

**PENGENALAN CITRA DIGITAL
TRANSFORMASI DAN PERBAIKAN CITRA PADA DOMAIN
FREKUENSI**



Dosen Pengampu : Dwi Shina Angreni, S.Si., M.Kom

Oleh : Ardiyansyah R

NIM : F551 20 001

**UNIVERSITAS TADULAKO
FAKULTAS TEKNIK - JURUSAN TEKNOLOGI INFORMASI
S1 TEKNIK INFOMATIKA**

2021/2022

I. TUJUAN

- A. Mahasiswa mampu memahami perbaikan citra pada domain frekuensi.
- B. Mahasiswa mampu membuat program perbaikan citra pada domain frekuensi.

II. ALAT DAN BAHAN

- A. Laptop
- B. Pycharm
- C. Python

III. TEORI DASAR

A. Transformasi Fourier

Transformasi Fourier diawali di abad ke-19 (tepatnya tahun 1822) oleh seorang matematikawan Perancis yang bernama Jean Baptiste Fourier. Dalam penelitiannya, Fourier berhasil menunjukkan bahwa semua fungsi yang bersifat periodik (sinyal) dalam waktu dapat diekspresikan sebagai fungsi penjumlahan (integral) trigonometri sinus dan cosinus dari berbagai frekuensi. Tidak peduli bagaimana rumitnya bentuk dari sebuah sinyal, selama sinyal tersebut periodik dan memenuhi beberapa kondisi matematika, maka sinyal tersebut akan dapat direpresentasikan dalam penjumlahan fungsi *sinus* dan *cosinus*. Dalam proses pengolahan citra, transformasi fourier dapat digunakan untuk perbaikan citra (*image restoration*) atau peningkatan kualitas citra (*image enhancement*).

B. Low-Pass Filtering

Low-pass filtering adalah proses filter yang melewatkan komponen citra dengan nilai intensitas yang rendah dan meredam komponen citra dengan nilai intensitas yang tinggi. Low pass filter akan menyebabkan citra menjadi lebih halus dan lebih blur. Aturan kernel untuk low-pass filter adalah:

1. Semua koefisien kernel harus positif

2. Jumlah semua koefisien kernel harus sama dengan 1

C. High-Pass Filtering

Berkebalikan dengan *low-pass filtering*, *high-pass filtering* adalah proses filter yang melewatkan komponen citra dengan nilai intensitas yang tinggi dan meredam komponen citra dengan nilai intensitas yang rendah. *High pass* filter akan menyebabkan tepi objek tampak lebih tajam dibandingkan sekitarnya. Aturan kernel untuk high-pass filter adalah:

