

---

## ANALISIS DAN IDENTIFIKASI GENRE MUSIK MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE*

Innosensius Julian<sup>1)</sup>, Nahumi Nugrahaningsih<sup>2)</sup>, Ariesta Lestari<sup>3)</sup>, Licantik<sup>4)</sup>, Felicia Sylviana<sup>5)</sup>

<sup>1)2)3)</sup> Jurusan Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya  
Jalan Hendrik Timang Kampus Tunjung Nyaho, Palangka Raya

<sup>1)</sup> innosensiusjulian@gmail.com

<sup>2)</sup> nahuminugrahaningsih@it.upr.ac.id

<sup>3)</sup> ariestalestari@it.upr.ac.id

<sup>4)</sup> licantik@it.upr.ac.id

<sup>5)</sup> feliciasylviana@it.upr.ac.id

### Abstrak

Penelitian ini bertujuan untuk mengembangkan metode identifikasi genre musik dalam musik Indonesia menggunakan Mel-frequency cepstral coefficients (MFCC) sebagai fitur akustik. Sebanyak 400 sampel musik dari berbagai genre digunakan dalam studi ini. Analisis menggunakan Confusion Matrix untuk mengukur akurasi prediksi genre musik berdasarkan model yang dikembangkan. Hasil pengujian menunjukkan nilai precision yang signifikan untuk setiap genre: blues (84%), classical (88%), jazz (71%), dan pop (94%). Selain itu, recall untuk setiap genre juga menunjukkan performa yang baik: blues (81%), classical (83%), jazz (83%), pop (89%). F1-score, yang menggabungkan precision dan recall, menunjukkan kinerja model yang seimbang: blues (82%), classical (86%), jazz (77%), pop (91%). Keseluruhan, model yang dikembangkan mencapai akurasi sebesar 84%, menegaskan efektivitas metode MFCC dalam mengidentifikasi genre musik. Penelitian ini menyoroti pentingnya penggunaan dataset yang representatif untuk meningkatkan akurasi identifikasi genre musik dalam konteks musik Indonesia.

**Kata kunci:** MFCC (Mel-frequency cepstral coefficients), SVM (Support Vector Machine), Identifikasi Musik

### Abstract

*This study aims to develop a method for identifying music genres in Indonesian music using Mel-frequency cepstral coefficients (MFCC) as acoustic features. A total of 400 music samples from various genres were utilized in this study. The analysis employed Confusion Matrix to measure the accuracy of music genre predictions based on the developed model. The test results revealed significant precision values for each genre: blues (84%), classical (88%), jazz (71%), and pop (94%). Additionally, the recall for each genre also demonstrated good performance: blues (81%), classical (83%), jazz (83%), pop (89%). The F1-score, combining precision and recall, indicated a balanced model performance: blues (82%), classical (86%), jazz (77%), pop (91%). Overall, the developed model achieved an accuracy of 84%, confirming the effectiveness of the MFCC method in music genre identification. This research highlights the importance of using representative datasets to enhance the accuracy of music genre identification within the context of Indonesian music..*

**Keywords:** MFCC (Mel-Frequency cepstral coefficients), SVM (Support Vector Machine), Music Identification

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Klasifikasi genre musik merupakan tantangan tersendiri. Kompleksitasnya terletak pada subjektivitas persepsi manusia terhadap musik dan keragaman elemen yang membentuk sebuah genre. Fitur musik seperti tempo, irama, melodi, dan harmoni memainkan peran penting dalam penentuan genre. Oleh karena itu, pendekatan otomatis menggunakan teknologi pembelajaran mesin, seperti Support Vector Machine (SVM), menjadi sangat relevan. SVM, yang merupakan salah satu algoritma pembelajaran mesin yang paling efisien dan populer, telah terbukti efektif dalam berbagai tugas klasifikasi. Dengan kemampuannya untuk menangani data berdimensi tinggi dan menemukan hyperplane optimal dalam ruang fitur, SVM dapat menjadi alat yang kuat untuk klasifikasi genre musik. Namun, penerapan SVM dalam konteks ini masih memerlukan eksplorasi dan penelitian lebih lanjut, terutama dalam hal pemilihan fitur yang efektif dan penanganan subtleties dari berbagai genre musik. Mengingat pentingnya dan tantangan yang ada dalam klasifikasi genre musik, penelitian ini bertujuan untuk mengembangkan model klasifikasi genre musik dengan menggunakan algoritma SVM. Penelitian ini akan fokus pada ekstraksi dan seleksi fitur musik yang efektif, implementasi model SVM, dan evaluasi kinerjanya dalam mengklasifikasikan berbagai genre musik. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi pada peningkatan teknologi pengenalan dan klasifikasi genre musik, serta mendukung pengembangan aplikasi yang berhubungan dengan musik dalam dunia digital.

