



Nama: **Tawakkal Rabbani M(122140029)**
Environment untuk Multimedia

Tugas Ke: **Worksheet 1: Setup Python**

Mata Kuliah: **Sistem Teknologi Multimedia (IF25-40305)**

Tanggal: August 29, 2025

1 Tujuan Pembelajaran

Setelah menyelesaikan worksheet ini, mahasiswa diharapkan mampu:

- Memahami pentingnya manajemen environment Python untuk pengembangan multimedia
- Menginstall dan mengkonfigurasi Python environment menggunakan conda, venv, atau uv
- Menginstall library-library Python yang diperlukan untuk multimedia processing
- Memverifikasi instalasi dengan mengimpor dan menguji library multimedia
- Mendokumentasikan proses konfigurasi dan hasil pengujian dalam format \LaTeX

2 Latar Belakang

Python telah menjadi bahasa pemrograman yang sangat populer untuk multimedia processing karena memiliki ekosistem library yang sangat kaya. Namun, untuk dapat bekerja dengan multimedia secara efektif, kita perlu mengatur environment Python dengan benar dan menginstall library-library yang tepat.

Manajemen environment Python sangat penting untuk:

- Menghindari konflik antar library (dependency conflict)
- Memastikan reproducibility dari project
- Memudahkan kolaborasi antar developer
- Memisahkan project yang berbeda dengan requirement yang berbeda

3 Instruksi Tugas

3.1 Persiapan

Sebelum memulai, pastikan Anda telah:

- Menginstall Python 3.8 atau lebih baru di sistem Anda
- Memilih salah satu tool manajemen environment: **conda**, **venv**, atau **uv**
- Membuka terminal/command prompt
- Menyiapkan dokumen \LaTeX ini untuk dokumentasi

3.2 Bagian 1: Membuat Environment Python

Pilih **SALAH SATU** dari tiga opsi berikut dan ikuti langkah-langkahnya:

3.2.1 Opsi 1: Menggunakan Conda (Direkomendasikan untuk pemula)

Jalankan perintah berikut di terminal:

```
1 # Membuat environment baru dengan nama 'multimedia'
2 conda create -n multimedia python=3.11
3
4 # Mengaktifkan environment
5 conda activate multimedia
6
7 # Verifikasi environment aktif
8 conda info --envs
```

Kode 1: Membuat environment dengan Conda

3.2.2 Opsi 2: Menggunakan venv (Built-in Python)

```
1 # Membuat environment baru
2 python3 -m venv multimedia-env
3
4 # Mengaktifkan environment (Linux/Mac)
5 source multimedia-env/bin/activate
6
7 # Mengaktifkan environment (Windows)
8 # multimedia-env\Scripts\activate
9
10 # Verifikasi environment aktif
11 which python
```

Kode 2: Membuat environment dengan venv

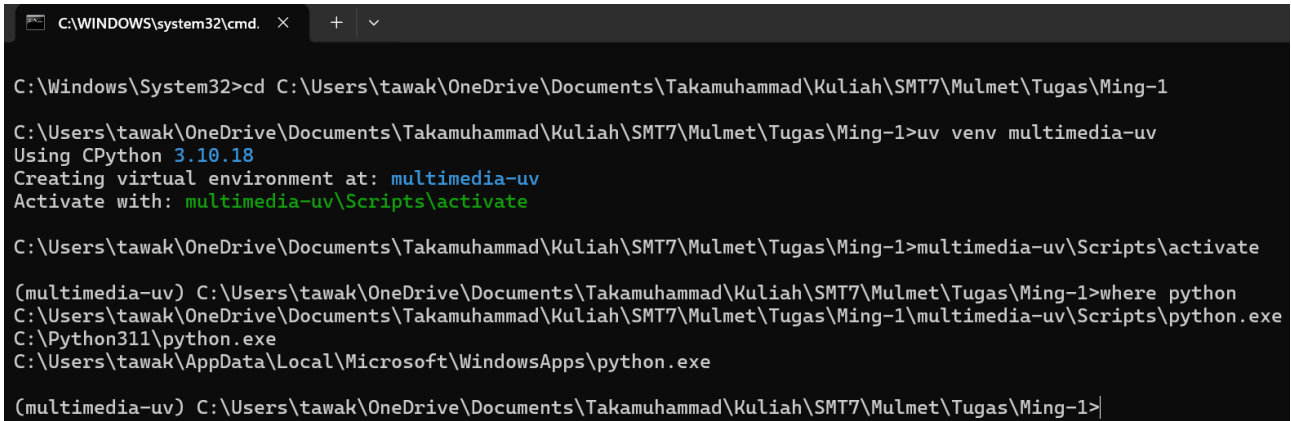
3.2.3 Opsi 3: Menggunakan uv (Modern dan cepat)

```
1 # Install uv terlebih dahulu jika belum ada
2 # pip install uv
3
4 # Membuat environment baru
5 uv venv multimedia-uv
6
7 # Mengaktifkan environment (Linux/Mac)
8 source multimedia-uv/bin/activate
9
10 # Mengaktifkan environment (Windows)
11 # multimedia-uv\Scripts\activate
12
13 # Verifikasi environment aktif
14 which python
```

