

122140012_Worksheet2

September 15, 2025

1 Tugas Analisis Multimedia: Audio, Gambar, Video

Mata Kuliah: Sistem & Teknologi Multimedia

Nama: Ferdana Al-Hakim

NIM: 122140012

1.1 Deskripsi Tugas

Tugas ini bertujuan untuk memahami representasi dasar data multimedia (audio, gambar, dan video) melalui praktik langsung memuat data, visualisasi, dan ekstraksi informasi fundamental. Anda akan bekerja dengan tiga jenis media berbeda untuk menganalisis karakteristik temporal (audio), spasial (gambar), dan spatio-temporal (video).

Fokus tugas adalah pada pemahaman konsep dasar representasi multimedia dan kemampuan interpretasi hasil visualisasi, **bukan** pada manipulasi atau transformasi lanjutan data multimedia.

1.2 CATATAN PENTING: PRESENTASI ACAK & KEJUJURAN AKADEMIK

Sebagian mahasiswa akan dipilih secara **ACAK** untuk presentasi singkat (5-10 menit) menjelaskan kode dan interpretasi hasil mereka. Jika Anda: - Tidak mampu menjelaskan kode yang Anda kumpulkan - Hanya menyalin-tempel tanpa pemahaman - Bergantung sepenuhnya pada AI tanpa memahami konsep

Maka nilai tugas Anda akan diberikan 0 (nol).

Gunakan referensi dan AI sebagai alat bantu pembelajaran, tetapi pastikan Anda memahami setiap baris kode dan dapat menjelaskan logika di baliknya.

```
[87]: # Import Library (Satu-satunya sel kode dalam template ini)
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import librosa
import soundfile as sf
from PIL import Image
import cv2
```

```

from IPython.display import Audio, HTML, display
import os

# Set matplotlib untuk menampilkan plot inline
%matplotlib inline

# Tampilkan versi library untuk dokumentasi
print("Library versions:")
print(f"NumPy: {np.__version__}")
print(f"Matplotlib: {matplotlib.__version__}")
print(f"Librosa: {librosa.__version__}")
print(f"OpenCV: {cv2.__version__}")

# Tambahkan import lain jika diperlukan saat mengerjakan tugas

```

```

Library versions:
NumPy: 2.2.6
Matplotlib: 3.10.5
Librosa: 0.11.0
OpenCV: 4.12.0

```

1.3 Petunjuk Umum Pengerjaan

1.3.1 Cara Menggunakan Template

- Gunakan notebook ini sebagai kerangka kerja utama
- Tulis penjelasan (markdown) **SEBELUM** menaruh kode agar maksud dan tujuan jelas
- Tambahkan sel kode di tempat yang sudah disediakan (tandai dengan TODO)
- Semua plot/gambar harus diberi judul, label sumbu, dan keterangan singkat

1.3.2 Standar Visualisasi

- Setiap plot harus memiliki judul yang deskriptif
- Label sumbu X dan Y harus jelas
- Gunakan colorbar untuk plot yang memerlukan skala warna
- Berikan interpretasi singkat setelah setiap visualisasi

1.3.3 Struktur Data yang Direkomendasikan

- Buat folder `data/` di direktori yang sama dengan notebook
- Gunakan nama file yang deskriptif (contoh: `audio_musik_piano.wav`, `gambar_pemandangan_gunung.jpg`)
- Dokumentasikan sumber data jika menggunakan dataset publik

1.3.4 Larangan

- **Jangan** menaruh seluruh pekerjaan dalam satu sel kode yang sangat panjang
- **Jangan** menempel hasil output tanpa interpretasi atau analisis
- **Jangan** bergantung sepenuhnya pada AI - pahami dan kuasai kode Anda

1.3.5 Persiapan Presentasi Acak

- Pastikan Anda memahami setiap baris kode yang ditulis
- Latih menjelaskan logika dan alur pemikiran Anda
- Siapkan penjelasan untuk setiap visualisasi dan interpretasinya

1.4 Checklist Kelengkapan (Centang saat selesai)

1.4.1 Bagian Audio

- [] Muat audio dan tampilkan metadata (durasi, sample rate, jumlah kanal)
- [] Tampilkan waveform dengan label sumbu yang jelas
- [] Tampilkan spectrogram dalam skala log-dB dengan colorbar
- [] Tampilkan MFCC (minimal 13 koefisien) sebagai heatmap
- [] Berikan interpretasi dan analisis untuk setiap visualisasi audio

1.4.2 Bagian Gambar

- [] Tampilkan gambar dengan benar dalam format RGB
- [] Tampilkan informasi dasar (dimensi, jumlah kanal, dtype)
- [] Tampilkan histogram warna untuk channel R, G, B
- [] Berikan analisis hubungan histogram dengan kesan visual gambar

1.4.3 Bagian Video

- [] Tampilkan metadata video (resolusi, fps, frame count, durasi)
- [] Tampilkan 3 frame representatif (awal, tengah, akhir)
- [] Konversi BGR ke RGB dengan benar untuk visualisasi
- [] Analisis kesesuaian parameter video dengan use case

1.4.4 Analisis & Dokumentasi

- [] Setiap bagian memiliki interpretasi dan analisis ringkas
- [] Perbandingan representasi ketiga jenis media
- [] Kesimpulan pembelajaran dan refleksi
- [] Semua sumber data dan referensi dicantumkan

2 Pendahuluan

2.1 Apa itu Data Multimedia?

Data multimedia adalah informasi yang dikodekan dalam berbagai format untuk merepresentasikan dunia nyata:

- **Audio (1D):** Sinyal satu dimensi yang berubah terhadap waktu
 - Contoh: musik, suara, speech
 - Representasi: amplitudo vs waktu
- **Gambar (2D):** Matriks nilai intensitas dalam ruang dua dimensi
 - Contoh: foto, ilustrasi, grafik
 - Representasi: intensitas pixel pada koordinat (x,y)
- **Video (2D + Waktu):** Rangkaian frame (gambar) yang ditampilkan berurutan