### 1.2 Tujuan Penelitian

Tujuan dari analisis dan identifikasi ini adalah mengidentifikasi dan menganalisis fitur – fitur music apa saja yang memiliki pengaruh signifikan dalam proses klasifikasi menggunakan SVM.

## 2. TINJAUAN PUSTAKA

Tinjauan pustaka digunakan sebagai pembandingan dan acuan untuk pengembangan sistem . Penelitian ini menggunakan beberapa sumber pustaka yang berhubungan dengan kasus yang akan diteliti. Penelitian yang pertama yaitu penelitian Yuwono et al [1], yaitu “Music Genre Classification Using Support Vector Machine Techniques”. Penelitian ini mengeksplorasi efektivitas SVM dalam mengelola kompleksitas data musik berdimensi tinggi. Mereka menunjukkan bagaimana SVM mampu membedakan genre musik dengan tingkat akurasi yang tinggi, memanfaatkan fitur audio yang beragam.

Penelitian oleh Narkhede, Mathur dan Bhaskar (2022)[2], yaitu “Automatic Classification of Music Genre Using SVM”. Tujuan dari penelitian ini adalah membuat sebuah program aplikasi yang mampu menampung pengaduan secara online, mencatat kemajuan penanganan, mencatat dan menampilkan tanggapan, serta memudahkan administrator dalam membuat laporan. Metodologi yang digunakan pada penelitian ini adalah dengan metode Software Development Life Cycle (SDLC) model waterfall, dengan tahapan perencanaan , analisis , perancangan , implementasi, dan tahap pemeliharaan. Terdapat 2 Orang Aktor pada pembuatan website ini yaitu Mahasiswa, dan Admin mengambil langkah lebih lanjut dengan mengintegrasikan SVM ke dalam sistem klasifikasi otomatis. Penelitian ini menyoroti potensi teknologi dalam menyederhanakan proses ekstraksi fitur, pra-pemrosesan, dan pelatihan model, memperlihatkan bagaimana otomatisasi dapat meningkatkan efisiensi dan kecepatan proses klasifikasi.

Penelitian oleh Setiadi et al, [3], yaitu “ Effect of Feature Selection on The Accuracy of Music Genre Classification using SVM Classifier ”, menyoroti pentingnya pemilihan fitur yang tepat dalam meningkatkan akurasi klasifikasi. Mereka menunjukkan bahwa pemilihan fitur spektral dan temporal yang tepat sangat krusial dalam meningkatkan performa model SVM, menandakan bahwa tidak hanya algoritma itu sendiri, tetapi juga bagaimana fitur disiapkan dan dipilih yang memainkan peran kunci dalam klasifikasi yang efektif.

### 3. METODE PENELITIAN

Metodologi penelitian yang digunakan dalam penelitian ini adalah pengumpulan data, pra-pemrosesan data, implementasi SVM, evaluasi model, analisis hasil.

#### 1. Pengumpulan Data

Pengumpulan data akan dilakukan dengan mencari melalui online resource seperti Google, Youtube, Spotify, dan platform yang menyediakan jasa streaming musik. Lagu yang akan dikumpulkan merupakan data lagu dari berbagai genre.

#### 2. Pra-Pemrosesan Data

Dalam tahap Operation and Maintenance ini, sistem diinstal dan mulai digunakan. Selain itu juga memperbaiki error yang tidak ditemukan pada tahap pembuatan. Dalam tahap ini juga dilakukan pengembangan sistem seperti penambahan fitur dan fungsi baru.

#### 3. Implementasi SVM

Pemilihan kernel SVM: Memilih kernel yang tepat (linear, polynomial, RBF, dll.). Pelatihan model: Cara model SVM akan dilatih dengan data yang ada. Validasi silang: Menggunakan teknik validasi silang untuk menguji efektivitas model

#### 4. Evaluasi Model

Pada tahap ini akan dilakukan training model, yaitu dengan menggunakan confusion matrix seperti pengujian akurasi, presisi, recall, dan F1-Score.