1. Koefisien kernel boleh positif, negative, atau nol.
2. Jumlah semua koefisien kernel adalah 0 atau 1.

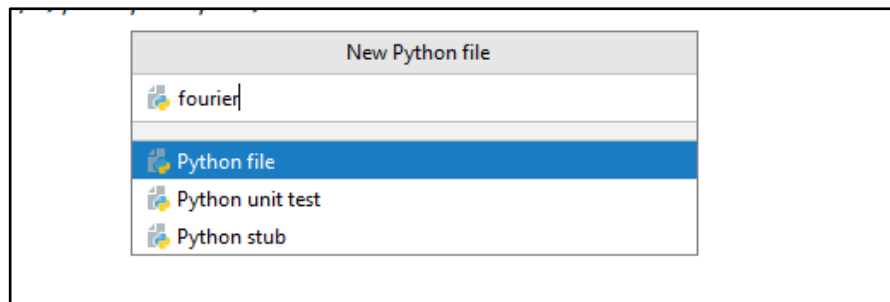
IV. LANGKAH KERJA

A. Menginstal package yang diperlukan :

1. Numpy
2. Opencv-python
3. matplotlib

B. Transformasi Fourier

1. Membuat file fourier.py

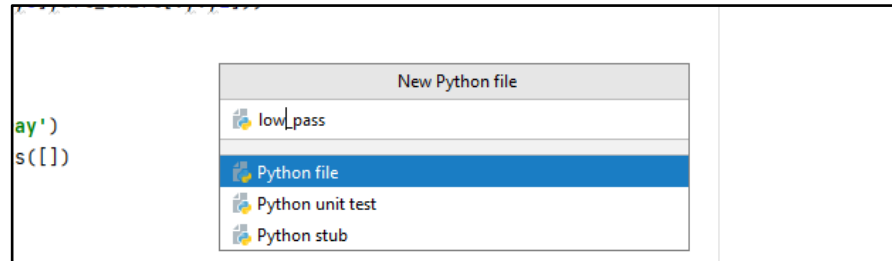


2. Memasukkan kode program

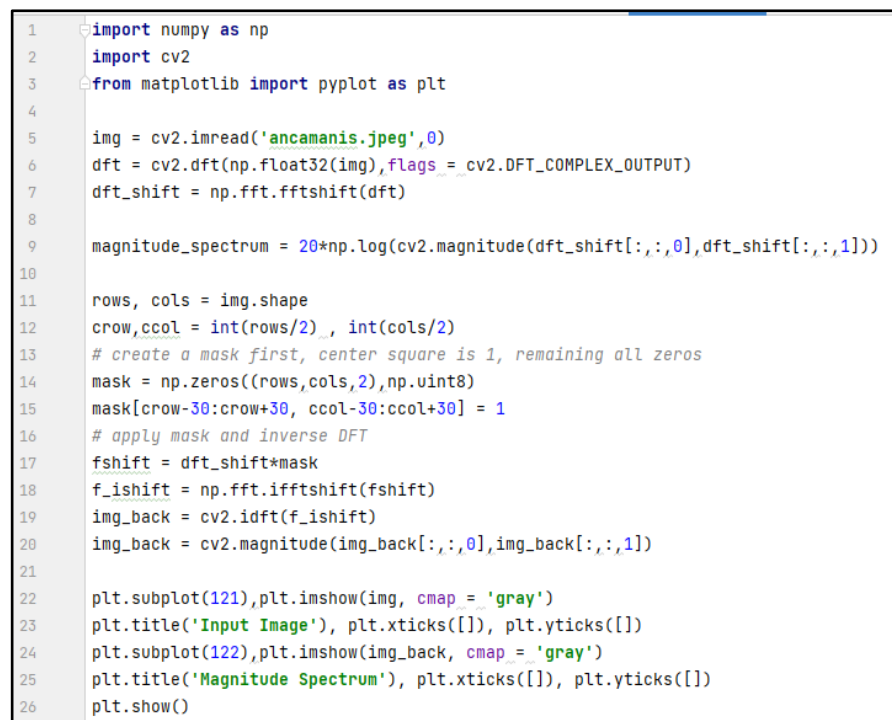
```
1 import numpy as np
2 import cv2
3 from matplotlib import pyplot as plt
4
5 img = cv2.imread('ancamanis.jpeg',0)
6
7 dft = cv2.dft(np.float32(img), flags = cv2.DFT_COMPLEX_OUTPUT)
8 dft_shift = np.fft.fftshift(dft)
9
10 magnitude_spectrum = 20*np.log(cv2.magnitude(dft_shift[:, :, 0], dft_shift[:, :, 1]))
11
12 plt.subplot(121), plt.imshow(img, cmap = 'gray')
13 plt.title('Input Image'), plt.xticks([], plt.yticks([]))
14 plt.subplot(122), plt.imshow(magnitude_spectrum, cmap = 'gray')
15 plt.title('Magnitude Spectrum'), plt.xticks([], plt.yticks([]))
16 plt.show()
```