Kode 3: Membuat environment dengan uv

Dokumentasikan di sini:

- Tool manajemen environment yang Anda pilih: **[Menggunakan uv]**
- Screenshot atau copy-paste output dari perintah verifikasi environment



```
C:\WINDOWS\system32\cmd. X + v

C:\Windows\System32>cd C:\Users\tawak\OneDrive\Documents\Takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1

C:\Users\tawak\OneDrive\Documents\Takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>uv venv multimedia-uv
Using CPython 3.10.18
Creating virtual environment at: multimedia-uv
Activate with: multimedia-uv\Scripts\activate

C:\Users\tawak\OneDrive\Documents\Takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>multimedia-uv\Scripts\activate

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>where python
C:\Users\tawak\OneDrive\Documents\Takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1\multimedia-uv\Scripts\python.exe
C:\Python311\python.exe
C:\Users\tawak\AppData\Local\Microsoft\WindowsApps\python.exe

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>
```

Gambar 1: Output verifikasi dari environment uv

3.3 Bagian 2: Instalasi Library Multimedia

Setelah environment aktif, install library-library berikut:

3.3.1 Library Audio Processing

```
1 # Untuk conda:
2 conda install -c conda-forge librosa soundfile scipy
3
4 # Untuk pip (venv/uv):
5 pip install librosa soundfile scipy
```

Kode 4: Instalasi library audio

3.3.2 Library Image Processing

```
1 # Untuk conda:
2 conda install -c conda-forge opencv pillow scikit-image matplotlib
3
4 # Untuk pip (venv/uv):
5 pip install opencv-python pillow scikit-image matplotlib
```

Kode 5: Instalasi library image

3.3.3 Library Video Processing

```
1 # Untuk conda:
2 conda install -c conda-forge ffmpeg
3 pip install moviepy
4
5 # Untuk pip (venv/uv):
6 pip install moviepy
```

Kode 6: Instalasi library video

3.3.4 Library General Purpose

```
1 # Untuk conda:
2 conda install numpy pandas jupyter
3
```

```

4 # Untuk pip (venv/uv):
5 pip install numpy pandas jupyter

```

Kode 7: Instalasi library umum

Dokumentasikan di sini:

- Perintah instalasi yang Anda gunakan

```

1 # Perintah instalasi audio processing
2 uv pip install librosa soundfile scipy
3
4 # Perintah instalasi image processing
5 uv pip install opencv-python pillow scikit-image matplotlib
6
7 # Perintah instalasi video processing
8 uv pip install moviepy
9
10 # Perintah instalasi library umum
11 uv pip install numpy pandas jupyter
12

```

Kode 8: Perintah instalasi library multimedia

- Screenshot proses instalasi atau output sukses

```

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>uv pip install librosa soundfile scipy
Using Python 3.10.18 environment at: multimedia-uv
Resolved 25 packages in 4.06s
* Preparing packages... (11/15)
scikit-learn ----- 4.59 MiB/8.49 MiB
numpy ----- 4.73 MiB/12.31 MiB
llvmlite ----- 4.64 MiB/28.93 MiB
scipy ----- 4.39 MiB/39.35 MiB