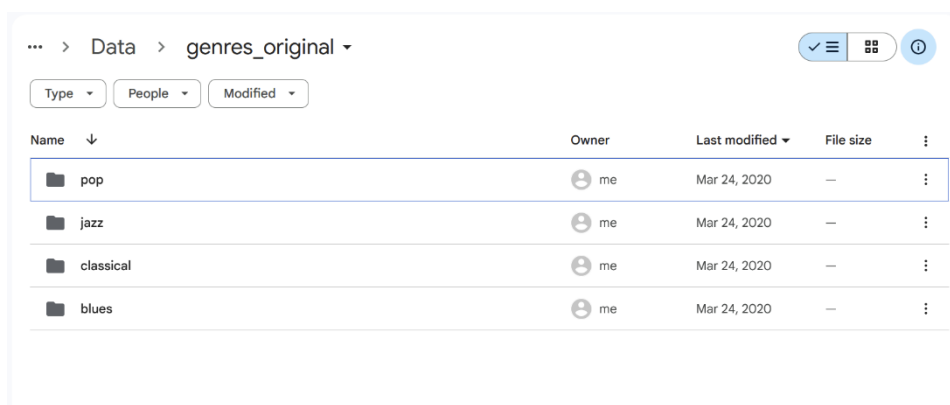
#### 5. Analisis Hasil

Pada tahap ini hasil dari evaluasi model akan dianalisis untuk mendapatkan fitur-fitur apa saja yang mempengaruhi dan akan dilakukan perbandingan dengan Teknik klasifikasi lain jika ada.

### 4. PEMBAHASAN DAN IMPLEMENTASI

#### 4.1. Pengumpulan Dataset

Pengumpulan dataset dilakukan dengan cara mendownload lagu dan genre, lalu memotongnya secara manual agar durasinya sama yaitu sepanjang 30 detik dalam format WAV, selain itu penelitian dan pengkodean menggunakan google collab sehingga dataset dimasukkan kedalam google drive agar dapat diakses oleh google colab tanpa harus mengunggahnya terlebih dahulu. Seperti pada gambar 4.1.



Gambar 1 Folder Dataset di dalam GDrive

#### 4.2. Visualisasi Data

Setelah proses pengumpulan dataset selesai dilakukan dan data tersimpan di Google Drive, langkah selanjutnya dalam penelitian adalah melakukan visualisasi data. Visualisasi data merupakan tahap penting dalam penelitian untuk menggambarkan informasi secara grafis guna memudahkan pemahaman dan analisis data. Setiap genre musik diproses untuk diambil waveforms-nya dan informasi lainnya seperti sample rate.

```

Processing folder: blues
File: blues.00051.wav
y: [-0.21359253 -0.35723877 -0.31430054 ... -0.0574646 -0.01867676
-0.01022339]

y shape: (661794,)

Sample Rate (KHz): 22050

Processing folder: classical
File: classical.00019.wav
y: [-0.00057983 -0.0017395 -0.00259399 ... -0.00448608 -0.00271606
-0.00289917]

y shape: (661794,)

Sample Rate (KHz): 22050

Processing folder: jazz
File: jazz.00026.wav
y: [ 0.00198364 0.00143433 -0.00170898 ... 0.03482056 0.03631592
0.0345459 ]

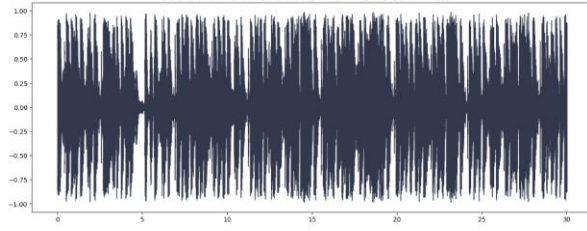
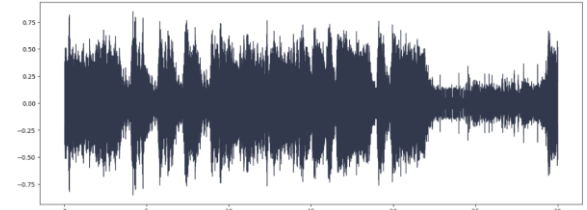
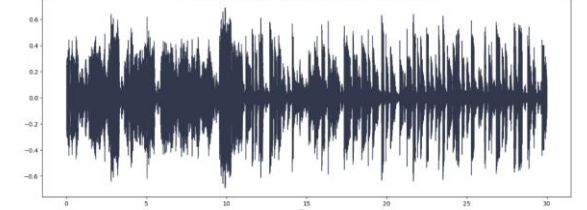
y shape: (661794,)

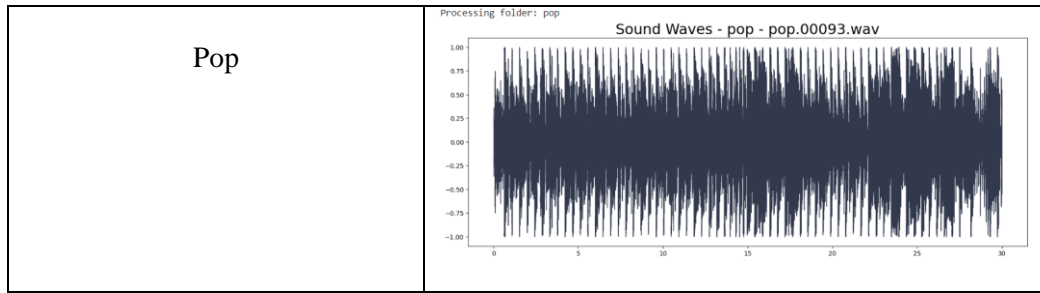
Sample Rate (KHz): 22050

Processing folder: pop
File: pop.00023.wav
y: [-0.00378418 0.04263306 0.04504395 ... -0.31472778 -0.2567749
-0.3123474 ]
    
```