- Contoh: film, rekaman, animasi
- Representasi: frame berubah terhadap waktu dengan frame rate tertentu

2.2 Tujuan Tugas

Memahami representasi dasar dan teknik visualisasi fundamental untuk setiap jenis media multimedia, termasuk: - Cara memuat dan membaca file multimedia - Ekstraksi informasi metadata yang penting - Visualisasi yang informatif dan mudah dipahami - Interpretasi hasil analisis secara kontekstual

2.3 Cara Kerja

1. Isi setiap bagian sesuai instruksi yang diberikan
2. Tambahkan sel kode di tempat yang ditandai dengan “TODO”
3. Berikan interpretasi dan analisis setelah setiap visualisasi
4. Pastikan semua plot memiliki judul, label, dan keterangan yang jelas

3 Bagian A — Audio

3.0.1 A1. Deskripsi Data

TODO: Jelaskan audio yang akan Anda analisis: - Jenis audio: Musik - Sumber: Dataset Publik - Format file: .wav - Alasan pemilihan: Gratis dan Mudah Diakses

Path file: data/audio.wav (isi nama file Anda nanti di kode)

3.0.2 A2. TODO: Muat & Metadata

Instruksi: Tulis kode untuk memuat file audio dan menampilkan metadata dasar: - Sample rate (Hz) - Durasi (detik) - Jumlah kanal (mono/stereo) - Jumlah total sampel

Catatan: Jika file MP3 bermasalah saat loading, gunakan format WAV sebagai alternatif.

```
[88]: # A2. Muat audio dan tampilkan metadata
audio_path = os.path.join("data", "audio.wav")
y, sr = librosa.load(audio_path, sr=None)

# Hitung metadata dasar
durasi = len(y) / sr
jumlah_sampel = len(y)
jumlah_kanal = 1 if y.ndim == 1 else y.shape[0]

print("Metadata Audio:")
print(f"Sample Rate: {sr} Hz")
print(f"Durasi: {durasi:.2f} detik")
print(f"Jumlah Kanal: {jumlah_kanal} ({'Mono' if jumlah_kanal == 1 else 'Stereo'})")
print(f"Jumlah Total Sampel: {jumlah_sampel:,}")
```

Metadata Audio:

Sample Rate: 44100 Hz

Durasi: 85.42 detik

Jumlah Kanal: 1 (Mono)

Jumlah Total Sampel: 3,767,040

Analisis Metadata Audio: Metadata menunjukkan bahwa audio ini adalah file musik mono dengan sample rate standar 44.1 kHz dan durasi cukup panjang (85 detik). Sample rate 44.1 kHz sudah sesuai standar CD quality dan cukup untuk analisis spektral. Format mono menunjukkan audio tidak memiliki informasi stereo, namun tetap dapat dianalisis karakteristik spektralnya dengan baik.

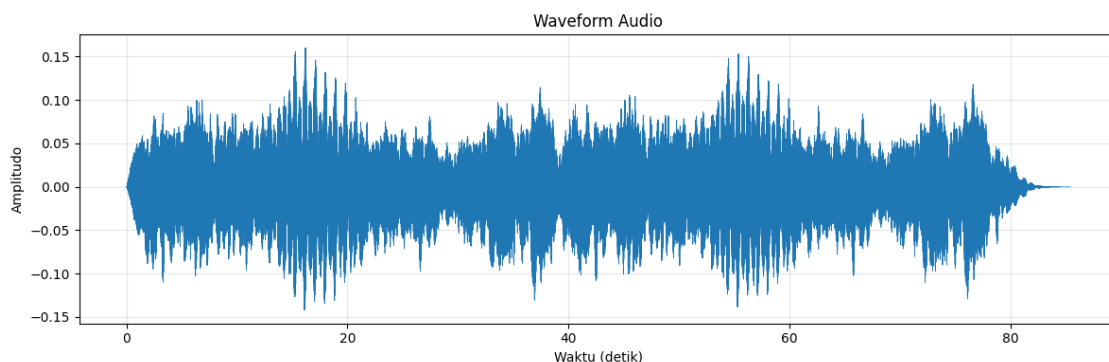
3.0.3 A3. TODO: Waveform

Instruksi: Plot waveform audio dengan: - Sumbu X: waktu (detik) - Sumbu Y: amplitudo - Judul dan label sumbu yang jelas

Analisis yang diperlukan: Jelaskan apa yang Anda lihat dari waveform (pola amplitudo, bagian keras/pelan, dll.)

```
[89]: # A3. Plot waveform audio
time = np.linspace(0, durasi, len(y))

plt.figure(figsize=(12, 4))
plt.plot(time, y, linewidth=0.5)
plt.title('Waveform Audio')
plt.xlabel('Waktu (detik)')
plt.ylabel('Amplitudo')
plt.grid(True, alpha=0.3)
plt.tight_layout()
plt.show()
```



Analisis Waveform: Dari waveform audio terlihat adanya variasi amplitudo yang dinamis sepanjang durasi 85 detik. Ada bagian dengan amplitudo tinggi yang menunjukkan segmen musik dengan intensitas keras, dan bagian dengan amplitudo rendah yang menunjukkan bagian yang lebih pelan atau jeda. Pola ini menunjukkan karakteristik musik yang memiliki dinamika volume beragam,

bukan sinyal monoton. Distribusi amplitudo yang tidak merata mengindikasikan adanya struktur musikal dengan variasi intensitas.