```

Gambar 2: Output proses instalasi audio processing

```

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>uv pip install opencv-python pillow scikit-image matplotlib
Using Python 3.10.18 environment at: multimedia-uv
Resolved 18 packages in 667ms
Prepared 5 packages in 20.03s
Installed 14 packages in 477ms
+ contourpy==1.3.2
+ cycler==0.12.1
+ fonttools==4.59.2
+ imageio==2.37.0
+ kiwisolver==1.4.9
+ matplotlib==3.10.5
+ networkx==3.4.2
+ opencv-python==4.12.0.88
+ pillow==11.3.0
+ pyparsing==3.2.3
+ python-dateutil==2.9.0.post0
+ scikit-image==0.25.2
+ six==1.17.0
+ tifffile==2025.5.10
(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>

```

Gambar 3: Output proses instalasi image processing

```

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>uv pip install moviepy
Using Python 3.10.18 environment at: multimedia-uv
Resolved 10 packages in 408ms
* Preparing packages... (3/4)
imageio-ffmpeg ----- 8.50 MiB/29.80 MiB

```

Gambar 4: Output proses instalasi video processing

```
(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>uv pip install numpy pandas jupyter
Using Python 3.10.18 environment at: multimedia-uv
Resolved 105 packages in 1.07s
* Preparing packages... (8/10)
widgetsnbextension ----- 1.02 MiB/2.09 MiB
notebook ----- 991.84 KiB/13.63 MiB
```

Gambar 5: Output proses instalasi library umum

- Daftar library yang berhasil diinstall dengan versinya

```
Installed 25 packages in 852ms
+ audioread==3.0.1
+ certifi==2025.8.3
+ cffi==1.17.1
+ charset-normalizer==3.4.3
+ decorator==5.2.1
+ idna==3.10
+ joblib==1.5.2
+ lazy-loader==0.4
+ librosa==0.11.0
+ llvmlite==0.44.0
+ msgpack==1.1.1
+ numba==0.61.2
+ numpy==2.2.6
+ packaging==25.0
+ platformdirs==4.4.0
+ pooch==1.8.2
+ pycparser==2.22
+ requests==2.32.5
+ scikit-learn==1.7.1
+ scipy==1.15.3
+ soundfile==0.13.1
+ soxr==0.5.0.post1
+ threadpoolctl==3.6.0
+ typing-extensions==4.15.0
+ urllib3==2.5.0

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>
```

Gambar 6: Daftar library audio processing

```
Installed 14 packages in 477ms
+ contourpy==1.3.2
+ cycler==0.12.1
+ fonttools==4.59.2
+ imageio==2.37.0
+ kiwisolver==1.4.9
+ matplotlib==3.10.5
+ networkx==3.4.2
+ opencv-python==4.12.0.88
+ pillow==11.3.0
+ pyparsing==3.2.3
+ python-dateutil==2.9.0.post0
+ scikit-image==0.25.2
+ six==1.17.0
+ tifffile==2025.5.10

(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>
```

Gambar 7: Daftar library image processing

```
Installed 6 packages in 82ms
+ colorama==0.4.6
+ imageio-ffmpeg==0.6.0
+ moviepy==2.2.1
+ proglog==0.1.12
+ python-dotenv==1.1.1
+ tqdm==4.67.1
(multimedia-uv) C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1>
```

Gambar 8: Daftar library video processing

```
Installed 90 packages in 3.55s
+ anyio==4.10.0
+ argon2-cffi==25.1.0
+ argon2-cffi-bindings==25.1.0
+ arrow==1.3.0
+ asttokens==3.0.0
+ async-lru==2.0.5
+ attrs==25.3.0
+ babel==2.17.0
+ beautifulsoup4==4.13.5
+ bleach==6.2.0
+ comm==0.2.3
+ debugpy==1.8.16
+ defusedxml==0.7.1
+ exceptiongroup==1.3.0
+ executing==2.2.0
+ fastjsonschema==2.21.2
+ fqdn==1.5.1
+ h11==0.16.0
+ httpcore==1.0.9
+ httpx==0.28.1
+ ipykernel==6.30.1
+ ipython==8.37.0
+ ipywidgets==8.1.7
+ isoduration==20.11.0
+ jedi==0.19.2
+ jinja2==3.1.6
```