Gambar 2 Sample *Waveform* setiap sample audio pada setiap genre

Tabel 1 Representasi *Waveform* di setiap genre

AUDIO	Representasi
Blues	
Classical	
Jazz	

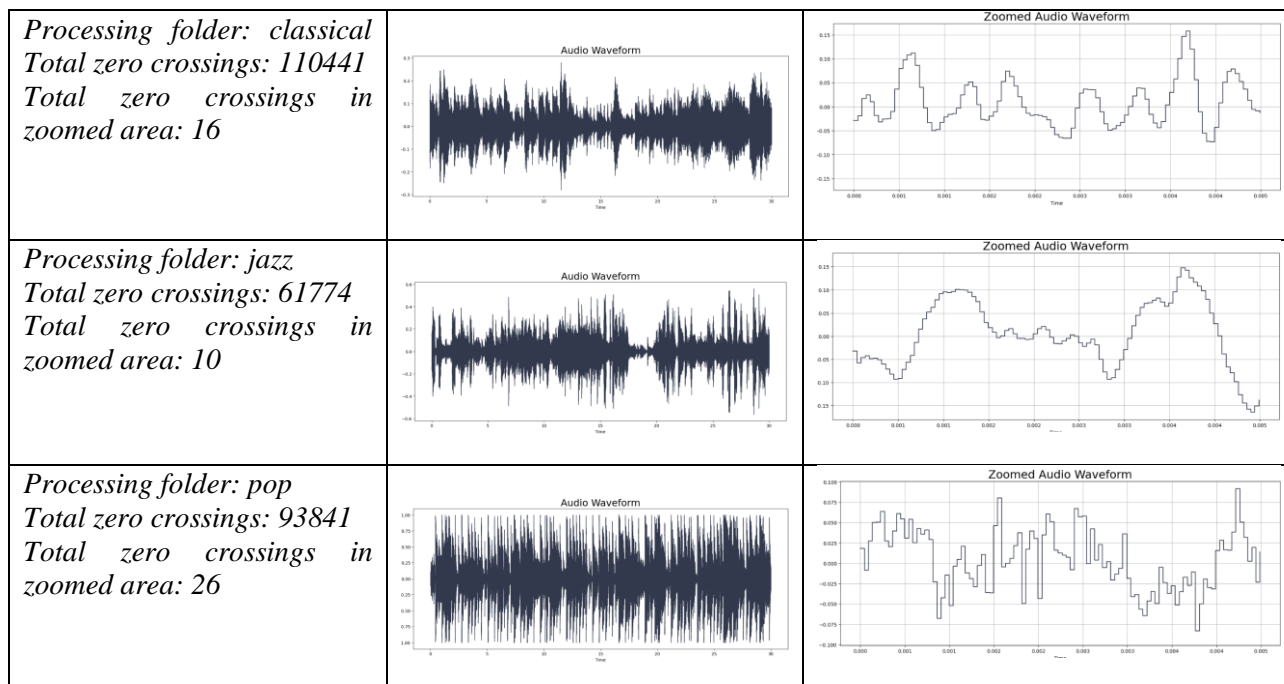


Tabel 2 Konversi Spectrogram ke Mel Spectrogram

Audio	Spectrogram	Mel Spectrogram
Blues		
Classical		
Jazz		
Pop		

Tabel 3 Ekstraksi Sinyal Audio

Audio	Audio Waveform	Zoomed Audio Waveform
<p>Processing folder : blues Total zero crossings: 77660 Total zero crossings in zoomed area: 21</p>		