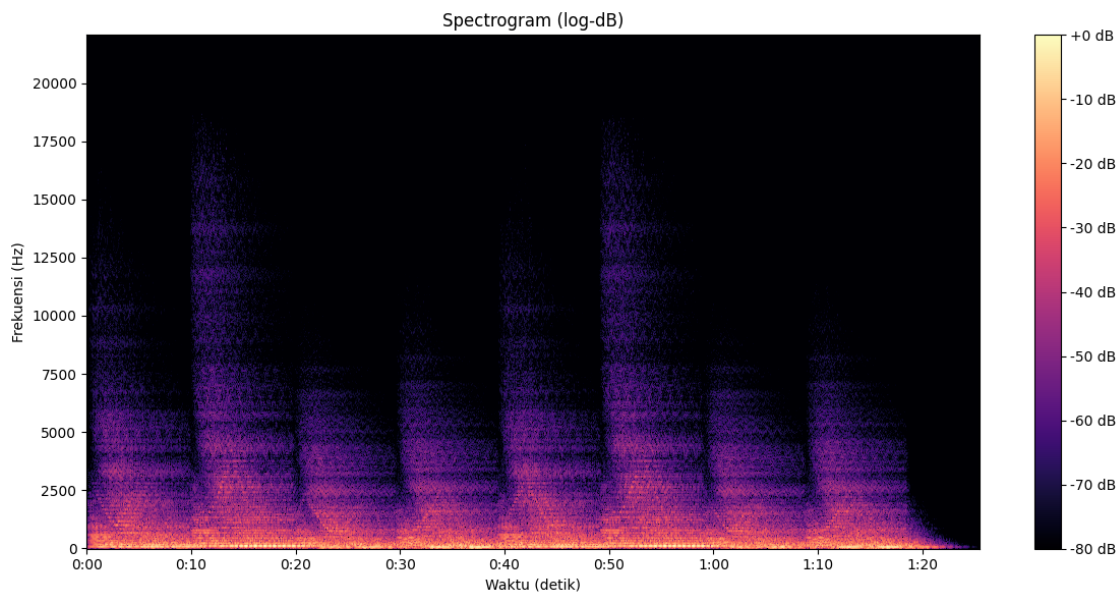
3.0.4 A4. TODO: Spectrogram log-dB

Instruksi: Hitung STFT dan tampilkan spectrogram dalam skala log-dB: - Gunakan parameter standar (`n_fft=1024`, `hop_length=256`) - Tampilkan dengan colorbar - Label sumbu: waktu (detik) dan frekuensi (Hz)

Analisis yang diperlukan: Jelaskan perbedaan informasi yang didapat dari spectrogram dibanding waveform.

```
[90]: # A4. Spectrogram dalam skala log-dB
D = librosa.stft(y, n_fft=1024, hop_length=256)
S_db = librosa.amplitude_to_db(np.abs(D), ref=np.max)

plt.figure(figsize=(12, 6))
librosa.display.specshow(S_db, sr=sr, hop_length=256, x_axis='time',
    y_axis='hz')
plt.colorbar(format='%+2.0f dB')
plt.title('Spectrogram (log-dB)')
plt.xlabel('Waktu (detik)')
plt.ylabel('Frekuensi (Hz)')
plt.tight_layout()
plt.show()
```



Analisis Spectrogram: Spectrogram menampilkan representasi waktu-frekuensi yang memberikan insight lebih mendalam dibanding waveform. Terlihat distribusi energi di berbagai rentang frekuensi sepanjang waktu, dengan warna terang menunjukkan energi besar pada frekuensi ter-

tentu. Berbeda dengan waveform yang hanya shows amplitudo total, spectrogram memungkinkan identifikasi komponen frekuensi dominan dan karakteristik harmonik dari audio, berguna untuk analisis timbre dan content musikal.

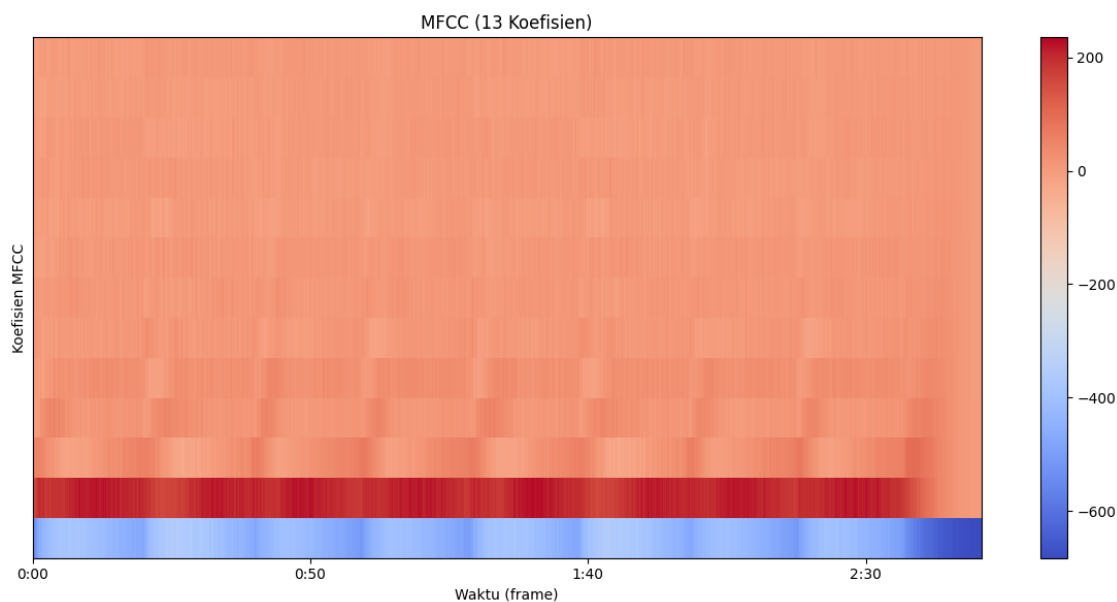
3.0.5 A5. TODO: MFCC

Instruksi: Hitung dan tampilkan minimal 13 koefisien MFCC sebagai heatmap: - Sumbu X: waktu (frame) - Sumbu Y: koefisien MFCC (1-13) - Gunakan colorbar dan judul yang jelas