Gambar 9: Daftar library umum

3.4 Bagian 3: Verifikasi Instalasi

Buat file Python sederhana untuk menguji semua library yang telah diinstall:

Jalankan script dan dokumentasikan hasilnya:

3.5 Bagian 4: Simple Test dengan Sample Code

Buat dan jalankan contoh sederhana untuk setiap kategori multimedia:

3.5.1 Test Audio Processing

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # Generate simple sine wave
5 duration = 2 # seconds
6 sample_rate = 44100
7 frequency = 440 # A4 note
8
9 t = np.linspace(0, duration, int(sample_rate * duration))
10 audio_signal = np.sin(2 * np.pi * frequency * t)
11
12 # Plot waveform
13 plt.figure(figsize=(10, 4))
14 plt.plot(t[:1000], audio_signal[:1000]) # Plot first 1000 samples
15 plt.title('Sine Wave (440 Hz)')
16 plt.xlabel('Time (s)')
17 plt.ylabel('Amplitude')
18 plt.grid(True)
19 plt.savefig('sine_wave_test.png', dpi=150, bbox_inches='tight')
20 plt.show()
21
22 print(f"Generated {duration}s sine wave at {frequency}Hz")
23 print(f"Sample rate: {sample_rate}Hz")
24 print(f"Total samples: {len(audio_signal)}")

```

Kode 9: Test audio processing sederhana

3.5.2 Test Image Processing

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 from PIL import Image
4
5 # Create a simple test image
6 width, height = 400, 300
7 image = np.zeros((height, width, 3), dtype=np.uint8)
8
9 # Add some patterns
10 image[:, :width//3, 0] = 255 # Red section
11 image[:, width//3:2*width//3, 1] = 255 # Green section
12 image[:, 2*width//3:, 2] = 255 # Blue section
13
14 # Add a white circle in the center
15 center_x, center_y = width//2, height//2
16 radius = 50
17 Y, X = np.ogrid[:height, :width]
18 mask = (X - center_x)**2 + (Y - center_y)**2 <= radius**2
19 image[mask] = [255, 255, 255]
20
21 # Display and save
22 plt.figure(figsize=(8, 6))
23 plt.imshow(image)
24 plt.title('Test Image with RGB Stripes and White Circle')
25 plt.axis('off')
26 plt.savefig('test_image.png', dpi=150, bbox_inches='tight')
27 plt.show()
28
29 print(f"Created test image: {width}x{height} pixels")
30 print(f"Image shape: {image.shape}")

```

```
31 print(f"Image dtype: {image.dtype}")
```

Kode 10: Test image processing sederhana

Dokumentasikan hasil eksekusi:

- Screenshot output dari kedua script di atas
- Gambar yang dihasilkan (sine_wave_test.png dan test_image.png)
- Error message jika ada dan cara mengatasinya

4 Bagian Laporan

4.1 Output Verifikasi Instalasi

Copy-paste output lengkap dari script **test_multimedia.py** di sini:

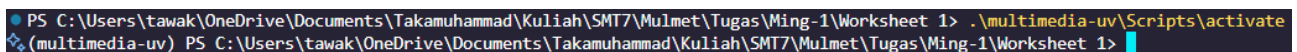
```
1 (multimedia-uv) PS C:\Users\tawak\OneDrive\Documents\takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1\
  Worksheet 1> uv run python .\test_multimedia.py
2 =====
3 Uji Instalasi Library Multimedia
4 =====
5 librosa terinstall (versi: 0.11.0)
6 soundfile terinstall (versi: 0.13.1)
7 scipy terinstall (versi: 1.15.3)
8 opencv-python terinstall (versi: 4.12.0)
9 pillow terinstall (versi: 11.3.0)
10 scikit-image terinstall (versi: 0.25.2)
11 matplotlib terinstall (versi: 3.10.5)
12 moviepy terinstall (versi: 2.1.2)
13 numpy terinstall (versi: 2.2.6)
14 pandas terinstall (versi: 2.3.2)
15 jupyter terinstall (versi: No version info)
16 =====
17 Pengujian selesai
18 =====
```