### 4.3. Hitung Matriks Evaluasi Model

```

1. # Menampilkan classification report (akurasi, precision,
   recall, dan F1-score)
2. print("Classification Report:")
3. print(classification_report(y_test, y_pred))
SVC(C=0.1, gamma='auto', kernel='poly', random_state=42)
Type data fitur: float64
Dimensi data fitur: (320, 13)
Classification Report:
           precision    recall  f1-score   support

 blues          0.84      0.81      0.82         26
classical       0.88      0.83      0.86         18
 jazz           0.71      0.83      0.77         18
 pop            0.94      0.89      0.91         18

 accuracy                   0.84         80
macro avg          0.84      0.84      0.84         80
weighted avg       0.84      0.84      0.84         80

```

Berikut adalah hasil dari evaluasi model SVC (*Support Vector Classifier*) yang telah dilatih. Berikut adalah penjelasan untuk setiap bagian dari *classification report*:

1. **Precision:**
  - a) Precision merupakan proporsi dari true positive (positif yang diprediksi dengan benar) dari semua instance yang diprediksi positif oleh model.
  - b) Precision untuk masing-masing kelas adalah sebagai berikut:
    1. Blues: 0.84 (84% dari instance yang diprediksi sebagai blues adalah benar)
    2. Classical: 0.88 (88% dari instance yang diprediksi sebagai classical adalah benar)
    3. Jazz: 0.71 (71% dari instance yang diprediksi sebagai jazz adalah benar)

4. Pop: 0.94 (94% dari instance yang diprediksi sebagai pop adalah benar)

**2. Recall:**

- a) Recall (atau sensitivity) merupakan proporsi dari true positive dari semua instance yang sebenarnya positif.
- b) Recall untuk masing-masing kelas adalah sebagai berikut:
  - 1. Blues: 0.81 (81% dari instance blues yang sebenarnya teridentifikasi dengan benar)
  - 2. Classical: 0.83 (83% dari instance classical yang sebenarnya teridentifikasi dengan benar)
  - 3. Jazz: 0.83 (83% dari instance jazz yang sebenarnya teridentifikasi dengan benar)
  - 4. Pop: 0.89 (89% dari instance pop yang sebenarnya teridentifikasi dengan benar)

**3. F1-score:**

- a) F1-score adalah harmonic mean dari precision dan recall. Ini memberikan indikasi keseluruhan kualitas dari suatu kelas.
- b) F1-score untuk masing-masing kelas adalah sebagai berikut:
  - 1. Blues: 0.82
  - 2. Classical: 0.86
  - 3. Jazz: 0.77
  - 4. Pop: 0.91

**4. Support:**

- a) Support adalah jumlah instance yang termasuk dalam setiap kelas dalam data pengujian.

**5. Accuracy:**

- a) Akurasi adalah proporsi dari instance yang diprediksi dengan benar dari keseluruhan instance.
- b) Akurasi keseluruhan dari model adalah 0.84 atau 84%.

**6. Macro Average dan Weighted Average:**

- a) Macro average adalah rata-rata dari precision, recall, dan F1-score untuk setiap kelas secara independen.
- b) Weighted average memberi bobot pada rata-rata berdasarkan jumlah instance dalam setiap kelas.

Dengan demikian, classification report ini memberikan gambaran lengkap tentang kinerja model SVC dalam mengklasifikasikan sampel-sampel dari empat kelas yang berbeda.