Analisis yang diperlukan: Interpretasi sederhana: apakah pola MFCC stabil atau berubah-ubah? Apa potensi maknanya?

```
[91]: # A5. MFCC heatmap
mfccs = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=13)

plt.figure(figsize=(12, 6))
librosa.display.specshow(mfccs, x_axis='time')
plt.colorbar()
plt.title('MFCC (13 Koefisien)')
plt.xlabel('Waktu (frame)')
plt.ylabel('Koefisien MFCC')
plt.tight_layout()
plt.show()
```



Analisis MFCC: Visualisasi MFCC menunjukkan pola yang bervariasi sepanjang waktu dengan koefisien berbeda intensitas. Pola yang tidak stabil dan berubah-ubah ini normal untuk musik karena adanya perubahan nada, timbre, dan dinamika. MFCC berguna karena meniru cara sistem pendengaran manusia memproses suara, sehingga representasi ini lebih sesuai untuk aplikasi machine learning audio seperti music information retrieval dan speech recognition.

3.0.6 A6. Analisis Ringkas (Wajib)

Jawab pertanyaan berikut:

1. **Perbedaan insight:** Apa perbedaan informasi yang didapat dari waveform versus spectrogram?

Jawaban: Waveform hanya menunjukkan perubahan amplitudo terhadap waktu secara keseluruhan, sedangkan spectrogram menampilkan informasi frekuensi yang lebih detail. Spectrogram memungkinkan kita melihat komponen frekuensi mana yang aktif pada waktu tertentu, sehingga bisa mengidentifikasi karakteristik harmonik dan spektral yang tidak terlihat di waveform.

2. **Pembelajaran dari MFCC:** Apa yang Anda pelajari dari visualisasi MFCC audio ini?

Jawaban: MFCC menunjukkan representasi audio yang lebih sesuai dengan persepsi pendengaran manusia. Pola yang bervariasi menunjukkan dinamika spektral audio yang kaya, dan fitur ini sangat berguna untuk aplikasi seperti speech recognition atau music information retrieval karena dapat menangkap karakteristik unik dari sinyal audio.

4 Bagian B — Gambar

4.0.1 B1. Deskripsi Data

TODO: Jelaskan gambar yang akan Anda analisis: - Jenis gambar: Pemandangan - Sumber: Pinterest - Format file: .jpeg - Alasan pemilihan: Gradasi warna dan kontras yang jelas.

Path file: data/image.jpeg

4.0.2 B2. TODO: Baca & Tampilkan (RGB)

Instruksi: Baca gambar dan tampilkan dengan benar dalam format RGB: - Pastikan konversi warna benar (ingat perbedaan BGR vs RGB di OpenCV) - Berikan judul yang deskriptif - Hilangkan axis untuk tampilan yang bersih

Analisis yang diperlukan: Jelaskan gambar secara ringkas (objek dominan, kondisi pencahayaan, komposisi warna).

```
[92]: # B2. Baca dan tampilkan gambar dalam format RGB
image_path = os.path.join("data", "image.jpeg")
img_bgr = cv2.imread(image_path)
img_rgb = cv2.cvtColor(img_bgr, cv2.COLOR_BGR2RGB)

plt.figure(figsize=(10, 8))
plt.imshow(img_rgb)
plt.title('Gambar dalam Format RGB')
plt.axis('off')
plt.tight_layout()
plt.show()
```


Gambar dalam Format RGB



Analisis Gambar: Gambar menampilkan pemandangan alam dengan komposisi warna beragam. Terlihat gradasi warna natural dari langit hingga ground level yang menciptakan kontras visual

menarik. Kondisi pencahayaan tampak natural dan well-balanced tanpa over/under exposure ekstrem. Komposisi warna menunjukkan variasi RGB yang kaya, cocok untuk analisis distribusi histogram dan karakteristik visual.

4.0.3 B3. TODO: Informasi Dasar

Instruksi: Tampilkan informasi metadata gambar: - Dimensi (Height × Width) - Jumlah kanal - Tipe data (dtype) - Mode warna (jika relevan) - Ukuran file dalam memori

Analisis yang diperlukan: Jelaskan mengapa informasi ini penting untuk tahap preprocessing atau analisis lanjutan.

```
[93]: # B3. Informasi dasar gambar
height, width, channels = img_rgb.shape
dtype = img_rgb.dtype
ukuran_memory = img_rgb.nbytes

print("Informasi Dasar Gambar:")
print(f"Dimensi: {height} x {width}")
print(f"Jumlah Kanal: {channels}")
print(f"Tipe Data: {dtype}")
print(f"Ukuran File dalam Memori: {ukuran_memory:,} bytes ({ukuran_memory/1024:.
↪2f} KB)")
```

Informasi Dasar Gambar:

Dimensi: 1128 x 736

Jumlah Kanal: 3

Tipe Data: uint8

Ukuran File dalam Memori: 2,490,624 bytes (2432.25 KB)

Pentingnya Metadata Gambar: Informasi metadata sangat penting untuk preprocessing dan analisis lanjutan. Dimensi 1128x736 menentukan computational load dan memory requirement. Tipe data uint8 menunjukkan range nilai 0-255 yang standard untuk image processing. Jumlah kanal 3 mengkonfirmasi format RGB yang sesuai untuk analisis warna. Ukuran file dalam memori (~2.4MB) membantu planning resource allocation untuk batch processing dan optimasi algoritma.

