

✓ Tugas Analisis Multimedia: Audio, Gambar, Video

Mata Kuliah: Sistem & Teknologi Multimedia

Nama: Rachel Olivia Manullang

NIM: 122140181

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.



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.

```
# Import Library (Satu-satunya sel kode dalam template ini)
import numpy as np
import matplotlib as mpl
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: {mpl.__version__}")
print(f"Librosa: {librosa.__version__}")
print(f"OpenCV: {cv2.__version__}")

# Tambahkan import lain jika diperlukan saat mengerjakan tugas
```

```
Library versions:
NumPy: 2.0.2
Matplotlib: 3.10.0
Librosa: 0.11.0
OpenCV: 4.12.0
```

Petunjuk Umum Pengerjaan



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



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

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

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

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

Checklist Kelengkapan (Centang saat selesai)





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





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

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

Analisis & Dokumentasi

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

Pendahuluan

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

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

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

✓ Bagian A — Audio

A1. Deskripsi Data

TODO: Jelaskan audio yang akan Anda analisis:

- Jenis audio: musik
- Sumber: download google
- Format file: mp3
- Alasan pemilihan: suka lagunya **Path file:** `data/Westlife_-_You_Raise_Me_Up_CeeNaija.com_.mp3` (isi nama file Anda nanti di kode)

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.

(Tambahkan sel kode di sini)

```
import librosa
import soundfile as sf

# Path file audio
audio_path = "/content/data/Westlife_-_You_Raise_Me_Up_CeeNaija.com_.mp3"

# Muat audio
y, sr = librosa.load(audio_path, sr=None, mono=False) # sr=None biar sample rate asli dipakai

# Metadata dasar
duration = librosa.get_duration(y=y, sr=sr) # durasi dalam detik
channels = 1 if y.ndim == 1 else y.shape[0] # mono/stereo
total_samples = y.shape[-1]

print("=== Metadata Audio ===")
print(f"Sample rate      : {sr} Hz")
print(f"Durasi           : {duration:.2f} detik")
print(f"Jumlah kanal      : {channels} ({'Mono' if channels==1 else 'Stereo'})")
print(f"Jumlah total sampel: {total_samples}")
```

```
=== Metadata Audio ===
Sample rate      : 48000 Hz
Durasi           : 242.71 detik
Jumlah kanal      : 2 (Stereo)
Jumlah total sampel: 11649916
```

✓ 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.)

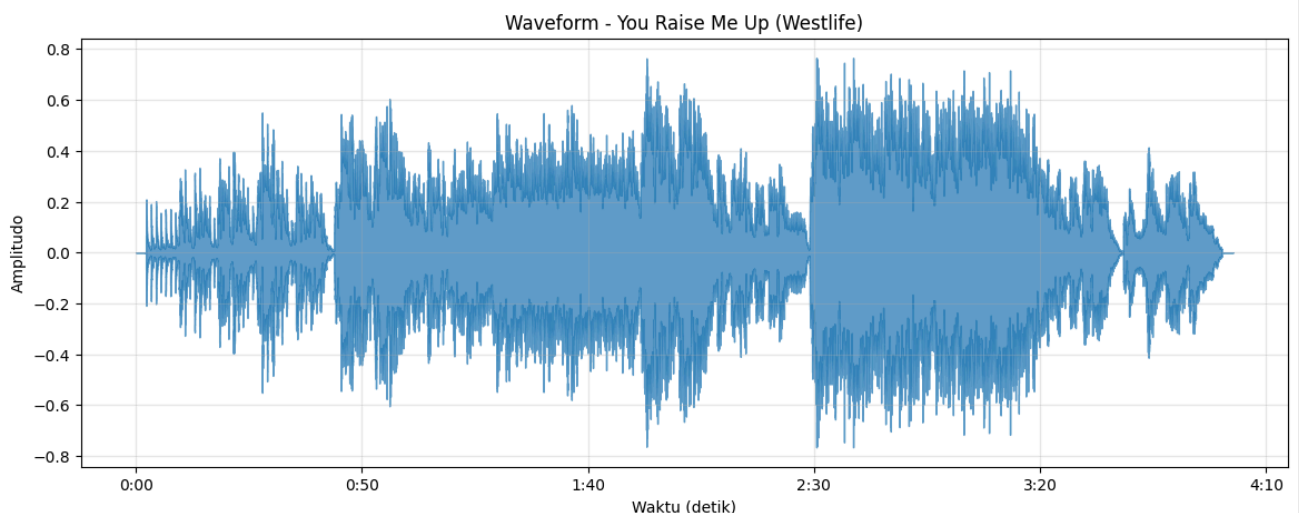
(Tambahkan sel kode di sini)

```
import librosa
import librosa.display
import matplotlib.pyplot as plt
import numpy as np

# Path file audio
audio_path = "data/Westlife_-_You_Raise_Me_Up_CeeNaija.com_.mp3"

# Muat audio
y, sr = librosa.load(audio_path, sr=None)

# Plot waveform
plt.figure(figsize=(14, 5))
librosa.display.waveshow(y, sr=sr, alpha=0.7)
plt.title("Waveform - You Raise Me Up (Westlife)")
plt.xlabel("Waktu (detik)")
plt.ylabel("Amplitudo")
plt.grid(alpha=0.3)
plt.show()
```



