



Nama: **Reynaldi Cristian Simamora (122140116)** Tugas Ke: **Worksheet 1: Setup Python Environment untuk Multimedia**

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

Tanggal: October 3, 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: **conda**
- Screenshot atau copy-paste output dari perintah verifikasi environment

Output : verifikasi instalasi

```
1 D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda activate multimedia
2
3 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda info --envs
4
5 # conda environments:
6 #
7 base                C:\ProgramData\miniconda3
8 handson             C:\Users\ANSEN\.conda\envs\handson
9 multimedia          * C:\Users\ANSEN\.conda\envs\multimedia
10 myenv               C:\Users\ANSEN\.conda\envs\myenv
11
12 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>
```

Kode 4: Output verifikasi instalasi Environment Melalui Conda

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 5: 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 6: 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 7: 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 8: Instalasi library umum

Dokumentasikan di sini:

- Perintah instalasi yang Anda gunakan

```

1 //Saya menggunakan instalansi via conda untuk keempat library di atas
2 conda install -c conda --forge librosa soundfile scipy
3 conda install -c conda-forge opencv pillow scikit-image matplotlib
4 conda install -c conda-forge ffmpeg
5 conda install numpy pandas jupyter
6

```

Kode 9: Instalasi library video

- Screenshot proses instalasi atau output sukses

```

1 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda install -c conda-forge
  librosa soundfile scipy
2 Retrieving notices: done
3 Channels:
4 - conda-forge
5 - defaults
6 Platform: win-64
7 Collecting package metadata (repodata.json): done
8 Solving environment: done
9 ==> WARNING: A newer version of conda exists. <==
10 current version: 25.3.1
11 latest version: 25.7.0
12 Please update conda by running
13 $ conda update -n base -c defaults conda
14
15 ## Package Plan ##
16
17 environment location: C:\Users\ANSEN\.conda\envs\multimedia
18
19 added / updated specs:
20 - librosa
21 - scipy
22 - soundfile
23 The following packages will be downloaded:
24
25 package | build
26 -----|-----
27 ca-certificates-2025.8.3 | h4c7d964_0 151 KB conda-forge
28 cpython-3.11.13 | py311hd8ed1ab_0 46 KB conda-forge
29 importlib-metadata-8.7.0 | pyhe01879c_1 34 KB conda-forge
30 jsonschema-with-format-nongpl-4.25.1 | he01879c_0 5 KB conda-forge
31 .... | .....
32 tomli-2.2.1 | pyhe01879c_2 21 KB conda-forge
33 zipp-3.23.0 | pyhd8ed1ab_0 22 KB conda-forge
34 -----|-----
35 Total: 137.6 MB
36 The following NEW packages will be INSTALLED:
37
38 cpython conda-forge/noarch::cpython-3.11.13-py311hd8ed1ab_0
39 importlib-metadata conda-forge/noarch::importlib-metadata-8.7.0-pyhe01879c_1
40 jsonschema-with-f~ conda-forge/noarch::jsonschema-with-format-nongpl-4.25.1-he01879c_0
41 jupyter_client conda-forge/noarch::jupyter_client-8.6.3-pyhd8ed1ab_1
42 .... | .....
43 zipp conda-forge/noarch::zipp-3.23.0-pyhd8ed1ab_0
44
45 The following packages will be UPDATED:
46
47 ca-certificates pkgs/main/win-64::ca-certificates-202~ --> conda-forge/noarch::ca-
  certificates-2025.8.3-h4c7d964_0

```

```

48  openssl                pkgs/main::openssl-3.0.17-h35632f6_0 --> conda-forge::openssl-3.5.2-
    h725018a_0
49  scipy                  pypi/pypi::scipy-1.16.1-pypi_0 --> conda-forge/win-64::scipy
    -1.16.2-py311h9a1c30b_0
50
51 The following packages will be SUPERSEDED by a higher-priority channel:
52
53  jupyterlab              pypi/pypi::jupyterlab-4.4.7-pypi_0 --> conda-forge/noarch::
    jupyterlab-4.4.7-pyhd8edlab_0
54
55
56 Proceed ([y]/n)? y
57
58 Downloading and Extracting Packages:
59
60 Preparing transaction: done
61 Verifying transaction: done
62 Executing transaction: done
63
64 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda install -c conda-forge
    opencv pillow scikit-image matplotlib
65 Channels:
66 - conda-forge
67 - defaults
68 Platform: win-64
69 Collecting package metadata (repodata.json): done
70 Solving environment: done
71
72 ==> WARNING: A newer version of conda exists. <==
73     current version: 25.3.1
74     latest version: 25.7.0
75
76 Please update conda by running
77
78     $ conda update -n base -c defaults conda
79
80 ## Package Plan ##
81
82 environment location: C:\Users\ANSEN\.conda\envs\multimedia
83
84 added / updated specs:
85 - matplotlib
86 - opencv
87 - pillow
88 - scikit-image
89
90 The following packages will be downloaded:
91
92 package | build
93 -----|-----
94 _libavif_api-1.3.0 | h57928b3_2 10 KB conda-forge
95 aom-3.9.1 | he0c23c2_0 1.9 MB conda-forge
96 blosc-1.21.6 | h4190f5b_0 100 KB
97 brotli-1.0.9 | hcfcfb64_9 20 KB conda-forge
98 .... | ...
99 -----
100 Total: 375.3 MB
101
102 The following NEW packages will be INSTALLED:
103
104 _libavif_api conda-forge/win-64::_libavif_api-1.3.0-h57928b3_2
105 aom conda-forge/win-64::aom-3.9.1-he0c23c2_0