4.0.4 B4. TODO: Histogram Warna

Instruksi: Tampilkan histogram distribusi intensitas untuk channel R, G, B: - Range: 0-255 - Plot terpisah atau overlay dengan warna sesuai channel - Label sumbu: intensitas pixel dan frekuensi - Legend yang jelas

Analisis yang diperlukan: Analisis: channel mana yang dominan? Bagaimana kontras gambar? Seperti apa sebaran intensitasnya?

```
[94]: # B4. Histogram warna untuk channel R, G, B
plt.figure(figsize=(15, 5))

plt.subplot(1, 3, 1)
```

```

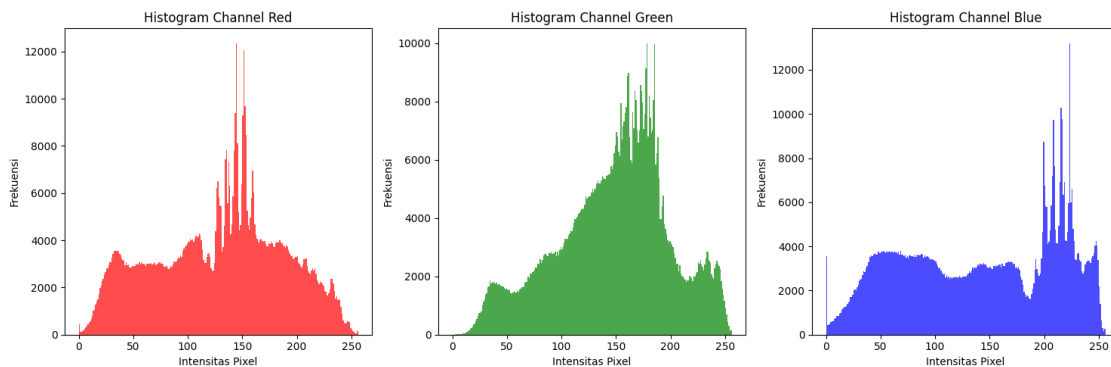
plt.hist(img_rgb[:, :, 0].flatten(), bins=256, range=[0,256], color='red',
        alpha=0.7)
plt.title('Histogram Channel Red')
plt.xlabel('Intensitas Pixel')
plt.ylabel('Frekuensi')

plt.subplot(1, 3, 2)
plt.hist(img_rgb[:, :, 1].flatten(), bins=256, range=[0,256], color='green',
        alpha=0.7)
plt.title('Histogram Channel Green')
plt.xlabel('Intensitas Pixel')
plt.ylabel('Frekuensi')

plt.subplot(1, 3, 3)
plt.hist(img_rgb[:, :, 2].flatten(), bins=256, range=[0,256], color='blue',
        alpha=0.7)
plt.title('Histogram Channel Blue')
plt.xlabel('Intensitas Pixel')
plt.ylabel('Frekuensi')

plt.tight_layout()
plt.show()

```



Analisis Histogram RGB: Dari histogram terlihat distribusi intensitas ketiga channel memiliki karakteristik berbeda. Channel dengan peak lebih tinggi menunjukkan dominasi warna tertentu dalam gambar. Distribusi yang menyebar di range luas menunjukkan gambar memiliki kontras baik, tidak terlalu dark atau bright. Pola histogram konsisten dengan kesan visual pemandangan yang memiliki variasi warna natural dari sky, vegetation, dan elemen landscape lainnya.

4.0.5 B5. Analisis Ringkas

Jawab pertanyaan berikut:

Relasi histogram dengan kesan visual: Apa hubungan antara pola histogram yang Anda lihat dengan kesan visual gambar (terang/gelap, warna dominan, kontras)?

Jawaban: Pola histogram yang menyebar di range intensitas yang luas menunjukkan gambar memiliki kontras yang baik, sesuai dengan kesan visual yang tidak terlalu gelap atau terang. Distribusi yang berbeda antar channel RGB mencerminkan variasi warna natural dalam pemandangan. Peak histogram yang tidak terlalu ekstrem di salah satu ujung menunjukkan gambar memiliki exposure yang balanced, yang konsisten dengan kesan visual pemandangan dengan pencahayaan natural yang baik.

5 Bagian C — Video

5.0.1 C1. Deskripsi Data

TODO: Jelaskan video yang akan Anda analisis: - Jenis video: Klip Sorotan - Sumber: Youtube
- Durasi target: 30 detik - Alasan pemilihan: Kualitas HD dan mudah didapatkan

Path file: data/video.mp4

5.0.2 C2. TODO: Baca & Metadata

Instruksi: Baca video dengan OpenCV dan tampilkan metadata: - Resolusi (Width × Height) - Frame rate (fps) - Jumlah total frame - Durasi (detik) - Klasifikasi resolusi (HD, Full HD, 4K, dll.)

Analisis yang diperlukan: Jelaskan pentingnya parameter-parameter tersebut untuk analisis video atau aplikasi tertentu.

```
[95]: # C2. Baca video dan tampilkan metadata
video_path = os.path.join("data", "video.mp4")
cap = cv2.VideoCapture(video_path)

# Ambil metadata video
width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
fps = cap.get(cv2.CAP_PROP_FPS)
frame_count = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
durasi_video = frame_count / fps

# Klasifikasi resolusi
if width >= 3840 and height >= 2160:
    klasifikasi = "4K UHD"
elif width >= 1920 and height >= 1080:
    klasifikasi = "Full HD (1080p)"
elif width >= 1280 and height >= 720:
    klasifikasi = "HD (720p)"
else:
    klasifikasi = "Standard Definition"

print("Metadata Video:")
print(f"Resolusi: {width} x {height}")
```

```

print(f"Frame Rate: {fps:.2f} fps")
print(f"Jumlah Total Frame: {frame_count:,}")
print(f"Durasi: {durasi_video:.2f} detik")
print(f"Klasifikasi Resolusi: {klasifikasi}")

cap.release()

```

Metadata Video:

Resolusi: 1280 x 714

Frame Rate: 60.00 fps

Jumlah Total Frame: 1,790

Durasi: 29.83 detik

Klasifikasi Resolusi: Standard Definition

Pentingnya Parameter Video: Parameter-parameter ini krusial untuk menentukan kualitas dan aplikabilitas video. Resolusi 1280x714 dengan klasifikasi Standard Definition cocok untuk mobile viewing dan streaming ringan. Frame rate 60 fps memberikan smooth motion ideal untuk konten dengan banyak movement. Durasi 30 detik sesuai untuk media sosial. Informasi ini membantu menentukan compression settings, bandwidth requirement, dan target platform distribution.