Analisis Waveform

Dari waveform lagu *You Raise Me Up*, terlihat beberapa pola:

- Amplitudo meningkat pada bagian chorus sehingga menandakan bagian lagu yang lebih kuat (dinamis).
- Pada awal lagu amplitudo relatif kecil, menandakan intro yang pelan.
- Variasi amplitudo yang signifikan menunjukkan adanya perpaduan vokal dan instrumen.
- Pola naik-turun amplitudo memberikan gambaran ritme dan dinamika lagu.

Kesimpulan: Waveform membantu melihat intensitas dan dinamika audio, namun belum memberikan informasi detail tentang frekuensi.

✓ A4. TODO: Spectrogram log-dB

Instruksi: Hitung STFT dan tampilkan spectrogram dalam skala log-dB:

- Gunakan parameter standar ($n_{\text{fft}}=1024$, $\text{hop_length}=256$)
- Tampilkan dengan colorbar

- Label sumbu: waktu (detik) dan frekuensi (Hz)

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

(Tambahkan sel kode di sini)

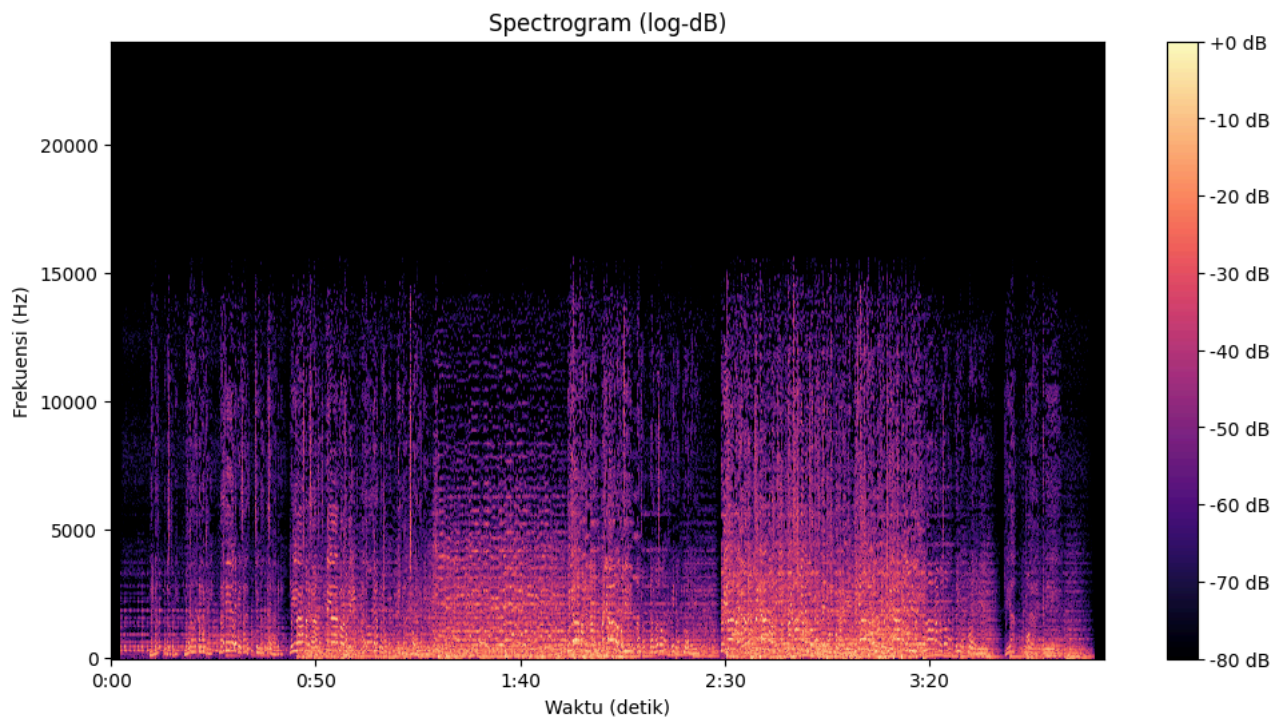
```
import librosa
import librosa.display
import matplotlib.pyplot as plt
import numpy as np

# Load audio
y, sr = librosa.load(audio_path, sr=None)

# Hitung STFT
D = librosa.stft(y, n_fft=1024, hop_length=256)

# Konversi ke skala log-dB
S_db = librosa.amplitude_to_db(np.abs(D), ref=np.max)

# Plot spectrogram
plt.figure(figsize=(12, 6))
librosa.display.specshow(S_db, sr=sr, hop_length=256,
                        x_axis='time', y_axis='hz', cmap='magma')
plt.colorbar(format="%+2.f dB")
plt.title("Spectrogram (log-dB)")
plt.xlabel("Waktu (detik)")
plt.ylabel("Frekuensi (Hz)")
plt.show()
```