```

```

106 ....
107 1-h2f0f97f_3
108 zlib-ng conda-forge/win-64::zlib-ng-2.0.7-hcfcfb64_0
109 zstd pkg/main/win-64::zstd-1.5.6-h8880b57_0
110
111 Proceed ([y]/n)? y
112
113 Downloading and Extracting Packages:
114 qtwebengine-6.7.3 | 146.1 MB | #####6 | 15%
115 opencv-4.10.0 | 40.0 MB | #
#####9
| 58%
116 qtwebengine-6.7.3 | 146.1 MB | #####7 | 15%
117 opencv-4.10.0 | 40.0 MB | #
#####1
| 58%
118 qtwebengine-6.7.3 | 146.1 MB | #####7 | 15%
119
120 Preparing transaction: done
121 Verifying transaction: done
122 Executing transaction: done
123
124 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda install -c conda-forge
ffmpeg
125 Channels:
126 - conda-forge
127 - defaults
128 Platform: win-64
129 Collecting package metadata (repodata.json): done
130 Solving environment: done
131
132
133 ==> WARNING: A newer version of conda exists. <==
134 current version: 25.3.1
135 latest version: 25.7.0
136
137 Please update conda by running
138
139 $ conda update -n base -c defaults conda
140
141 ## Package Plan ##
142
143 environment location: C:\Users\ANSEN\.conda\envs\multimedia
144
145 added / updated specs:
146 - ffmpeg
147
148
149 The following packages will be downloaded:
150
151 package | build
152 -----|-----
153 ffmpeg-4.3.1 | ha925a31_0 26.2 MB conda-forge
154 -----|-----
155 Total: 26.2 MB
156
157 The following NEW packages will be INSTALLED:
158
159 ffmpeg conda-forge/win-64::ffmpeg-4.3.1-ha925a31_0
160 Proceed ([y]/n)? y

```

```

160 Downloading and Extracting Packages:
161
162 Preparing transaction: done
163 Verifying transaction: done
164 Executing transaction: done
165
166 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda install numpy pandas
jupyter
167 Channels:
168 - defaults
169 Platform: win-64
170 Collecting package metadata (repodata.json): done
171 Solving environment: done
172
173 ==> WARNING: A newer version of conda exists. <==
174     current version: 25.3.1
175     latest version: 25.7.0
176
177 Please update conda by running
178
179     $ conda update -n base -c defaults conda
180
181 ## Package Plan ##
182
183 environment location: C:\Users\ANSEN\.conda\envs\multimedia
184
185 added / updated specs:
186 - jupyter
187 - numpy
188 - pandas
189
190
191 The following packages will be downloaded:
192
193 package | build
194 -----|-----
195 asttokens-3.0.0 | py311haa95532_0 49 KB
196 bottleneck-1.4.2 | py311h57dcf0c_0 146 KB
197 ca-certificates-2025.9.9 | haa95532_0 127 KB
198 ..... | .....
199 widgetsnbextension-4.0.13 | py311haa95532_0 949 KB
200 zeromq-4.3.5 | h6c54ac7_1 4.0 MB
201 -----|-----
202 Total: 212.6 MB
203
204 The following NEW packages will be INSTALLED:
205
206 asttokens pkgs/main/win-64::asttokens-3.0.0-py311haa95532_0
207 blas pkgs/main/win-64::blas-1.0-mkl
208 bottleneck pkgs/main/win-64::bottleneck-1.4.2-py311h57dcf0c_0
209 ... .....
210 widgetsnbextension pkgs/main/win-64::widgetsnbextension-4.0.13-py311haa95532_0
211 zeromq pkgs/main/win-64::zeromq-4.3.5-h6c54ac7_1
212
213 The following packages will be REMOVED:
214
215 libblas-3.9.0-35_h5709861_mkl
216 libcbblas-3.9.0-35_h2a3cdd5_mkl
217 liblapack-3.9.0-35_hf9ab0e9_mkl
218
219 The following packages will be UPDATED:
220