Kode 11: Output verifikasi instalasi

4.2 Screenshot Hasil Test

Sisipkan screenshot atau gambar hasil dari:

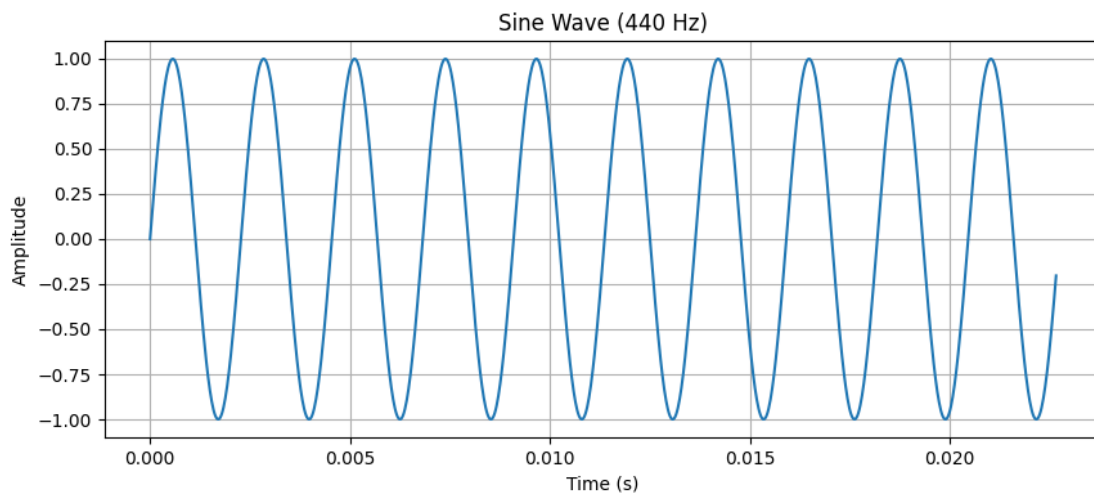
- Terminal/command prompt yang menunjukkan environment aktif



```
PS C:\Users\tawak\OneDrive\Documents\takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1\Worksheet 1> .\multimedia-uv\Scripts\activate
(multimedia-uv) PS C:\Users\tawak\OneDrive\Documents\takamhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1\Worksheet 1>
```

Gambar 10: Output environment uv aktif

- Output dari script test audio (sine wave plot)

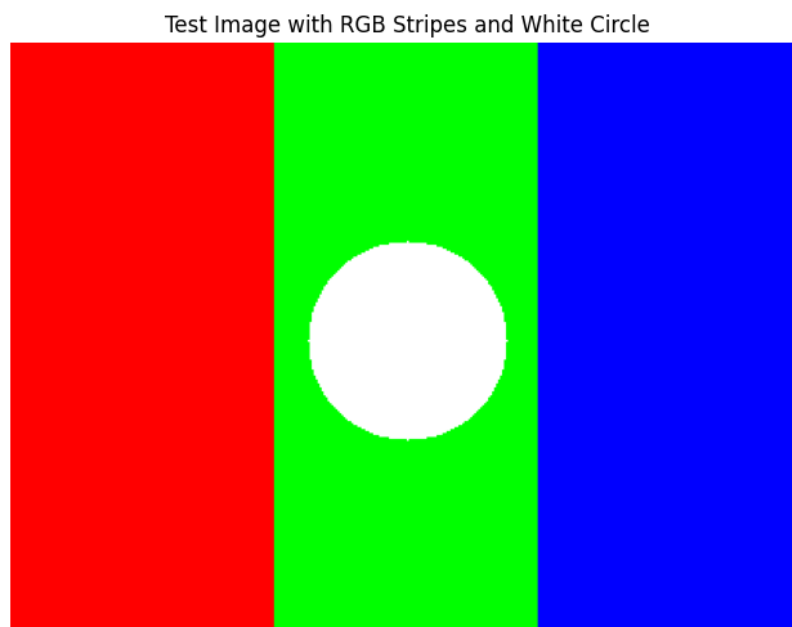


Gambar 11: Output gambar dari script test audio (sine wave plot)

```
(multimedia-uv) PS C:\Users\tawak\OneDrive\Documents\takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1\Worksheet 1> uv run python test_audio.py
Generated 2s sine wave at 440Hz
Sample rate: 44100Hz
Total samples: 88200
```