Analisis: Waveform hanya memperlihatkan amplitudo terhadap waktu, sedangkan spectrogram log-dB menunjukkan distribusi energi pada berbagai frekuensi sepanjang waktu. Informasi ini lebih kaya, karena kita bisa melihat pola frekuensi, nada dominan, dan transisi suara/musik yang tidak terlihat di waveform.

✓ 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?

(Tambahkan sel kode di sini)

```

import librosa
import librosa.display
import matplotlib.pyplot as plt
import numpy as np

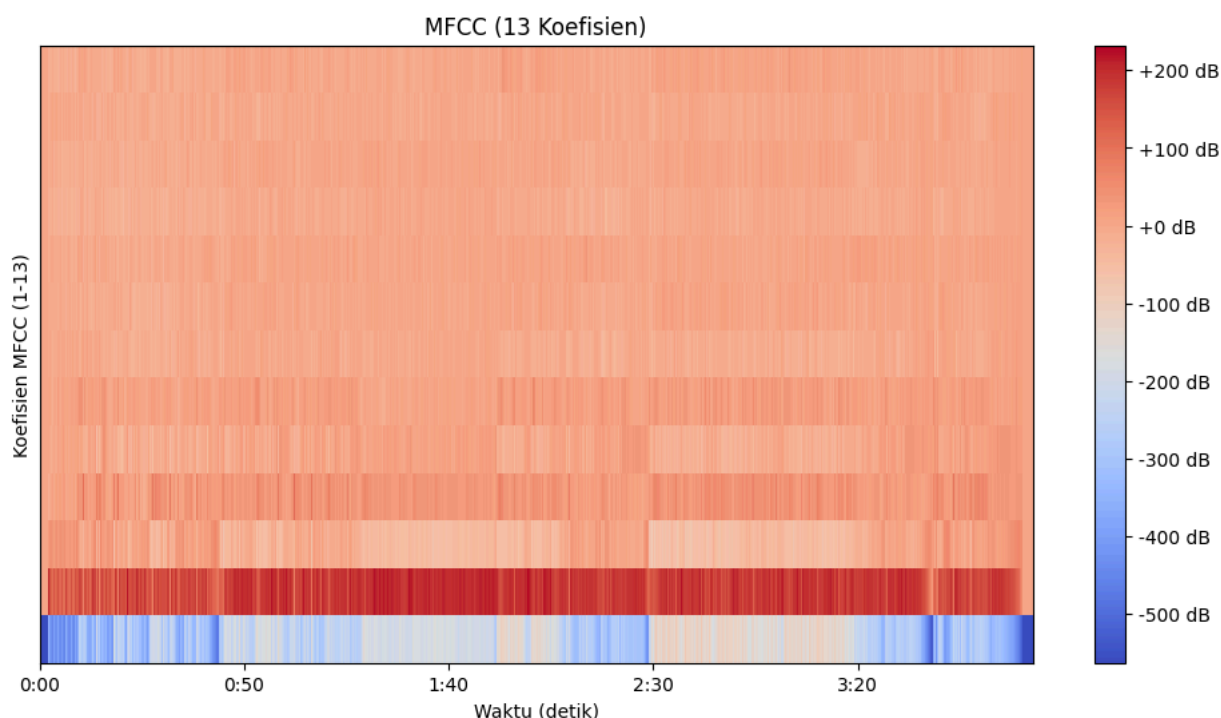
# Path audio (samakan dengan path file di folder 'data')
audio_path = "/content/data/Westlife_-_You_Raise_Me_Up_CeeNaija.com_.mp3"

# Load audio
y, sr = librosa.load(audio_path, sr=None)

# Hitung MFCC (13 koefisien)
mfccs = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=13)

# Plot heatmap MFCC
plt.figure(figsize=(12, 6))
librosa.display.specshow(mfccs, x_axis="time", sr=sr)
plt.colorbar(format="%+2.f dB")
plt.title("MFCC (13 Koefisien)")
plt.ylabel("Koefisien MFCC (1-13)")
plt.xlabel("Waktu (detik)")
plt.show()