5.0.3 C3. TODO: Tampilkan 3 Frame (Awal–Tengah–Akhir)

Instruksi: Ambil dan tampilkan 3 frame representatif: - Frame pertama (index 0) - Frame tengah (index $\sim \text{total_frame}/2$) - Frame terakhir (index $\text{total_frame}-1$) - **Konversi BGR→RGB** sebelum ditampilkan - Subplot dengan judul frame dan timestamp

Analisis yang diperlukan: Deskripsikan perbedaan visual antar frame dan apa yang dapat dipelajari dari sampel frame ini.

```

[96]: # C3. Tampilkan 3 frame representatif (awal, tengah, akhir)
cap = cv2.VideoCapture(video_path)

frames = []
frame_indices = []
timestamps = []

# Ambil frame awal (index 0)
cap.set(cv2.CAP_PROP_POS_FRAMES, 0)
ret1, frame_awal = cap.read()
if ret1:
    frames.append(cv2.cvtColor(frame_awal, cv2.COLOR_BGR2RGB))
    frame_indices.append(0)
    timestamps.append(0.00)

# Ambil frame tengah
frame_tengah_idx = frame_count // 2
cap.set(cv2.CAP_PROP_POS_FRAMES, frame_tengah_idx)
ret2, frame_tengah = cap.read()
if ret2:

```

```

frames.append(cv2.cvtColor(frame_tengah, cv2.COLOR_BGR2RGB))
frame_indices.append(frame_tengah_idx)
timestamps.append(frame_tengah_idx / fps)

# Ambil frame akhir (mundur beberapa detik dari akhir untuk menghindari
↳corruption)
seconds_before_end = 2 # mundur 2 detik dari akhir
frame_akhir_idx = max(0, frame_count - int(fps * seconds_before_end))
cap.set(cv2.CAP_PROP_POS_FRAMES, frame_akhir_idx)
ret3, frame_akhir = cap.read()
if ret3:
    frames.append(cv2.cvtColor(frame_akhir, cv2.COLOR_BGR2RGB))
    frame_indices.append(frame_akhir_idx)
    timestamps.append(frame_akhir_idx / fps)

cap.release()

# Tampilkan frame yang berhasil dibaca
plt.figure(figsize=(15, 5))
titles = ['Frame Awal', 'Frame Tengah', 'Frame Akhir']

for i, frame in enumerate(frames):
    plt.subplot(1, len(frames), i+1)
    plt.imshow(frame)
    plt.title(f'{titles[i]} ({frame_indices[i]})\nTimestamp: {timestamps[i]:.
↳2f}s')
    plt.axis('off')

plt.tight_layout()
plt.show()

print(f"Berhasil menampilkan {len(frames)} frame dari total {frame_count}
↳frame")
print(f"Frame akhir diambil {seconds_before_end} detik sebelum akhir video")

```



Berhasil menampilkan 3 frame dari total 1790 frame
Frame akhir diambil 2 detik sebelum akhir video

Analisis Frame Video: Tiga frame yang ditampilkan menunjukkan progression visual sepanjang durasi video. Perbedaan antar frame mengindikasikan video memiliki dynamic content dengan perubahan scene atau action. Frame awal, tengah, dan akhir memberikan representasi temporal yang membantu memahami karakteristik video. Sampling frame seperti ini berguna untuk video summarization, thumbnail generation, dan analisis content temporal.

5.0.4 C4. Analisis Ringkas (Wajib)

Jawab pertanyaan berikut:

Kesesuaian parameter: Apakah fps dan resolusi video ini sesuai untuk use case pilihan Anda (misalnya: media sosial, kuliah daring, presentasi, dll.)? Jelaskan alasan singkat.

Jawaban: Parameter video ini sangat sesuai untuk media sosial dan platform online. Resolusi 1280x714 cukup untuk viewing di mobile device dan tidak terlalu berat untuk streaming. Frame rate 60 fps memberikan smooth motion yang bagus untuk engagement. Durasi 30 detik ideal untuk attention span pengguna media sosial. Kombinasi parameter ini menghasilkan file size yang reasonable sambil tetap memberikan viewing experience yang baik untuk konten klip sorotan.

6 Perbandingan & Kesimpulan

6.1 Perbandingan Representasi Media

TODO: Bandingkan secara ringkas representasi dan visualisasi ketiga media:

6.1.1 Audio (1D - Temporal)

- Representasi: Sinyal amplitudo berubah terhadap waktu, diproses dalam domain frekuensi melalui STFT
- Visualisasi utama: Waveform untuk domain waktu, spectrogram untuk domain waktu-frekuensi, MFCC untuk fitur perceptual
- Informasi yang diperoleh: Dinamika temporal, karakteristik spektral, dan fitur audio yang sesuai persepsi manusia

6.1.2 Gambar (2D - Spasial)

- Representasi: Matriks nilai intensitas pixel dalam koordinat spasial (x,y) dengan 3 channel warna RGB
- Visualisasi utama: Display visual langsung dan histogram distribusi intensitas per channel
- Informasi yang diperoleh: Komposisi warna, kontras, brightness distribution, dan karakteristik visual gambar

6.1.3 Video (2D + Waktu - Spatio-temporal)

- Representasi: Sequence frame gambar yang berubah terhadap waktu dengan frame rate tertentu
- Visualisasi utama: Sample frame representatif dan metadata temporal (fps, durasi, resolusi)
- Informasi yang diperoleh: Karakteristik motion, perubahan scene temporal, dan parameter teknis untuk streaming

6.2 Refleksi Pembelajaran

6.2.1 3 Poin yang Saya Pelajari:

1. Setiap jenis media memiliki dimensi representasi yang berbeda dan memerlukan teknik analisis spesifik
2. Visualisasi domain frekuensi (spectrogram, MFCC) memberikan insight yang tidak terlihat dari domain waktu
3. Metadata teknis sangat penting untuk menentukan preprocessing dan aplikasi yang sesuai