```

```

221 ca-certificates conda-forge/noarch::ca-certificates-2~ --> pkgs/main/win-64::ca-
certificates-2025.9.9-haa95532_0
222 mkl conda-forge::mkl-2024.2.2-h57928b3_16 --> pkgs/main::mkl-2025.0.0-
h5da7b33_930
223 tbb conda-forge::tbb-2021.13.0-h18a62a1_3 --> pkgs/main::tbb-2022.0.0-
h214f63a_0
224
225 The following packages will be SUPERSEDED by a higher-priority channel:
226
227 llvm-openmp conda-forge::llvm-openmp-20.1.8-hfa2b~ --> pkgs/main::llvm-openmp
-20.1.8-h29ce207_0
228 numpy conda-forge::numpy-2.3.3-py311h80b3fa~ --> pkgs/main::numpy-2.3.3-
py311hc2e1e29_0
229 scipy conda-forge::scipy-1.16.2-py311h9a1c3~ --> pkgs/main::scipy-1.16.1-
py311h86a6471_0
230
231 The following packages will be DOWNGRADED:
232
233 setuptools 78.1.1-py311haa95532_0 --> 72.1.0-py311haa95532_0
234
235 Proceed ([y]/n)?
236
237 Downloading and Extracting Packages:
238 mkl-2025.0.0 | 104.7 MB | #
239 #####
| 100%
240 scipy-1.16.1 | 28.1 MB | #
241 #####
| 100%
242 pandoc-2.12 | 14.6 MB | #
243 #####
| 100%
244 pandas-2.3.2 | 14.3 MB | #
245 #####
| 100%
246 notebook-7.4.5 | 11.0 MB | #
247 #####
| 100%
248 numpy-base-2.3.3 | 9.7 MB | #
249 #####
| 100%
250 zeromq-4.3.5 | 4.0 MB | #
251 #####
| 100%
252 debugpy-1.8.16 | 3.5 MB | #
253 #####
| 100%
254 pyqt-6.7.1 | 3.5 MB | #
255 #####
| 100%
256 setuptools-72.1.0 | 3.0 MB | #
257 #####
| 100%
258 pygments-2.19.1 | 2.2 MB | #
259 #####
| 100%
260 intel-openmp-2025.0. | 2.1 MB | #
261 #####
| 100%
262 ipython-9.1.0 | 1.2 MB | #
263 #####

```



```

252 | 100%
jedi-0.19.2 | 1.2 MB | #
#####
253 | 100%
tbb-devel-2022.0.0 | 1.1 MB | #
#####
254 | 100%
widgetsnbextension-4 | 949 KB | #
#####
255 | 100%
prompt-toolkit-3.0.4 | 746 KB | #
#####
256 | 100%
sip-6.10.0 | 718 KB | #
#####
257 | 100%
psutil-7.0.0 | 571 KB | #
#####
258 | 100%
259 Preparing transaction: done
260 Verifying transaction: done
261 Executing transaction: done
262

```

Kode 10: Output instalansi library

- Daftar library yang berhasil diinstall dengan versinya

3.4 Bagian 3: Verifikasi Instalasi

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

```

1 (multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda list | findstr "librosa soundfile
  scipy opencv pillow scikit-image matplotlib ffmpeg jupyter"
2 ffmpeg                4.3.1                ha925a31_0      conda-forge
3 imageio-ffmpeg        0.6.0                pypi_0         pypi
4 jupyter               1.1.1                py311haa95532_0
5 jupyter-lsp           2.3.0                pypi_0         pypi
6 jupyter_client        8.6.3                pyhd8ed1ab_1   conda-forge
7 jupyter_console      6.6.3                py311haa95532_0
8 jupyter_core          5.8.1                pyh5737063_0   conda-forge
9 jupyter_events        0.12.0               pyh29332c3_0   conda-forge
10 jupyter_server       2.17.0               pyhcf101f3_0   conda-forge
11 jupyter_server_terminals 0.5.3               pyhd8ed1ab_1   conda-forge
12 jupyterlab           4.4.7                pyhd8ed1ab_0   conda-forge
13 jupyterlab_pygments   0.3.0                pyhd8ed1ab_2   conda-forge
14 jupyterlab_server    2.27.3               pyhd8ed1ab_1   conda-forge
15 jupyterlab_widgets    3.0.15               py311haa95532_0
16 librosa              0.11.0               pypi_0         pypi
17 matplotlib           3.10.6               py311h1ea47a8_1 conda-forge
18 matplotlib-base       3.10.6               py311h43afe63_0
19 matplotlib-inline     0.1.7                pypi_0         pypi
20 opencv                4.10.0               py311h28596fa_7
21 opencv-python         4.12.0.88            pypi_0         pypi
22 pillow                11.3.0               py311hb328d1f_0
23 scikit-image          0.25.2               py311h11fd7f3_2 conda-forge
24 scipy                 1.16.1               py311h86a6471_0
25 soundfile             0.13.1               pypi_0         pypi