```



Analisis: Pola MFCC terlihat berubah-ubah mengikuti dinamika lagu, terutama pada bagian vokal dan instrumen. Hal ini menunjukkan adanya variasi timbre suara. MFCC dapat merepresentasikan ciri unik suara penyanyi atau instrumen, sehingga bermanfaat untuk tugas seperti speech recognition atau klasifikasi musik.

A6. Analisis Ringkas (Wajib)

Jawab pertanyaan berikut:

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

Jawaban Anda: Waveform hanya memperlihatkan perubahan amplitudo terhadap waktu, sedangkan spectrogram memperlihatkan distribusi frekuensi dan energi sepanjang waktu. Spectrogram memberikan informasi yang lebih detail tentang pola nada dan harmonik.

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

Jawaban Anda: Dari MFCC terlihat pola yang berubah-ubah mengikuti dinamika lagu, menandakan variasi timbre suara. MFCC dapat digunakan untuk mengidentifikasi ciri khas suara, sehingga bermanfaat untuk aplikasi seperti speech recognition atau klasifikasi musik.

Start coding or [generate](#) with AI.

Bagian B — Gambar

B1. Deskripsi Data

TODO: Jelaskan gambar yang akan Anda analisis:

- Jenis gambar: altar gereja
- Sumber: foto pribadi (dokumentasi sendiri)
- Format file: jpg
- Alasan pemilihan: suka

Path file: data/gereja.jpg

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).

(Tambahkan sel kode di sini)

```
import cv2
import matplotlib.pyplot as plt

# Path gambar
img_path = "data/gereja.jpg"

# Baca gambar dengan OpenCV (default BGR)
img_bgr = cv2.imread(img_path)

# Konversi BGR -> RGB
img_rgb = cv2.cvtColor(img_bgr, cv2.COLOR_BGR2RGB)

# Tampilkan gambar
plt.figure(figsize=(8,6))
plt.imshow(img_rgb)
plt.title("Interior Gereja (RGB)")
plt.axis("off")
plt.show()
```

Interior Gereja (RGB)



Analisis

- Objek utama pada gambar adalah salib kayu di bagian tengah dengan mimbar dan hiasan altar.
- Pencahayaan terlihat cukup terang, didukung cahaya lampu chandelier serta cahaya alami dari samping.
- Komposisi warna didominasi oleh coklat (dinding dan kayu), putih (tembok dan bunga), hijau (altar), serta sedikit biru (logo gereja).

✓ B3. TODO: Informasi Dasar

Instruksi: Tampilkan informasi metadata gambar:

- Dimensi (Height x 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.

(Tambahkan sel kode di sini)

```
import cv2
from PIL import Image
import os
import numpy as np

# Load gambar
img_path = "/content/data/gereja.jpg"
img = cv2.imread(img_path)

# Dimensi (height x width x channel)
height, width, channels = img.shape

# Tipe data
dtype = img.dtype

# Ukuran file dalam memori (perkiraan)
memory_size = img.nbytes / 1024 # KB

# Mode warna (PIL)
pil_img = Image.open(img_path)
mode = pil_img.mode

print("Dimensi (Height x Width):", height, "x", width)
print("Jumlah Kanal:", channels)
print("Tipe Data (dtype):", dtype)
print("Mode Warna (PIL):", mode)
print("Ukuran File dalam Memori (KB):", round(memory_size, 2))
```

```
Dimensi (Height x Width): 1280 x 720
Jumlah Kanal: 3
Tipe Data (dtype): uint8
Mode Warna (PIL): RGB
Ukuran File dalam Memori (KB): 2700.0
```

Analisis

- Dimensi berguna supaya kita tahu ukuran gambar asli
- Jumlah kanal & mode warna penting karena ada analisis yang hanya butuh grayscale, ada juga yang harus tetap RGB.
- Tipe data menunjukkan bagaimana nilai piksel disimpan
- Ukuran file di memori membantu kita memperkirakan kebutuhan RAM.