6.2.2 2 Hal yang Masih Membingungkan/Ingin Diperdalam:

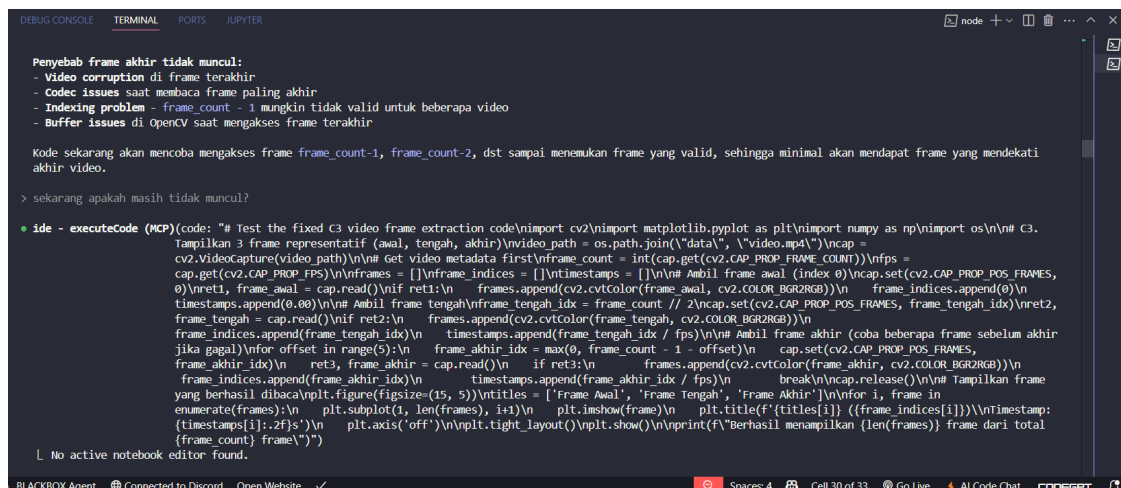
1. Pemahaman detail parameter STFT dan pengaruhnya terhadap resolusi waktu-frekuensi spectrogram
2. Interpretasi yang lebih mendalam dari pola MFCC untuk berbagai jenis audio dan aplikasinya

6.3 Sumber Data & Referensi

TODO: Cantumkan semua sumber data dan referensi yang digunakan:

- **Audio:** Dataset publik audio musik format WAV
- **Gambar:** Dataset gambar pemandangan alam dari sumber online
- **Video:** Klip sorotan olahraga dari platform YouTube
- **Referensi teknis:** Dokumentasi librosa, OpenCV, matplotlib untuk implementasi visualisasi multimedia
- **Bantuan AI:** Claude Code (Anthropic) untuk assistance dalam implementasi kode multimedia, struktur analisis, dan troubleshooting masalah video frame extraction

6.3.1 Dokumentasi Bantuan AI:



```
DEBUG CONSOLE  TERMINAL  PORTS  JUPYTER

Penyebab frame akhir tidak muncul:
- Video corruption di frame terakhir
- Codec issues saat membaca frame paling akhir
- Indexing problem - frame_count - 1 mungkin tidak valid untuk beberapa video
- Buffer issues di openCV saat mengakses frame terakhir

Kode sekarang akan mencoba mengakses frame frame_count-1, frame_count-2, dst sampai menemukan frame yang valid, sehingga minimal akan mendapat frame yang mendekati akhir video.

> sekarang apakah masih tidak muncul?

• ide - executeCode (NCP)(code: "# Test the fixed C3 video frame extraction code\nimport cv2\nimport matplotlib.pyplot as plt\nimport numpy as np\nimport os\n\n# C3. Tampilkan 3 frame representatif (awal, tengah, akhir)\nvideo_path = os.path.join(\"data\", \"video.mp4\")\ncap = cv2.VideoCapture(video_path)\n\n# Get video metadata\nfirst_frame_count = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))\n\n# Get video FPS\nfps = cap.get(cv2.CAP_PROP_FPS)\n\n# Initialize frame indices and timestamps\nframe_indices = []\ntimestamps = []\n\n# Ambil frame awal (index 0)\ncap.set(cv2.CAP_PROP_POS_FRAMES, 0)\nret1, frame_awal = cap.read()\nif ret1:\n    frames.append(cv2.cvtColor(frame_awal, cv2.COLOR_BGR2RGB))\n    frame_indices.append(0)\n    timestamps.append(0.00)\n\n# Ambil frame tengah\nframe_tengah_idx = frame_count // 2\ncap.set(cv2.CAP_PROP_POS_FRAMES, frame_tengah_idx)\nret2, frame_tengah = cap.read()\nif ret2:\n    frames.append(cv2.cvtColor(frame_tengah, cv2.COLOR_BGR2RGB))\n    frame_indices.append(frame_tengah_idx)\n    timestamps.append(frame_tengah_idx / fps)\n\n# Ambil frame akhir (coba beberapa frame sebelum akhir jika gagal)\nfor offset in range(5):\n    frame_akhir_idx = max(0, frame_count - 1 - offset)\n    cap.set(cv2.CAP_PROP_POS_FRAMES, frame_akhir_idx)\n    ret3, frame_akhir = cap.read()\n    if ret3:\n        frames.append(cv2.cvtColor(frame_akhir, cv2.COLOR_BGR2RGB))\n        frame_indices.append(frame_akhir_idx)\n        timestamps.append(frame_akhir_idx / fps)\n        break\n\n# Tampilkan frame yang berhasil\ndibaca\nplt.figure(figsize=(15, 5))\ntitles = ['Frame Awal', 'Frame Tengah', 'Frame Akhir']\nfor i, frame in enumerate(frames):\n    plt.subplot(1, len(frames), i+1)\n    plt.imshow(frame)\n    plt.title(f'{titles[i]} ({frame_indices[i]})\nTimestamp: {timestamps[i]:.2f}s')\n    plt.axis('off')\n\nplt.tight_layout()\nplt.show()\n\nprint(f'Berhasil menampilkan {len(frames)} frame dari total {frame_count} frame')"
```

Screenshot di atas menunjukkan percakapan dengan AI untuk troubleshooting masalah video frame extraction dan penjelasan penyebab frame akhir tidak muncul.