```

Kode 11: Verifikasi Versi Library yang Diinstal

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 12: 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 13: 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
[Deskripsi masalah dan solusi di sini]

4 Bagian Laporan

4.1 Output Verifikasi Instalasi

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

```

1 # Saya menjalankan script test audio dan image di atas file ipynb
2 # Output dari script test audio (sine_wave_test.png)
3
4 Generated 2s sine wave at 440Hz
5 Sample rate: 44100Hz
6 Total samples: 88200
7
8 # Output dari script test image (test_image.png)
9
10 Created test image: 400x300 pixels
11 Image shape: (300, 400, 3)
12 Image dtype: uint8

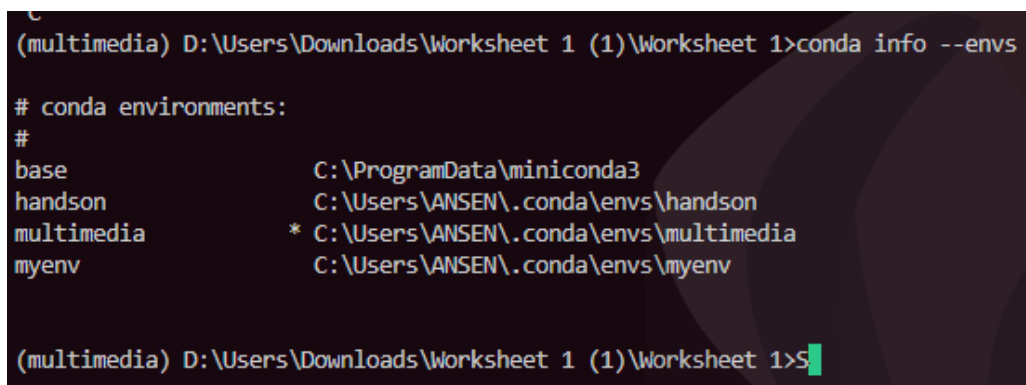
```

Kode 14: Output verifikasi instalasi

4.2 Screenshot Hasil Test

Sisipkan screenshot atau gambar hasil dari:

- Terminal/command prompt yang menunjukkan environment aktif



```

(multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>conda info --envs

# conda environments:
#
base                  C:\ProgramData\miniconda3
handson               C:\Users\ANSEN\.conda\envs\handson
multimedia            * C:\Users\ANSEN\.conda\envs\multimedia
myenv                 C:\Users\ANSEN\.conda\envs\myenv

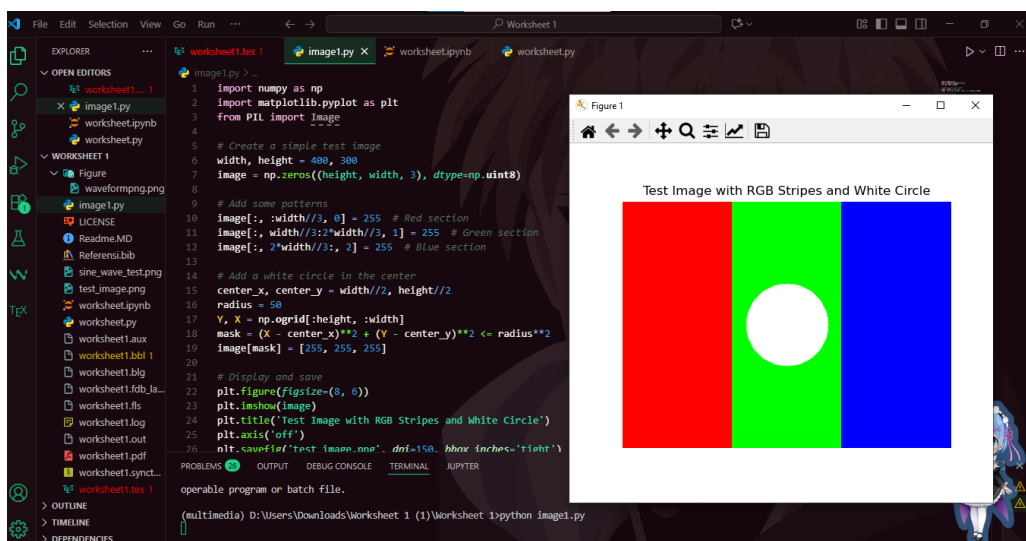
(multimedia) D:\Users\Downloads\Worksheet 1 (1)\Worksheet 1>S

```

- Output dari script test audio (sine wave plot)



- Output 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?