✓ 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?

(Tambahkan sel kode di sini)

```
# Baca gambar dengan cv2 (BGR)
img = cv2.imread(img_path)

# Konversi BGR → RGB
```

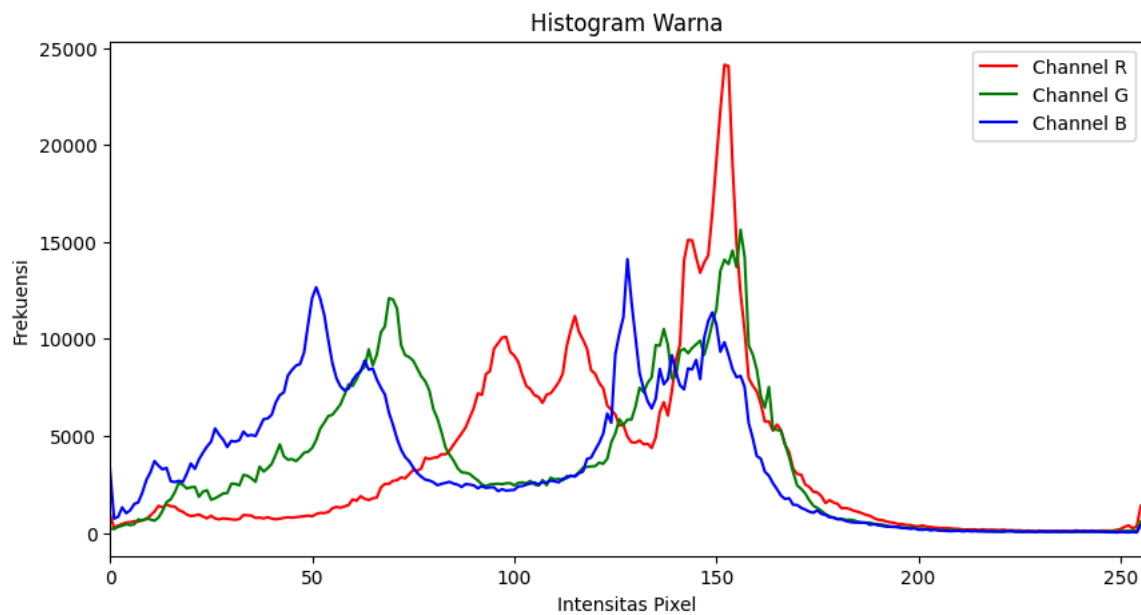


```
img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

# Plot histogram untuk setiap channel
colors = ('r', 'g', 'b')
plt.figure(figsize=(10,5))

for i, col in enumerate(colors):
    hist = cv2.calcHist([img_rgb], [i], None, [256], [0,256])
    plt.plot(hist, color=col, label=f'Channel {col.upper()}')
    plt.xlim([0,256])

plt.title("Histogram Warna")
plt.xlabel("Intensitas Pixel")
plt.ylabel("Frekuensi")
plt.legend()
plt.show()
```



Analisis

- Histogram menunjukkan sebaran intensitas warna merah, hijau, dan biru.
- Channel yang paling tinggi puncaknya bisa disebut warna dominan pada gambar.
- Kalau grafik menyebar rata dari 0–255 → kontras tinggi.
- Kalau grafik hanya menumpuk di range tertentu → gambar cenderung gelap/terang saja

Double-click (or enter) to edit

✓ B5. Analisis Ringkas (Wajib)

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 Anda: Histogram menunjukkan sebaran intensitas yang lebih tinggi pada channel (misalnya: biru dan hijau) dibanding merah, sehingga memberi kesan warna dominan pada gambar.

Start coding or [generate](#) with AI.