Gambar 12: Output terminal dari script test audio (sine wave plot)

- Output dari script test image (RGB stripes dengan circle)



Gambar 13: Output gambar dari script test image (RGB stripes dengan circle)

```
(multimedia-uv) PS C:\Users\tawak\OneDrive\Documents\Takamuhammad\Kuliah\SMT7\Mulmet\Tugas\Ming-1\Worksheet 1> uv run python test_image.py
Created test image: 400x300 pixels
Image shape: (300, 400, 3)
Image dtype: uint8
```

Gambar 14: Output terminal dari script test image (RGB stripes dengan circle)

4.3 Analisis dan Refleksi

Jawab pertanyaan berikut:

1. Mengapa penting menggunakan environment terpisah untuk project multimedia?

Menggunakan environment terpisah penting untuk mencegah konflik antar library, menjaga konsistensi versi agar project dapat direproduksi, serta memudahkan kolaborasi dan troubleshooting. Dengan isolasi ini, setiap project multimedia dapat berjalan mandiri tanpa mengganggu project lain.

2. Apa perbedaan utama antara conda, venv, dan uv? Mengapa Anda memilih tool yang Anda gunakan?

Perbedaan utama dari conda, venv, dan uv adalah sebagai berikut:

- *Conda: lengkap, bisa mengelola library Python maupun non-Python, tapi relatif berat dan instalasinya lebih lama.*
- *venv: bawaan Python, ringan, tapi hanya membuat virtual environment tanpa fitur tambahan, sehingga manajemen paket tetap bergantung pada pip.*
- *uv: tool modern berbasis pip, sangat cepat, ringan, dan efisien dalam membuat serta mengelola environment.*

Saya memilih uv sebab lebih ringan dan cepat dibandingkan conda maupun venv, sehingga sesuai untuk kebutuhan project multimedia yang membutuhkan banyak library.

3. Library mana yang paling sulit diinstall dan mengapa?

Selama saya mencoba untuk menginstall semua library multimedia, tidak ada library yang sulit diinstall, namun ada library yang membutuhkan waktu yang lama yaitu Library General Purpose (numpy, pandas, jupyter) karena ukurannya yang besar dan banyak packages yang harus diunduh sekitar 90 packages.

4. Bagaimana cara mengatasi masalah dependency conflict jika terjadi?

Masalah dependency conflict dapat diatasi dengan membuat environment terpisah, menyesuaikan versi library sesuai kebutuhan, menggunakan file requirements.txt atau environment.yml untuk konsistensi, serta melakukan update atau downgrade paket secara selektif.

5. Jelaskan fungsi dari masing-masing library yang berhasil Anda install!

Berikut adalah fungsi dari library yang berhasil diinstall:

- *Librosa: untuk analisis dan pemrosesan sinyal audio, seperti ekstraksi fitur (MFCC, tempo, spectrogram).*
- *Soundfile: untuk membaca dan menulis file audio (format seperti WAV, FLAC).*
- *Scipy: menyediakan fungsi ilmiah, termasuk pemrosesan sinyal digital (filtering, Fourier transform).*
- *OpenCV (opencv-python): library populer untuk pemrosesan gambar dan video (deteksi wajah, edge detection, filtering).*

- *Pillow (PIL)*: untuk manipulasi gambar sederhana (*resize, crop, rotasi, format konversi*).
- *Scikit-image*: untuk pemrosesan gambar berbasis sains (*segmentasi, transformasi, analisis fitur*).
- *Matplotlib*: untuk visualisasi data, membuat grafik, serta menampilkan gambar.
- *MoviePy*: untuk editing video berbasis Python, seperti memotong, menggabungkan, menambahkan teks, efek, atau audio.
- *NumPy*: library dasar untuk komputasi numerik dengan array multidimensi yang efisien.
- *Pandas*: untuk manipulasi dan analisis data dalam bentuk tabel (*DataFrame*).
- *Jupyter*: menyediakan notebook interaktif untuk menjalankan kode Python, visualisasi, dan dokumentasi dalam satu tempat.