Penggunaan environment terpisah sangat penting karena project multimedia biasanya membutuhkan banyak library dengan dependensi yang kompleks, bahkan terkadang berbeda versi antara satu project dengan project lain. Dengan membuat environment khusus, kita dapat menghindari benturan versi library, menjaga stabilitas project, serta mempermudah replikasi di komputer lain tanpa merusak konfigurasi global Python.

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

conda merupakan package manager sekaligus environment manager yang mampu mengatur library Python maupun non-Python (misalnya *ffmpeg* atau *OpenCV* berbasis C++). **venv** hanya membuat virtual environment Python sederhana, tanpa manajemen dependensi tingkat lanjut. Sedangkan **uv** lebih modern dan berfokus pada performa instalasi yang lebih cepat, tetapi ekosistemnya masih relatif baru. Pada project ini saya memilih menggunakan **conda** karena lebih stabil dalam menangani dependensi multimedia yang kompleks.

3. Library mana yang paling sulit diinstall dan mengapa?

Library yang paling sulit diinstall adalah **librosa**, karena selain membutuhkan **numpy** dan **scipy**, library ini juga bergantung pada **soundfile** yang membutuhkan dukungan dari **libsndfile** (library non-Python). Hal ini sering menyebabkan error apabila instalasi hanya menggunakan **pip** tanpa dependensi sistem yang lengkap. Dengan **conda**, masalah ini lebih mudah teratasi karena paket biner sudah tersedia.

4. Bagaimana cara mengatasi masalah dependency conflict jika terjadi?

Cara yang umum dilakukan adalah dengan memperbarui atau menurunkan versi library sesuai rekomendasi dari error log. Selain itu, saya juga bisa mencoba membuat environment baru yang lebih bersih, kemudian menginstal library satu per satu dengan urutan yang benar. Jika library tidak tersedia di conda, saya melengkapinya dengan **pip install** di dalam environment tersebut. Strategi ini efektif mengurangi potensi konflik dependensi.

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

- **librosa**: Digunakan untuk analisis dan pemrosesan sinyal audio, seperti ekstraksi fitur (MFCC, mel-spectrogram).
- **soundfile**: Membaca dan menulis file audio dengan format **.wav**, **.flac**, dan lainnya.
- **scipy**: Library sains yang mendukung operasi matematika lanjutan, optimasi, dan pemrosesan sinyal.
- **opencv**: Pemrosesan citra dan video, termasuk deteksi objek, filtering, dan transformasi.
- **pillow**: Manipulasi gambar (membaca, menulis, editing gambar format umum).
- **scikit-image**: Analisis citra ilmiah seperti segmentasi, filtering, dan feature extraction.
- **matplotlib**: Visualisasi data dalam bentuk grafik, plot, maupun gambar.
- **ffmpeg**: Toolkit multimedia untuk konversi, decoding, dan encoding file audio-video.
- **jupyter**: Lingkungan interaktif berbasis notebook untuk eksperimen dan dokumentasi kode Python.

4.4 Troubleshooting

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

- **Masalah 1**: Saat menginstal library multimedia menggunakan **pip** di dalam environment conda yang sudah aktif, proses instalasi gagal karena beberapa dependensi non-Python (misalnya **libsndfile** untuk **soundfile**) tidak terpenuhi. Hal ini menyebabkan error dan library tidak dapat digunakan.

Solusi: Menggunakan **conda install -c conda-forge <nama_library>** untuk menginstal library tersebut. Dengan conda, dependensi non-Python juga akan terpasang otomatis sehingga instalasi berhasil tanpa error.

- **Masalah 2:** Saat menyusun laporan menggunakan LaTeX, proses kompilasi membutuhkan waktu lama dan muncul error **Package caption: Unused \captionsetup[table]**. Error ini muncul karena terdapat perintah konfigurasi caption untuk **table**, sementara di dokumen tidak ada environment tabel yang aktif.

Solusi: Menghapus atau menonaktifkan baris `\captionsetup[table]{...}` jika memang tidak menggunakan tabel. Alternatifnya, jika ingin tetap mendukung caption untuk tabel, pastikan ada lingkungan **table** yang sesuai di dokumen. Dengan cara ini error dapat diatasi dan proses kompilasi berjalan lebih cepat.

