucita - Cinta Tak Terbatas Waktu.wav	26	27	26	1	0,962963	0,981132
17	Via Vallen - 5 Centi.wav	22	22	51	0,431373	1	0,60274
18	Via Vallen - Cinta Kurang Gizi.wav	70	73	70	1	0,958904	0,979021
19	Via Vallen - Sayang.wav	25	25	28	0,892857	1	0,943396
20	Zaskia Gotik - Sudah Cukup Sudah.wav	26	28	26	1	0,928571	0,962963
RATA RATA					0,942784	0,920482	0,918706

3.6. Pencarian Chorus Pada Musik Pop

Hasil pencarian *chorus*, *F Measure*, *Precision* dan *Recall* pada musik pop menggunakan *Refrain Detecting Method* dapat dilihat pada tabel 15.

Tabel 15. Hasil Pencarian Chorus Pada Musik Pop

No.	Nama File	PTCTB (Detik)	PTCT (Detik)	PTCB (Detik)	R	P	F Measure
1	Andmesh - Cinta Luar Biasa.wav	33	34	34	0,970588	0,970588	0,970588
2	Ari Lasso - Arti Cinta.wav	33	34	33	1	0,970588	0,985075
3	Armada - Apa Kabar Sayang.wav	41	42	41	1	0,97619	0,987952
4	Anneth - Mungkin Hari Ini Esok Atau Nanti.wav	33	34	33	1	0,970588	0,985075
5	Armada - Harusnya Aku.wav	42	42	42	1	1	1
6	Bondan - Ya Sudahlah.wav	20	21	21	0,952381	0,952381	0,952381
7	Cassandra - Cinta Terbaik.wav	25	25	25	1	1	1
8	Geisha - Cinta Dan Benci.wav	27	27	33	0,818182	1	0,9
9	Geisha - Lumpuhkan Ingatanmu.wav	0	24	34	0	0	0
10	Geisha - Pergi Saja.wav	27	27	27	1	1	1
11	Hijau Daun - Suara Ku Berharap.wav	20	23	20	1	0,869565	0,930233
12	Irwansyah - Pecinta Wanita.wav	25	26	25	1	0,961538	0,980392
13	Tulus - Hati Hati di Jalan.wav	23	24	23	1	0,958333	0,978723
14	Noah - Tak Lagi Sama.wav	20	21	20	1	0,952381	0,97561
15	Peterpan - Ada Apa Denganmu.wav	57	58	57	1	0,982759	0,991304
16	Peterpan - Menghapus Jejakmu.wav	13	25	13	1	0,52	0,684211
17	Ungu - Disini Untukmu.wav	23	23	24	0,958333	1	0,978723
18	Vageto - Kehadiranmu.wav	36	36	36	1	1	1
19	Tiara Andini - Merasa Indah.wav	35	35	35	1	1	1
20	Yovie Nuno - Menjaga Hati.wav	27	27	27	1	1	1
RATA RATA					0,934974	0,904246	0,915013

4. KESIMPULAN

RefrainD dapat menentukan chorus secara otomatis pada musik dangdut dan pop Indonesia dengan akurasi yang baik yakni berdasarkan pengujian *F measure* diperoleh 91.8% pada musik dangdut dan pada musik pop sebesar 91.5%. Pada musik dangdut, metode RefrainD memiliki akurasi lebih baik dalam menentukan chorus.

DAFTAR PUSTAKA

- [1] G. Agustiniingsih, "Konstruksi Gaya Hidup Melalui Musik sebagai Produk Budaya Populer," *J. Komun. dan Bisnis*, vol. 6, no. 2, pp. 16–22, 2018.
- [2] I. P. Wulandari, "Jatung Utang Sebagai Pengiring Tari Hudog Pada Masyarakat Suku Dayak Kenyah di Desa Sungai Payang Kecamatan Loa Kulu Kabupaten Kutai Kartanegara Kalimantan Timur," *Selonding*, vol. 12, no. 12, pp. 1824–1839, 2017, doi: 10.24821/selonding.v12i12.2931.
- [3] K. Xu, "Formation and Training of Mass Chorus Team," *Int. Core J. Eng.*, vol. 7, p. 2021, doi: 10.6919/ICJE.202108_7(8).0026.
- [4] Y. Huang, S. Chou, and Y. Yang, "Pop Music Highlighter: Marking the Emotion Keypoints," vol. 1, pp. 68–78, 2018.
- [5] D. Octavia Kumalasari, L. Novamizanti, and I. Nyoman Apraz Ramatryana, "Penentuan lokasi chorus pada musik mp3 menggunakan

koefisien korelasi 2-d pada frame berbasis ciri mel-frequency cepstral coefficient (mfcc)," 2019.