4.4 Troubleshooting

Dokumentasikan masalah yang Anda hadapi (jika ada) dan cara mengatasinya:

- **Masalah 1:** *Ketika saya mencoba instalasi library multimedia, awalnya semua package terinstall di global environment, bukan di environment uv yang saya buat.*
Solusi: *Jadi untuk mengatasi masalah tersebut, environment uv harus aktif terlebih dahulu, kemudian untuk perintah instalasinya harus diawali dengan perintah "uv" sebagai contoh "uv pip install librosa soundfile scipy".*
- **Masalah 2:** *Saya tidak mengetahui cara menyisipkan gambar di dalam dokumen LaTeX.*
Solusi: *Untuk menyisipkan gambar, gunakan command `\includegraphics`.*

5 Export Environment untuk Reproduksi

Sebagai langkah terakhir, export environment Anda agar dapat direproduksi:

5.1 Untuk Conda

```
1 conda env export > environment.yml
```

Kode 12: Export conda environment

5.2 Untuk venv/uv

```
1 pip freeze > requirements.txt
```

Kode 13: Export pip requirements

Copy-paste isi file `environment.yml` atau `requirements.txt` di sini:

```
1 anyio==4.10.0
2 argon2-cffi==25.1.0
3 argon2-cffi-bindings==25.1.0
4 arrow==1.3.0
5 asttokens==3.0.0
6 async-lru==2.0.5
7 attrs==25.3.0
8 audioread==3.0.1
9 babel==2.17.0
10 beautifulsoup4==4.13.5
11 bleach==6.2.0
```

```
12 certifi==2025.8.3
13 cffi==1.17.1
14 charset-normalizer==3.4.3
15 colorama==0.4.6
16 comm==0.2.3
17 contourpy==1.3.2
18 cycler==0.12.1
19 debugpy==1.8.16
20 decorator==5.2.1
21 defusedxml==0.7.1
22 exceptiongroup==1.3.0
23 executing==2.2.0
24 fastjsonschema==2.21.2
25 fonttools==4.59.2
26 fqdn==1.5.1
27 h11==0.16.0
28 httpcore==1.0.9
29 httpx==0.28.1
30 idna==3.10
31 imageio==2.37.0
32 imageio-ffmpeg==0.6.0
33 ipykernel==6.30.1
34 ipython==8.37.0
35 ipywidgets==8.1.7
36 isoduration==20.11.0
37 jedi==0.19.2
38 jinja2==3.1.6
39 joblib==1.5.2
40 json5==0.12.1
41 jsonpointer==3.0.0
42 jsonschema==4.25.1
43 jsonschema-specifications==2025.4.1
44 jupyter==1.1.1
45 jupyter-client==8.6.3
46 jupyter-console==6.6.3
47 jupyter-core==5.8.1
48 jupyter-events==0.12.0
49 jupyter-lsp==2.3.0
50 jupyter-server==2.17.0
51 jupyter-server-terminals==0.5.3
52 jupyterlab==4.4.6
53 jupyterlab-pygments==0.3.0
54 jupyterlab-server==2.27.3
55 jupyterlab-widgets==3.0.15
56 kiwisolver==1.4.9
57 lark==1.2.2
58 lazy-loader==0.4
59 librosa==0.11.0
60 llvmlite==0.44.0
61 markupsafe==3.0.2
62 matplotlib==3.10.5
63 matplotlib-inline==0.1.7
64 mistune==3.1.3
65 moviepy==2.2.1
66 msgpack==1.1.1
67 nbclient==0.10.2
68 nbconvert==7.16.6
69 nbformat==5.10.4
70 nest-asyncio==1.6.0
71 networkx==3.4.2
72 notebook==7.4.5
73 notebook-shim==0.2.4
```

```
74 numba==0.61.2
75 numpy==2.2.6
76 opencv-python==4.12.0.88
77 overrides==7.7.0
78 packaging==25.0
79 pandas==2.3.2
80 pandocfilters==1.5.1
81 parso==0.8.5
82 pillow==11.3.0
83 platformdirs==4.4.0
84 pooch==1.8.2
85 proglog==0.1.12
86 prometheus-client==0.22.1
87 prompt-toolkit==3.0.52
88 psutil==7.0.0
89 pure-eval==0.2.3
90 pycparser==2.22
91 pygments==2.19.2
92 pyparsing==3.2.3
93 python-dateutil==2.9.0.post0
94 python-dotenv==1.1.1
95 python-json-logger==3.3.0
96 pytz==2025.2
97 pywin32==311
98 pywinpty==3.0.0
99 pyyaml==6.0.2
100 pyzmq==27.0.2
101 referencing==0.36.2
102 requests==2.32.5
103 rfc3339-validator==0.1.4
104 rfc3986-validator==0.1.1
105 rfc3987-syntax==1.1.0
106 rpds-py==0.27.1
107 scikit-image==0.25.2
108 scikit-learn==1.7.1
109 scipy==1.15.3
110 send2trash==1.8.3
111 setuptools==80.9.0
112 six==1.17.0
113 sniffio==1.3.1
114 soundfile==0.13.1
115 soupsieve==2.8
116 soxr==0.5.0.post1
117 stack-data==0.6.3
118 terminado==0.18.1
119 threadpoolctl==3.6.0
120 tifffile==2025.5.10
121 tinycss2==1.4.0
122 tomli==2.2.1
123 tornado==6.5.2
124 tqdm==4.67.1
125 traitlets==5.14.3
126 types-python-dateutil==2.9.0.20250822
127 typing-extensions==4.15.0
128 tzdata==2025.2
129 uri-template==1.3.0
130 urllib3==2.5.0
131 wcwidth==0.2.13
132 webcolors==24.11.1
133 webencodings==0.5.1
134 websocket-client==1.8.0
```

```
135 widgetsnbextension==4.0.14
```