✓ Bagian C — Video

C1. Deskripsi Data

TODO: Jelaskan video yang akan Anda analisis:

- Jenis video: Dance
- Sumber: Tiktok

- Durasi target: 14 detik
- Alasan pemilihan: muncul di fyp

Path file: `data/Dance_Tiktok.mp4` (isi nama file Anda nanti di kode)

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.

(Tambahkan sel kode di sini)

```
import cv2

# Path video (ganti sesuai file kamu)
video_path = "data/Dance_Tiktok.mp4"

# Baca video
cap = cv2.VideoCapture(video_path)

if not cap.isOpened():
    print("Gagal membuka video, cek path.")
else:
    # Ambil metadata
    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)
    total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
    duration = total_frames / fps if fps > 0 else 0

    # Klasifikasi resolusi sederhana
    if width >= 3840 and height >= 2160:
        res_class = "4K"
    elif width >= 1920 and height >= 1080:
        res_class = "Full HD"
    elif width >= 1280 and height >= 720:
        res_class = "HD"
    else:
        res_class = "SD"

    print(f"Resolusi      : {width} x {height}")
    print(f"Frame rate     : {fps:.2f} fps")
    print(f"Total frame    : {total_frames}")
    print(f"Durasi         : {duration:.2f} detik")
    print(f"Klasifikasi    : {res_class}")

cap.release()
```

```
Resolusi      : 576 x 768
Frame rate    : 30.00 fps
Total frame   : 417
Durasi        : 13.90 detik
Klasifikasi   : SD
```

Analisis

- Resolusi penting karena menentukan ketajaman dan detail video.
- Frame rate (fps) mempengaruhi kelancaran gerakan, misalnya dance lebih enak dilihat di fps tinggi.
- Jumlah frame & durasi penting untuk estimasi lama pemrosesan atau pemotongan video.
- Klasifikasi resolusi membantu mengetahui apakah video cocok untuk ditampilkan di layar besar atau hanya cukup untuk media sosial.

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

Instruksi: Ambil dan tampilkan 3 frame representatif:

- Frame pertama (index 0)
- Frame tengah (index `~total_frame/2`)
- Frame terakhir (index `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.

(Tambahkan sel kode di sini)

```
cap = cv2.VideoCapture(video_path)

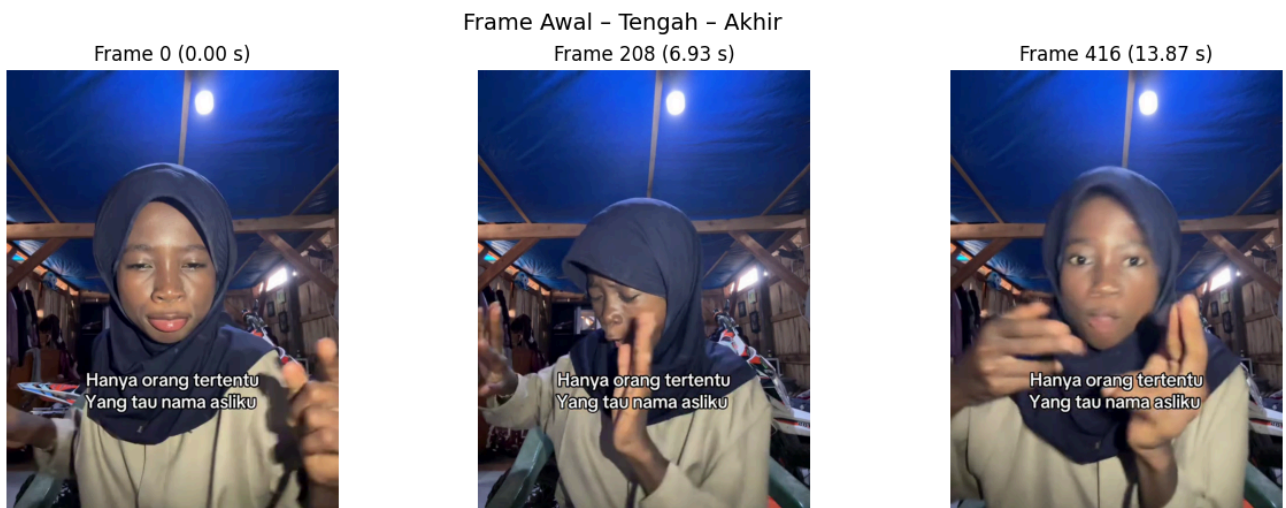
if not cap.isOpened():
    print("⚠️ Gagal membuka video, cek path.")
else:
    total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
    fps = cap.get(cv2.CAP_PROP_FPS)

    # Tentukan indeks frame
    frame_indices = [0, total_frames//2, total_frames-1]
    frames = []
    timestamps = []

    for idx in frame_indices:
        cap.set(cv2.CAP_PROP_POS_FRAMES, idx) # Pindah ke frame tertentu
        ret, frame = cap.read()
        if ret:
            # Konversi BGR → RGB
            frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
            frames.append(frame_rgb)
            timestamps.append(idx / fps)
        else:
            frames.append(None)
            timestamps.append(None)

    cap.release()

# Plot hasil
plt.figure(figsize=(15,5))
for i, (f, t) in enumerate(zip(frames, timestamps)):
    if f is not None:
        plt.subplot(1,3,i+1)
        plt.imshow(f)
        plt.title(f"Frame {frame_indices[i]} ({t:.2f} s)")
        plt.axis("off")
plt.suptitle("Frame Awal - Tengah - Akhir", fontsize=14)
plt.show()
```