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 15: Export conda environment

5.2 Untuk venv/uv

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

Kode 16: Export pip requirements

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

```
1 name: base
2 channels:
3   - defaults
4   - https://repo.anaconda.com/pkgs/main
5   - https://repo.anaconda.com/pkgs/r
6   - https://repo.anaconda.com/pkgs/msys2
7 dependencies:
8   - anaconda-anon-usage=0.5.0=py312hfc23b7f_100
9   - anaconda_powershell_prompt=1.1.0=haa95532_0
10  - anaconda_prompt=1.1.0=haa95532_0
11  - annotated-types=0.6.0=py312haa95532_0
12  - anyio=4.6.2=py312haa95532_0
13  - archspec=0.2.3=pyhd3eb1b0_0
14  - argon2-cffi=21.3.0=pyhd3eb1b0_0
15  - argon2-cffi-bindings=21.2.0=py312h827c3e9_1
16  - asttokens=3.0.0=py312haa95532_0
17  - async-lru=2.0.4=py312haa95532_0
18  - attrs=24.3.0=py312haa95532_0
19  - babel=2.16.0=py312haa95532_0
20  - beautifulsoup4=4.12.3=py312haa95532_0
21  - blas=1.0=mkl
22  - bleach=6.2.0=py312haa95532_0
23  - boltons=24.1.0=py312haa95532_0
24  - brotli-python=1.0.9=py312h5da7b33_9
25  - bzip2=1.0.8=h2bbff1b_6
26  - ca-certificates=2025.2.25=haa95532_0
27  - certifi=2025.1.31=py312haa95532_0
28  - cffi=1.17.1=py312h827c3e9_1
29  - charset-normalizer=3.3.2=pyhd3eb1b0_0
30  - colorama=0.4.6=py312haa95532_0
31  - comm=0.2.1=py312haa95532_0
```

```

32 - conda=25.3.1=py312haa95532_0
33 - conda-anaconda-telemetry=0.1.2=py312haa95532_0
34 - conda-anaconda-tos=0.1.2=py312haa95532_0
35 - conda-content-trust=0.2.0=py312haa95532_1
36 - conda-libmamba-solver=25.1.1=pyhd3eb1b0_0
37 - conda-package-handling=2.4.0=py312haa95532_0
38 - conda-package-streaming=0.11.0=py312haa95532_0
39 - contourpy=1.3.1=py312h214f63a_0
40 - cpp-expected=1.1.0=h214f63a_0
41 - cryptography=43.0.3=py312hbd6ee87_1
42 - cycpler=0.11.0=pyhd3eb1b0_0
43 - debugpy=1.8.11=py312h5da7b33_0
44 - decorator=5.1.1=pyhd3eb1b0_0
45 - defusedxml=0.7.1=pyhd3eb1b0_0
46 - distro=1.9.0=py312haa95532_0
47 - executing=0.8.3=pyhd3eb1b0_0
48 - expat=2.6.4=h8ddb27b_0
49 - fmt=9.1.0=h6d14046_1
50 - fonttools=4.55.3=py312h827c3e9_0
51 - freetype=2.13.3=h0620614_0
52 - frozendict=2.4.2=py312haa95532_0
53 - h11=0.14.0=py312haa95532_0
54 - httpcore=1.0.2=py312haa95532_0
55 - httpx=0.27.0=py312haa95532_0
56 - icc_rt=2022.1.0=h6049295_2
57 - icu=73.1=h6c2663c_0
58 - idna=3.7=py312haa95532_0
59 - intel-openmp=2023.1.0=h59b6b97_46320
60 - ipykernel=6.29.5=py312haa95532_1
61 - ipython=8.30.0=py312haa95532_0
62 - jedi=0.19.2=py312haa95532_0
63 - jinja2=3.1.6=py312haa95532_0
64 - jpeg=9e=h827c3e9_3
65 - json5=0.9.25=py312haa95532_0
66 - jsonpatch=1.33=py312haa95532_1
67 - jsonpointer=2.1=pyhd3eb1b0_0
68 - jsonschema=4.23.0=py312haa95532_0
69 - jsonschema-specifications=2023.7.1=py312haa95532_0
70 - jupyter-lsp=2.2.0=py312haa95532_0
71 - jupyter_client=8.6.3=py312haa95532_0
72 - jupyter_core=5.7.2=py312haa95532_0
73 - jupyter_events=0.12.0=py312haa95532_0
74 - jupyter_server=2.15.0=py312haa95532_0
75 - jupyter_server_terminals=0.4.4=py312haa95532_1
76 - jupyterlab=4.3.4=py312haa95532_0
77 - jupyterlab_pygments=0.3.0=py312haa95532_0
78 - jupyterlab_server=2.27.3=py312haa95532_0
79 - kiwisolver=1.4.8=py312h5da7b33_0
80 - krb5=1.20.1=h5b6d351_0
81 - lcms2=2.16=hb4a4139_0
82 - lerc=4.0.0=h5da7b33_0
83 - libarchive=3.7.7=h9243413_0
84 - libcurl=8.11.1=haff574d_0
85 - libdeflate=1.22=h5bf469e_0
86 - libffi=3.4.4=hd77b12b_1
87 - libiconv=1.16=h2bbff1b_3
88 - libmamba=2.0.5=hcd6fe79_1
89 - libmambapy=2.0.5=py312h214f63a_1
90 - libpng=1.6.39=h8cc25b3_0
91 - libpq=17.4=h70ee33d_0
92 - libsodium=1.0.18=h62dcd97_0
93 - libsolov=0.7.30=hf2fb9eb_1