Kode 14: Environment/Requirements file

6 Kesimpulan

Tuliskan kesimpulan Anda mengenai:

- Pengalaman setup Python environment untuk multimedia: *Ini adalah pengalaman pertama saya melakukan setup Python environment menggunakan uv. Awalnya saya mengalami kesulitan karena seperti mengaktifkan environment uv ternyata bermacam-macam caranya tergantung OS yang digunakan, pada laptop saya untuk mengaktifkan uv menggunakan perintah "multimedia-uv/Scripts/Activate". Selain itu juga saya mengalami kesulitan ketika menginstall library multimedia karena awalnya semua packages terinstall di global environment, ternyata perintah instalasinya harus diawali dengan perintah "uv". Terakhir ini adalah pengalaman pertama saya juga menggunakan LaTeX untuk membuat laporan, sehingga saya sering mengalami beberapa kesulitan seperti menyisipkan gambar dan lain-lain. Namun setelah saya mencari tahu melalui youtube dan chatgpt, saya bisa mengatasi masalah-masalah tersebut.*
- Persiapan untuk project multimedia selanjutnya: *Untuk persiapan project multimedia selanjutnya, mungkin saya akan mencoba memahami representasi dasar data multimedia (audio, gambar, dan video).*
- Saran untuk mahasiswa lain yang akan melakukan setup serupa: *Saran untuk mahasiswa lain yaitu pastikan memilih environment yang sesuai dengan spesifikasi Laptop/PC. Kemudian pastikan mengikuti langkah-langkah setup yang diberikan dengan teliti. Jika mengalami kesulitan, bisa mencari solusi melalui internet atau youtube.*

Pilihlah environment yang paling sesuai dengan spesifikasi Laptop/PC anda. Kemudian ikutilah langkah-langkah setup dengan teliti. Jika mengalami kesulitan, gunakan internet atau youtube untuk mencari solusi.

7 Referensi

1. [ChatGPT - Troubleshooting Environment uv](#)
2. [ChatGPT - Troubleshooting Worksheet 1 Multimedia](#)
3. [Youtube - Latex Tutorial](#)