C. Low-Pass Filter

1. Membuat file low_pass.py

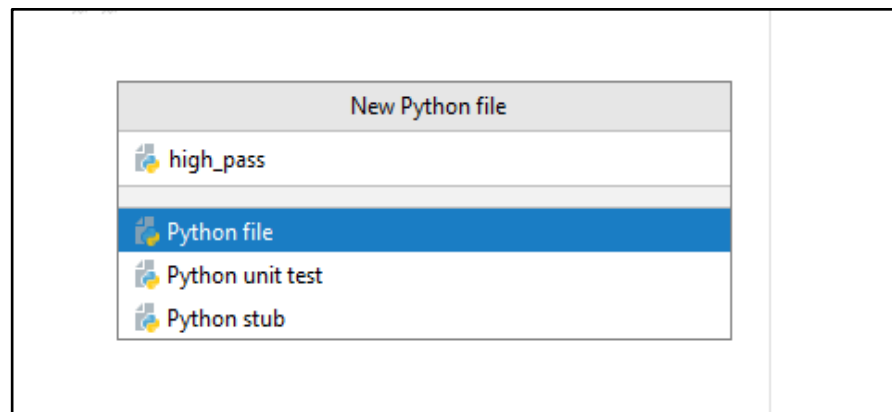


2. Memasukkan kode program



D. High-Pass Filter

1. Membuat file high_pass.py



2. Memasukkan kode program

```
import cv2
import numpy as np
from matplotlib import pyplot as plt

img = cv2.imread('ancamanis.jpeg',0)
f = np.fft.fft2(img)
fshift = np.fft.fftshift(f)
magnitude_spectrum = 20*np.log(np.abs(fshift))

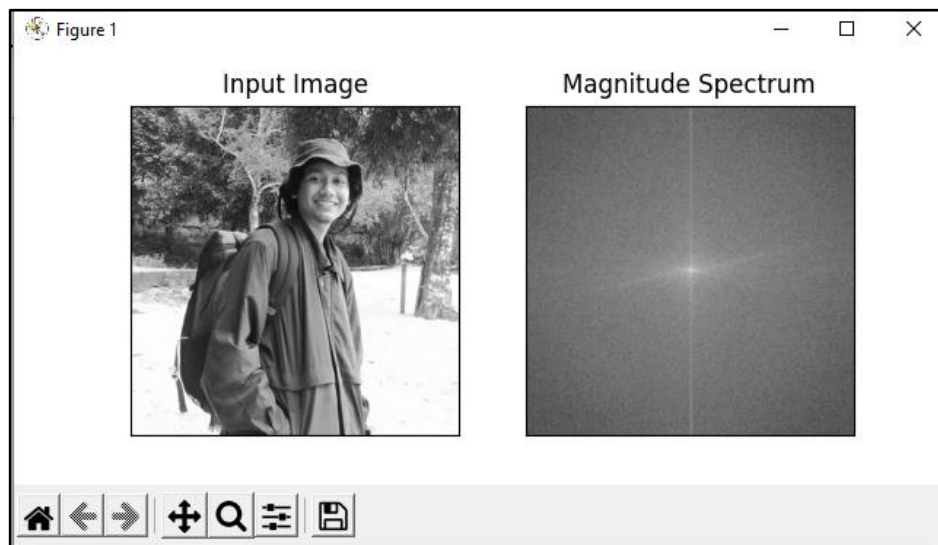
rows, cols = img.shape
crow,ccol = int(rows/2) ,int(cols/2)
fshift[crow-30:crow+30, ccol-30:ccol+30] = 0
f_ishift = np.fft.ifftshift(fshift)
img_back = np.fft.ifft2(f_ishift)
img_back = np.abs(img_back)

plt.subplot(131),plt.imshow(img, cmap = 'gray')
plt.title('Input Image'), plt.xticks([], plt.yticks([]))
plt.subplot(132),plt.imshow(img_back, cmap = 'gray')
plt.title('Image after HPF'), plt.xticks([], plt.yticks([]))
plt.subplot(133),plt.imshow(img_back)
plt.title('Result in JET'), plt.xticks([], plt.yticks([]))

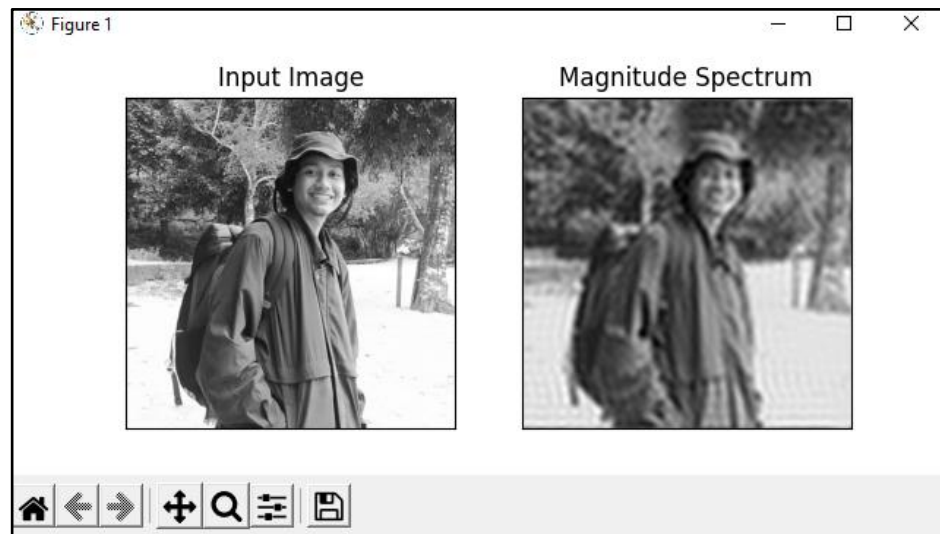
plt.show()
```