```

```

94 - libssh2=1.11.1=h2adbb87_0
95 - libtiff=4.5.1=h44ae7cf_1
96 - libwebp-base=1.3.2=h3d04722_1
97 - libxml2=2.13.5=h24da03e_0
98 - lz4-c=1.9.4=h2bbff1b_1
99 - markdown-it-py=2.2.0=py312haa95532_1
100 - markupsafe=3.0.2=py312h827c3e9_0
101 - matplotlib=3.10.0=py312haa95532_0
102 - matplotlib-base=3.10.0=py312he19b0ae_0
103 - matplotlib-inline=0.1.6=py312haa95532_0
104 - mdurl=0.1.0=py312haa95532_0
105 - menuinst=2.2.0=py312h5da7b33_1
106 - mistune=3.1.2=py312haa95532_0
107 - mkl=2023.1.0=h6b88ed4_46358
108 - mkl-service=2.4.0=py312h827c3e9_2
109 - mkl_fft=1.3.11=py312h827c3e9_0
110 - mkl_random=1.2.8=py312h0158946_0
111 - nbclient=0.10.2=py312haa95532_0
112 - nbconvert-core=7.16.6=py312haa95532_0
113 - nbformat=5.10.4=py312haa95532_0
114 - nest-asyncio=1.6.0=py312haa95532_0
115 - nlohmann_json=3.11.2=h6c2663c_0
116 - notebook-shim=0.2.4=py312haa95532_0
117 - numpy=2.0.1=py312hfd52020_1
118 - numpy-base=2.0.1=py312h4dde369_1
119 - openjpeg=2.5.2=hae555c5_0
120 - openssl=3.0.16=h3f729d1_0
121 - overrides=7.4.0=py312haa95532_0
122 - packaging=24.2=py312haa95532_0
123 - pandocfilters=1.5.0=pyhd3eb1b0_0
124 - parso=0.8.4=py312haa95532_0
125 - pcre2=10.42=h0ff8eda_1
126 - pillow=11.1.0=py312h096bffc_0
127 - pip=25.0=py312haa95532_0
128 - platformdirs=3.10.0=py312haa95532_0
129 - pluggy=1.5.0=py312haa95532_0
130 - prometheus-client=0.21.1=py312haa95532_0
131 - prompt-toolkit=3.0.43=py312haa95532_0
132 - prompt_toolkit=3.0.43=hd3eb1b0_0
133 - psutil=5.9.0=py312h827c3e9_1
134 - pure_eval=0.2.2=pyhd3eb1b0_0
135 - pybind11-abi=5=hd3eb1b0_0
136 - pycosat=0.6.6=py312h827c3e9_2
137 - pycparser=2.21=pyhd3eb1b0_0
138 - pydantic=2.10.3=py312haa95532_0
139 - pydantic-core=2.27.1=py312h636fa0f_0
140 - pygments=2.15.1=py312haa95532_1
141 - pyparsing=3.2.0=py312haa95532_0
142 - pyqt=6.7.1=py312h5da7b33_0
143 - pyqt6-sip=13.9.1=py312h827c3e9_0
144 - pysocks=1.7.1=py312haa95532_0
145 - python=3.12.9=h14ffc60_0
146 - python-dateutil=2.9.0post0=py312haa95532_2
147 - python-fastjsonschema=2.20.0=py312haa95532_0
148 - python-json-logger=3.2.1=py312haa95532_0
149 - pywin32=308=py312h5da7b33_0
150 - pywinpty=2.0.15=py312h72d21ff_0
151 - pyyaml=6.0.2=py312h827c3e9_0
152 - pyzmq=26.2.0=py312h5da7b33_0
153 - qtbase=6.7.3=h0804d20_0
154 - qtdeclarative=6.7.3=h5da7b33_0
155 - qtsvg=6.7.3=hf2fb9eb_0