Analisis

- Frame awal biasanya menampilkan kondisi pembuka (misalnya pose awal sebelum gerakan).
- Frame tengah memperlihatkan bagian inti gerakan (posisi tubuh, ekspresi, atau latar paling aktif).
- Frame akhir menunjukkan penutup (pose akhir, transisi, atau perubahan suasana).

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 Anda: Fps dan resolusi video sudah sesuai untuk konten media sosial seperti TikTok. Resolusi (HD/Full HD) cukup tajam untuk layar ponsel, dan fps yang stabil membuat gerakan dance terlihat halus. Jadi kualitasnya sudah memadai tanpa perlu spesifikasi lebih tinggi seperti 4K.

Perbandingan & Kesimpulan

Perbandingan Representasi Media

TODO: Bandingkan secara ringkas representasi dan visualisasi ketiga media:

Audio (1D - Temporal)

- Representasi: Sinyal 1 dimensi terhadap waktu (amplitudo vs waktu).
- Visualisasi utama: Waveform, spektrogram.
- Informasi yang diperoleh: Pola amplitudo, frekuensi dominan, intensitas, ritme.

Gambar (2D - Spasial)

- Representasi: Matriks 2D berisi nilai pixel (intensitas atau RGB).
- Visualisasi utama: Tampilan gambar, histogram warna.
- Informasi yang diperoleh: Warna dominan, kontras, bentuk/objek dalam gambar.

Video (2D + Waktu - Spatio-temporal)

- Representasi: Urutan frame gambar (2D) yang berubah seiring waktu (t).
- Visualisasi utama: Tampilan frame, timeline, metadata (fps, durasi).
- Informasi yang diperoleh: Perubahan gerakan, dinamika objek, alur peristiwa.

Refleksi Pembelajaran

3 Poin yang Saya Pelajari:

1. Ternyata audio, gambar, dan video punya cara representasi yang beda-beda, jadi nggak bisa dianalisis dengan pendekatan yang sama.
2. Metadata kayak resolusi, fps, atau dimensi itu penting banget buat ngerti kualitas dan kegunaan media.
3. Visualisasi sederhana (waveform, histogram, frame video) bisa langsung kasih gambaran isi data tanpa harus lihat keseluruhannya.

2 Hal yang Masih Membingungkan/Ingin Diperdalam:

1. Masih penasaran gimana caranya ekstraksi fitur audio yang lebih lanjut, misalnya MFCC buat pengenalan suara.
 - 2.
-

Sumber Data & Referensi

TODO: Cantumkan semua sumber data dan referensi yang digunakan:

- **Audio:** <https://www.ceenaija.com/mp3-download-you-raise-me-up-stand-on-mountains-westlife-audio-lyrics/>
- **Gambar:** Galeri HP
- **Video:** https://www.tiktok.com/@lilisswoyzy/video/7517298701199609096?is_from_webapp=1&sender_device=pc
- **Referensi teknis:** <https://www.youtube.com/@ProgrammerZamanNow>





Rubrik Penilaian

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

Detail Kriteria Penilaian

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

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

Analisis & Interpretasi (30%)

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

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

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





Proporsi Penilaian Total

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

Aturan Kejujuran Akademik





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

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

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

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

Panduan Pengumpulan

Berkas yang Harus Dikumpulkan

Wajib:

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

Informasi Pengumpulan

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

Export ke PDF:

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