- [6] N. Kroher and J.-M. Díaz-Báñez, "Flamenco Music Information Retrieval Automatic Content-Based Description of Flamenco Music Collections," 2018.
- [7] S. Kurniawan and S. Agustian, "Music Information Retrieval Menggunakan k-NN dan Cosine Similarity," *Semin. Nas. Teknol. Informasi, Komun. dan Ind.* 13, *Fak. Sains dan Teknol. UIN Sultan Syarif Kasim Riau*, no. November, pp. 94–101, 2021.
- [8] and Y. W. Ju-Chiang Wang, Jordan B.L. Smith, Jitong Chen, Xuchen Song, "Supervised Chorus Detection For Popular Music Using Convolutional Neural Network And Multi-Task Learning," pp. 566–570, 2021.
- [9] K. Watanabe and M. Goto, "A Chorus-Section Detection Method For Lyrics Text," 2020. [Online]. Available: <http://www.speech.cs.cmu.edu/cgi-bin/cmudict>.
- [10] M. Wahyu Setiawan, L. Novamizanti, and I. N. Apraz Ramatryana, "Pemisahan Chorus Pada Music Mp3 Menggunakan Koefisien Korelasi 2-D Berbasis Discrete Cosine Transform (Dct) Dan K-Nearest Neighbor (K-Nn) Separation Of Chorus On Mp3 Music Using 2-D Correlation Coefficient Based On Discrete Cosine Transform (Dct)

- And K-Nearest Neighbor (K-Nn),” 2019.
- [11] B. Adam, I. R. Magdalena, I. Nyoman, and A. Ramatryana, “Perancangan Dan Simulasi Pemisahan Reff Lagu Dengan Metode Discrete Cosine Transform (Dct) Design And Simulation Of Separating The Chorus Songs By Using Discrete Cosine Transform (Dct),” 2018.
- [12] N. Ono, S. Tsuchiya, S. Nakamura, and T. Yamamoto, “A Study of the Relationships between Music-impression, Visual-impression and Music Video Clip’s Impression,” 2018. [Online]. Available: <http://www.music-ir.org/mirex/wiki/MIREX>.
- [13] B. Saini, V. Singh, and S. Kumar, “Information Retrieval Models and Searching Methodologies : Survey,” *Int. J. Adv. Found. Res. Sci. Eng.*, vol. 1, no. 2, pp. 57–62, 2014.
- [14] IngwersenPeter and JärvelinKalervo, “Information retrieval in context,” *ACM SIGIR Forum*, vol. 39, no. 2, pp. 31–39, Dec. 2005, doi: 10.1145/1113343.1113351.
- [15] D. Stoller and P. Thesis, “Deep Learning for Music Information Retrieval in Limited Data Scenarios,” 2020.
- [16] I. G. Harsemadi, “Perbandingan Distance Measure Pada K-Means Clustering Untuk Pengelompokkan Musik Terhadap Suasana Hati,” *Univ. AMIKOM Yogyakarta*, 2018, [Online]. Available: www.audionetwork.com.
- [17] I Komang Setia Buana, “Aplikasi Untuk Pengoprasian Komputer Dengan Mendeteksi Gerakan Menggunakan Opencv Python,” 2018.
- [18] M. Goto, “A chorus section detection method for musical audio signals and its application to a music listening station,” *IEEE Trans. Audio, Speech Lang. Process.*, vol. 14, no. 5, pp. 1783–1794, Sep. 2006, doi: 10.1109/TSA.2005.863204.
- [19] P. R. Sihombing and A. M. Arsani, “Perbandingan Metode Machine Learning Dalam Klasifikasi Kemiskinan Di Indonesia Tahun 2018,” *J. Tek. Inform. (JUTIF)*, Vol. 2, No. 1, Juni 2021, hlm. 51-56, vol. 2, no. 1, pp. 51–56, 2021.