```



```

156 - qttools=6.7.3=h0de5f00_0
157 - qtwebchannel=6.7.3=h5da7b33_0
158 - qtwebsockets=6.7.3=h5da7b33_0
159 - referencing=0.30.2=py312haa95532_0
160 - reproc=14.2.4=hd77b12b_2
161 - reproc-cpp=14.2.4=hd77b12b_2
162 - requests=2.32.3=py312haa95532_1
163 - rfc3339-validator=0.1.4=py312haa95532_0
164 - rfc3986-validator=0.1.1=py312haa95532_0
165 - rich=13.9.4=py312haa95532_0
166 - rpds-py=0.22.3=py312h636fa0f_0
167 - ruamel.yaml=0.18.6=py312h827c3e9_0
168 - ruamel.yaml.clib=0.2.8=py312h827c3e9_0
169 - scipy=1.15.2=py312h9d85e7c_1
170 - send2trash=1.8.2=py312haa95532_1
171 - setuptools=75.8.0=py312haa95532_0
172 - simdjson=3.10.1=h214f63a_0
173 - sip=6.10.0=py312h5da7b33_0
174 - six=1.17.0=py312haa95532_0
175 - sniffio=1.3.0=py312haa95532_0
176 - soupsieve=2.5=py312haa95532_0
177 - spdlog=1.11.0=h59b6b97_0
178 - sqlite=3.45.3=h2bbff1b_0
179 - stack_data=0.2.0=pyhd3eb1b0_0
180 - tbb=2021.8.0=h59b6b97_0
181 - terminado=0.17.1=py312haa95532_0
182 - tinycss2=1.4.0=py312haa95532_0
183 - tk=8.6.14=h0416ee5_0
184 - tornado=6.4.2=py312h827c3e9_0
185 - tqdm=4.67.1=py312hfc267ef_0
186 - traitlets=5.14.3=py312haa95532_0
187 - truststore=0.10.0=py312haa95532_0
188 - typing-extensions=4.12.2=py312haa95532_0
189 - typing_extensions=4.12.2=py312haa95532_0
190 - tzdata=2025a=h04d1e81_0
191 - unicodedata2=15.1.0=py312h827c3e9_1
192 - urllib3=2.3.0=py312haa95532_0
193 - vc=14.42=haa95532_4
194 - vs2015_runtime=14.42.34433=he0abc0d_4
195 - wcwidth=0.2.5=pyhd3eb1b0_0
196 - webencodings=0.5.1=py312haa95532_2
197 - websocket-client=1.8.0=py312haa95532_0
198 - wheel=0.45.1=py312haa95532_0
199 - win_inet_pton=1.1.0=py312haa95532_0
200 - winpty=0.4.3=4
201 - xz=5.4.6=h8cc25b3_1
202 - yaml=0.2.5=he774522_0
203 - yaml-cpp=0.8.0=hd77b12b_1
204 - zeromq=4.3.5=hd77b12b_0
205 - zlib=1.2.13=h8cc25b3_1
206 - zstandard=0.23.0=py312h4fc1ca9_1
207 - zstd=1.5.6=h8880b57_0
208 prefix: C:\ProgramData\miniconda3

```

Kode 17: Requirements file For Environment Reproduction

6 Kesimpulan

Tuliskan kesimpulan Anda mengenai:

- Pengalaman setup Python environment untuk multimedia

- Persiapan untuk project multimedia selanjutnya
- Saran untuk mahasiswa lain yang akan melakukan setup serupa

Kesan saya dalam membuat environment sebenarnya cukup mudah, karena sebelumnya saya sudah pernah mengambil mata kuliah DSP (Digital Signal Processing) yang juga menggunakan environment Python untuk mempermudah akses pekerjaan. Namun, pada mata kuliah ini saya kembali berlatih melakukan instalasi dan setup environment menggunakan **conda**, khususnya **miniconda**, agar lebih memahami manajemen environment serta menggunakan versi Python yang lebih rendah (3.8) demi stabilitas dan keamanan kompatibilitas library.

Kesulitan yang cukup terasa justru ada pada proses pembungkusan dokumentasi dengan file $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. Hal ini membutuhkan waktu ekstra untuk mengatur kode dan tata letak agar laporan lebih rapi, berbeda dengan pengalaman sebelumnya yang menggunakan notebook (**ipynb**) pada mata kuliah DSP. Meskipun demikian, proses instalasi environment dari tutorial yang diberikan cukup mudah, terlebih dengan tambahan `\usepackage{lmodern}` untuk mendukung tampilan font yang lebih baik.

Sebagai persiapan untuk project multimedia selanjutnya, saya merasa lebih siap bekerja dengan dataset audio maupun citra yang lebih kompleks, serta mengintegrasikan hasil pemrosesan tersebut ke dalam pipeline penelitian maupun aplikasi nyata. Saya juga akan lebih berhati-hati dalam memilih tools untuk manajemen environment agar tidak terhambat oleh masalah instalasi di kemudian hari.

Untuk mahasiswa lain yang akan melakukan setup serupa, disarankan menggunakan **conda** sebagai environment manager utama karena lebih andal dalam mengatasi dependensi multimedia. Selain itu, dokumentasikan setiap langkah dengan baik menggunakan $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ agar laporan lebih terstruktur, konsisten, dan mudah dipahami.

7 Referensi

Sertakan referensi yang Anda gunakan selama proses setup dan troubleshooting.

- [Chat GPT](#)
- [Conda Documentation](#)
- [Miniconda](#)

References