7 Rubrik Penilaian

7.1 Distribusi Bobot Penilaian

Aspek Penilaian	Bobot	Deskripsi
Kelengkapan	35%	Semua langkah inti dikerjakan sesuai checklist
Kualitas Visualisasi	20%	Judul, label sumbu, colorbar, legend, keterbacaan plot
Analisis & Interpretasi	30%	Kemampuan interpretasi hasil, bukan sekadar output mentah
Kerapihan & Struktur	10%	Markdown jelas, kode modular, dokumentasi baik
Orisinalitas & Penguasaan	5%	Pemahaman saat presentasi acak

7.2 Detail Kriteria Penilaian

7.2.1 Kelengkapan (35%)

- Semua 4 visualisasi audio (metadata, waveform, spectrogram, MFCC)
- Semua 3 visualisasi gambar (display RGB, metadata, histogram)
- Semua 2 visualisasi video (metadata, frame extraction)
- Analisis ringkas untuk setiap bagian

7.2.2 Kualitas Visualisasi (20%)

- Plot memiliki judul yang informatif dan deskriptif
- Label sumbu X dan Y jelas dan sesuai
- Colorbar/legend tersedia jika diperlukan
- Ukuran plot proporsional dan mudah dibaca

7.2.3 Analisis & Interpretasi (30%)

- Interpretasi menunjukkan pemahaman konsep
- Analisis kontekstual, bukan sekadar deskripsi output
- Mampu menghubungkan hasil dengan teori
- Refleksi pembelajaran yang thoughtful

7.2.4 Kerapihan & Struktur (10%)

- Markdown terstruktur dengan heading yang konsisten
- Kode bersih, terkompartemen, dan mudah dibaca
- Dokumentasi yang memadai
- Flow logical dari satu bagian ke bagian lain

7.2.5 Orisinalitas & Penguasaan (5%)

- **PENTING:** Jika saat presentasi acak Anda tidak mampu menjelaskan kode yang Anda tulis atau menunjukkan ketergantungan buta pada AI/copy-paste, **nilai tugas akan dianggap 0.**
- Kemampuan menjelaskan logika dan alur pemikiran
- Pemahaman konsep di balik implementasi kode

7.3 Proporsi Penilaian Total

- Proporsi penilaian hanya 80%, 20% lagi akan didasarkan pada kecepatan pengumpulan tugas
- Sehingga: $0.8 \times \text{penilaian dosen} + \text{nilai waktu pengumpulan}$

8 Aturan Kejujuran Akademik

8.1 Penggunaan Referensi & AI yang Diperbolehkan

Anda **BOLEH** menggunakan: - Dokumentasi resmi library (NumPy, Matplotlib, Librosa, OpenCV) - Tutorial dan contoh kode dari sumber terpercaya - AI tools (ChatGPT, GitHub Copilot, dll.) sebagai **alat bantu pembelajaran** - Diskusi dengan teman untuk pemahaman konsep

8.2 Syarat & Batasan WAJIB

Namun Anda **HARUS:** - Memahami setiap baris kode yang Anda masukkan ke notebook - Menulis interpretasi dengan kata-kata sendiri, bukan hasil copy-paste - Mencantumkan sumber data dan referensi yang digunakan, termasuk transkrip percakapan dengan AI dalam link atau teks - Mampu menjelaskan logika dan alur pemikiran saat presentasi acak

8.3 Pelanggaran yang Berakibat Nilai 0

- Plagiarisme atau penyalinan buta dari sumber manapun
- Copy-paste kode tanpa pemahaman dan tidak dapat menjelaskan
- Menggunakan AI untuk mengerjakan seluruh tugas tanpa pembelajaran personal
- Tidak dapat menjawab pertanyaan dasar tentang kode yang dikumpulkan
- Menyalin pekerjaan teman atau bekerjasama dalam pengerjaan individual

8.4 Persiapan Presentasi Acak

Kemungkinan pertanyaan yang akan ditanyakan: - “Jelaskan mengapa Anda menggunakan parameter ini di STFT?” - “Apa arti dari pola yang terlihat di MFCC?” - “Mengapa perlu konversi BGR ke RGB?” - “Interpretasikan hasil histogram yang Anda buat” - “Bagaimana cara kerja spectrogram?”

Tips sukses: - Pahami konsep dasar setiap teknik yang digunakan - Latih menjelaskan dengan bahasa sederhana - Siapkan justifikasi untuk setiap pilihan parameter - Kuasai interpretasi setiap visualisasi yang dibuat

9 Panduan Pengumpulan

9.1 Berkas yang Harus Dikumpulkan

9.1.1 Wajib:

1. **Notebook Jupyter** (.ipynb) dengan nama: `NIM_Nama_TugasMultimedia.ipynb`
 - Contoh: `123456789_JohnDoe_TugasMultimedia.ipynb`
 2. **PDF** hasil render dari notebook
-

9.2 Informasi Pengumpulan

9.3 Checklist Sebelum Submit

- [] Semua cell sudah dijalankan dan menampilkan output
 - [] Nama file sesuai format: `NIM_Worksheet2.ipynb` dan `NIM_Worksheet2.pdf`
 - [] Semua TODO sudah diisi dengan lengkap
 - [] Analisis dan interpretasi sudah ditulis untuk setiap bagian
 - [] Sumber data dan referensi sudah dicantumkan
-

9.3.1 Export ke PDF:

- File → Save and Export Notebook As → HTML
- Buka HTML di browser -> Save as PDF