

E. TRANSFORMASI FOURIER

Judul	PRAKTEK E1- DISCRETE FOURIER TRANSFORM
Deskripsi	<p>Proses pengolahan citra digital dapat dilakukan tidak hanya adri ranah spasial namun juga dari ranah frekuensi. Salah satu metode yang banyak dikenal dan digunakan dalam pengolahan citra ranah frekuensi adalah Discrete Fourier Transform (DFT). Proses DFT memiliki persamaan sebagai berikut ini</p> $FT : F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp[-2j\pi(ux/M + vy/N)]$ $InversFT : f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp[2j\pi(ux/M + vy/N)]$ <p>M = tinggi citra (jumlah baris) N = lebar citra (jumlah kolom)</p> <p>DFT dapat digunakan untuk memperbaiki kualitas citra dengan memanfaatkan Low Pass Filter (LPF) bertujuan mereduksi noise.</p>
Estimasi waktu	30 menit
Prerequisite	<ol style="list-style-type: none"> 1. siapkan noisy image dan citra normal dalam kualitas yang baik 2. Sudah membuat menu file Transformasi dengan sub menu DFT Smoothing Image
Listing Program	<pre> import cv2 import numpy as np from matplotlib import pyplot as plt x=np.arange(256) y=np.sin(2*np.pi*x/3) y+=max(y) img=np.array([[y[j]*127 for j in range (256)]for i in range(256)],dtype=np.uint8) plt.imshow(img) img=cv2.imread('noisy_image.png',0) dft=cv2.dft(np.float32(img),flags=cv2.DFT_COMPLEX_OUTPUT) dft_shift=np.fft.fftshift(dft) magnitude_spectrum=20*np.log((cv2.magnitude(dft_shift[:, :, 0],dft_shift[:, :, 1]))) rows, cols = img.shape crow, ccol = int(rows / 2), int(cols / 2) mask = np.zeros((rows, cols, 2), np.uint8) r = 50 center = [crow, ccol] x, y = np.ogrid[:rows, :cols] mask_area = (x - center[0]) ** 2 + (y - center[1]) ** 2 <= r*r </pre>

```

mask[mask_area] = 1

fshift=dft_shift*mask
fshift_mask_mag=20*np.log(cv2.magnitude(fshift[:, :, 0], fshift[:, :, 1]))
f_ishift=np.fft.ifftshift(fshift)

img_back=cv2.idft(f_ishift)
img_back=cv2.magnitude(img_back[:, :, 0], img_back[:, :, 1])

fig=plt.figure(figsize=(12,12))
ax1=fig.add_subplot(2,2,1)
ax1.imshow(img, cmap='gray')
ax1.title.set_text('Input Image')
ax2=fig.add_subplot(2,2,2)
ax2.imshow(magnitude_spectrum, cmap='gray')
ax2.title.set_text('FFT of Image')
ax3=fig.add_subplot(2,2,3)
ax3.imshow(fshift_mask_mag, cmap='gray')
ax3.title.set_text('FFT + Mask')
ax4=fig.add_subplot(2,2,4)
ax4.imshow(img_back, cmap='gray')
ax4.title.set_text('Inverse fourier')
plt.show()

```

Tugas

1. Implementasikan coding diatas dalam satu program dengan membuat sub menu DFT Smoothing Image
2. Analisis spektrum dan image yang dihasilkan jika dilakukan pengujian nilai pada variabel r. Amati, laporkan dan berikan penjelasan hasil pengamatannya

Judul

PRAKTEK E2- DISCRETE FOURIER TRANSFORM

Deskripsi

DFT selain dapat digunakan untuk memperbaiki kualitas citra dengan memanfaatkan Low Pass Filter (LPF) bertujuan mereduksi noise. DFT yang menggunakan HPF bertujuan untuk mendeteksi tepi.

Estimasi waktu

30 menit

Prerequisite

1. siapkan noisy image dan citra normal dalam kualitas yang baik
2. Sudah membuat menu file Transformasi dengan sub menu DFT Smoothing Image

Listing Program

```

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

x=np.arange(256)
y=np.sin(2*np.pi*x/3)

```

```

y+=max(y)

img=np.array([[y[j]*127 for j in range(256)] for i in
range(256)],dtype=np.uint8)

plt.imshow(img)
img=cv2.imread('lena_gray.jpg',0)

dft=cv2.dft(np.float32(img),flags=cv2.DFT_COMPLEX_OUTPUT)
dft_shift=np.fft.fftshift(dft)

magnitude_spectrum=20*np.log((cv2.magnitude(dft_shift[:, :, 0],dft_shif
t[:, :, 1])))

rows,cols=img.shape
crow,ccol=int(rows/2),int(cols/2)

mask=np.ones((rows,cols,2),np.uint8)
r=80
center=[crow,ccol]
x,y=np.ogrid[:rows,:cols]
mask_area=(x-center[0])**2+(y-center[1])**2 <=r*r
mask[mask_area]=0

fshift=dft_shift*mask
fshift_mask_mag=20*np.log(cv2.magnitude(fshift[:, :, 0],fshift[:, :, 1]))
f_ishift=np.fft.ifftshift(fshift)

img_back=cv2.idft(f_ishift)
img_back=cv2.magnitude(img_back[:, :, 0],img_back[:, :, 1])

fig=plt.figure(figsize=(12,12))
ax1=fig.add_subplot(2,2,1)
ax1.imshow(img, cmap='gray')
ax1.title.set_text('Input Image')
ax2=fig.add_subplot(2,2,2)
ax2.imshow(magnitude_spectrum, cmap='gray')
ax2.title.set_text('FFT of Image')
ax3=fig.add_subplot(2,2,3)
ax3.imshow(fshift_mask_mag, cmap='gray')
ax3.title.set_text('FFT + Mask')
ax4=fig.add_subplot(2,2,4)
ax4.imshow(img_back, cmap='gray')
ax4.title.set_text('Inverse fourier')
plt.show()

```

Tugas

1. Implementasikan coding diatas dalam satu program dengan membuat sub menu DFT Smoothing Image
 2. Analisis spektrum dan image yang dihasilkan jika dilakukan pengujian nilai pada variabel r. Amati, laporkan dan berikan penjelasan hasil pengamatannya
-