#### 4.4. Pengujian

Pengujian *system* dilakukan dengan cara mengimport lagu dari Pustaka *local*, lalu diproses sehingga sesuai dengan fitur pelatihan pada SVM.

```
1. # Fungsi untuk menyesuaikan dimensi fitur
2. def adjust_feature_dimension(features, num_features_expected,
3.   num_frames_expected):
4.     # Jika jumlah fitur kurang dari yang diharapkan, tambahkan
5.     # fitur tambahan dengan nilai default
6.     if features.shape[0] < num_features_expected:
7.         num_features_to_add = num_features_expected -
8.         features.shape[0]
9.         features = np.vstack((features,
10.          np.zeros((num_features_to_add, features.shape[1])))
11.         # Jika jumlah fitur lebih dari yang diharapkan, potong fitur
12.         # yang berlebihan
13.         elif features.shape[0] > num_features_expected:
14.             features = features[:num_features_expected, :]
```

```
10.
11.     # Jika jumlah frame kurang dari yang diharapkan, tambahkan
    frame tambahan dengan nilai default
12.     if features.shape[1] < num_frames_expected:
13.         num_frames_to_add = num_frames_expected -
    features.shape[1]
14.         features = np.hstack((features,
    np.zeros((features.shape[0], num_frames_to_add)))
15.     # Jika jumlah frame lebih dari yang diharapkan, potong
    frame yang berlebihan
16.     elif features.shape[1] > num_frames_expected:
17.         features = features[:, :num_frames_expected]
18.
19.     return features
20.
21. # Upload file audio
22. uploaded = files.upload()
23.
24. # Loop melalui file yang diunggah
25. for filename in uploaded.keys():
26.     # Baca file audio
27.     y, sr = librosa.load(filename, sr=None)
28.
29.     # Ekstraksi fitur audio (misalnya, MFCC)
30.     mfccs = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=13)
31.
32.     # Normalisasi fitur jika diperlukan
33.     mfccs_normalized = (mfccs - np.mean(mfccs)) /
    np.std(mfccs)
34.
35.     # Menyesuaikan dimensi fitur agar sesuai dengan model yang
    digunakan
36.     num_features_expected = 13 # Ganti dengan jumlah fitur
    yang digunakan dalam data pelatihan
37.     num_frames_expected = 320 # Ganti dengan jumlah frame
    yang digunakan dalam data pelatihan
38.     mfccs_normalized =
    adjust_feature_dimension(mfccs_normalized, num_features_expected,
    num_frames_expected)
39.
40.     # Load model SVM yang telah dilatih
41.     best_svm =
    joblib.load('music_genre_classification_model.pkl')
42.
43.     # Lakukan prediksi genre musik
44.     predicted_genre = best_svm.predict(mfccs_normalized.T) #
    Transpose mfccs_normalized agar dimensi sesuai
45.     # Cetak hasil prediksi
46.     print("Predicted genre for", filename, ":",
    predicted_genre[0])
```

Terdapat beberapa langkah yang dilakukan untuk memproses file audio, menyesuaikan dimensi fitur, memuat model SVM yang telah dilatih, dan melakukan prediksi genre musik untuk setiap file audio yang diunggah. Berikut adalah penjelasan untuk setiap langkah:

1. Fungsi untuk Menyesuaikan Dimensi Fitur (*adjust\_feature\_dimension*):
  - a) Ini adalah fungsi yang digunakan untuk menyesuaikan dimensi fitur agar sesuai dengan model yang digunakan untuk prediksi.
  - b) Jika jumlah fitur kurang dari yang diharapkan, fungsi akan menambahkan fitur tambahan dengan nilai default (nol).



- c) Jika jumlah fitur lebih dari yang diharapkan, fungsi akan memotong fitur yang berlebihan.
  - d) Hal yang sama dilakukan untuk jumlah frame (kolom) dari fitur.
2. Upload File Audio:
    - a) Pengguna diminta untuk mengunggah file audio yang akan diproses.
  3. Loop Melalui File yang Diunggah:
    - a) Setiap file audio yang diunggah akan diproses satu per satu.
  4. Baca File Audio:
    - a) File audio dibaca menggunakan `librosa.load()` untuk mendapatkan sinyal audio (`y`) dan tingkat sampel (`sr`).
  5. Ekstraksi Fitur Audio (Misalnya, MFCC):
    - a) Fitur audio (dalam contoh ini, MFCC) diekstraksi menggunakan `librosa.feature.mfcc()` dari sinyal audio yang dibaca sebelumnya.
  6. Normalisasi Fitur:
    - a) Fitur audio yang diekstraksi dapat dinormalisasi jika diperlukan. Dalam contoh ini, fitur MFCC dinormalisasi dengan mengurangi mean dan membagi dengan standar deviasi.
  7. Menyesuaikan Dimensi Fitur:
    - a) Dimensi fitur yang telah diekstraksi disesuaikan agar sesuai dengan dimensi yang digunakan dalam data pelatihan model SVM. Ini dilakukan dengan menggunakan fungsi `adjust_feature_dimension` yang telah didefinisikan sebelumnya.
  8. Memuat Model SVM yang Telah Dilatih:
    - a) Model SVM yang telah dilatih sebelumnya dimuat menggunakan `joblib.load()`.
  9. Melakukan Prediksi Genre Musik:
    - a) Fitur-fitur audio yang telah dinormalisasi dan disesuaikan dimensinya digunakan untuk melakukan prediksi genre musik dengan memanggil metode `predict()` dari model SVM.
  10. Cetak Hasil Prediksi:
    - a) Hasil prediksi genre musik untuk setiap file audio yang diunggah dicetak.