V. HASIL PERCOBAAN

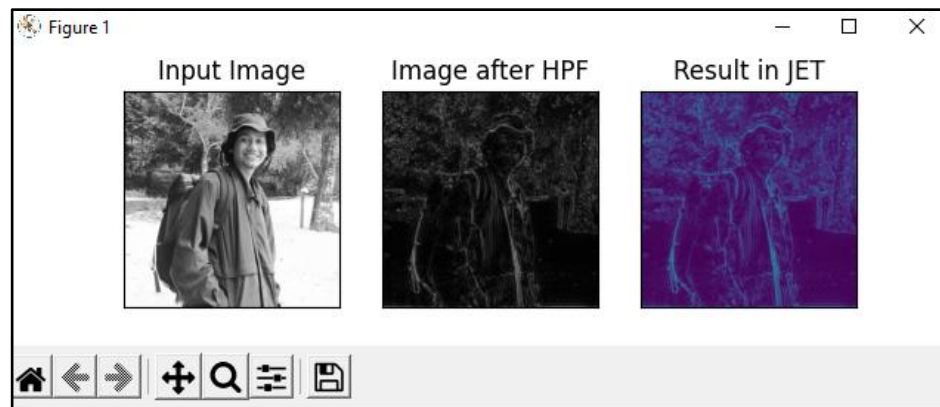
A. Transformasi Fourier



B. Low-Pass Filter



C. High-Pass Filter



VI. ANALISIS

Package yang digunakan adalah numpy untuk proses komputasi numerik, package opencv untuk mengelolah citra, dan matplotlib digunakan untuk visualisasi data yang biasanya berbentuk grafik. Fungsi `cv2.imread()` digunakan untuk menambahkan file gambar. Fungsi `cv2.dft()` digunakan untuk mencari domain frekuensi. Untuk menampilkan gambar dalam satu frame menggunakan `subplot()`, dan untuk menampilkan gambarnya memakai `plt.show()`. Untuk menjelaskan atau memberi judul pada gambar yang akan ditampilkan menggunakan `plt.title('isi judul gambar')`.