Dengan demikian, kode tersebut memungkinkan untuk memproses file audio, menyesuaikan dimensi fitur, memuat model SVM yang telah dilatih, dan melakukan prediksi genre musik untuk setiap file audio yang diunggah.

## 5. KESIMPULAN

Melalui serangkaian eksperimen dan evaluasi, telah berhasil mengembangkan model klasifikasi yang mampu mengidentifikasi genre musik dari sampel audio. Dalam evaluasi model, nilai pengujian confusion matrix pada genre blues didapatkan sebesar 84% precision, 81% recall, dan 82% f1-score, pada genre classical didapatkan sebesar 88% precision, 83% recall, dan 86% f1-score, pada genre jazz didapatkan sebesar 71% precision, 83% recall, dan 77% f1-score, pada genre pop didapatkan sebesar 94% precision, 89% recall dan 91% f1-score. Terdapat variasi performa model tergantung pada genre musik yang dinilai. Sebagai contoh, genre pop mencapai tingkat precision yang sangat tinggi sebesar 94%, menandakan kemampuan model untuk mengenali genre ini dengan presisi yang tinggi. Di sisi lain, genre jazz menunjukkan nilai

precision yang lebih rendah sebesar 71%, menandakan adanya tantangan dalam mengklasifikasikan sampel-sampel audio dari genre ini. Hasil ini menyoroti pentingnya analisis lebih lanjut terhadap faktor-faktor yang mempengaruhi performa prediksi untuk setiap genre musik, sehingga dapat dilakukan penyesuaian model yang lebih optimal.

Untuk meningkatkan performa model, beberapa saran dapat dipertimbangkan. Pertama, perlu diperhatikan proses ekstraksi fitur audio. Fitur-fitur yang relevan dan representatif dapat menjadi kunci dalam meningkatkan kemampuan model untuk membedakan antara genre musik yang berbeda. Selain itu, pemilihan dan pemrosesan fitur yang lebih cermat dapat memberikan hasil yang lebih baik. Penambahan data latihan yang lebih representatif dari setiap genre musik juga dapat menjadi langkah penting dalam meningkatkan performa model. Dengan dataset yang lebih besar dan bervariasi, model akan memiliki kemampuan yang lebih baik untuk mempelajari pola-pola yang kompleks dan unik dari setiap genre musik.

Selanjutnya, eksplorasi teknik-teknik pengembangan model yang lebih kompleks seperti ensemble learning dapat menjadi langkah selanjutnya. Dengan memanfaatkan kekuatan dari beberapa model yang berbeda, ensemble learning dapat meningkatkan performa prediksi secara signifikan. Selain itu, penyesuaian parameter dan pengoptimalan model yang lebih cermat juga dapat membantu dalam meningkatkan akurasi dan stabilitas model

#### **DAFTAR PUSTAKA**

- [1] A, Y., C, A. T., C, O., & I, M. (n.d.). Music Genre Classification Using Support Vector Machine Techniques. *International Conference on Information Management and Technology (ICIMTech)*, 511-516.
- [2] G, T., & P, C. (2002). Musical genre classification of audio signals," in *IEEE Transactions on Speech and Audio Processing*,. IEEE.
- [3] G, T., & P, C. (2002). Musical genre classification of audio signals," in *IEEE Transactions on Speech and Audio Processing*,. IEEE.
- [4] Narkhede, N., Mathur, S., & Bhaskar, A. (2022). *Automatic Classification of Music Genre Using SVM*. Singapore: Springer Nature Singapore.