

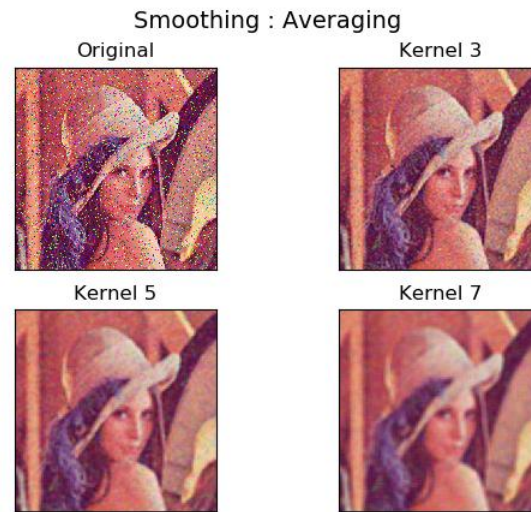
### Tugas 3 - Smoothing

#### IF5153 - Pemrosesan Pengelolaan Data Multimedia

Nama : Mohamad Hanifan

NIM : 23518026

#### A. Smoothing Result



**Gambar 3.** Smoothing menggunakan Averaging Technique

Pada Teknik *Averaging*, setiap piksel dirata-ratakan dengan piksel-piksel di sekitarnya. Atau, dengan kata lain adalah dengan mengonvolusikan gambar terhadap kernel pada **Gambar 2**. Kernel pada **Gambar 2** tersebut adalah contoh kernel berdiameter 5.

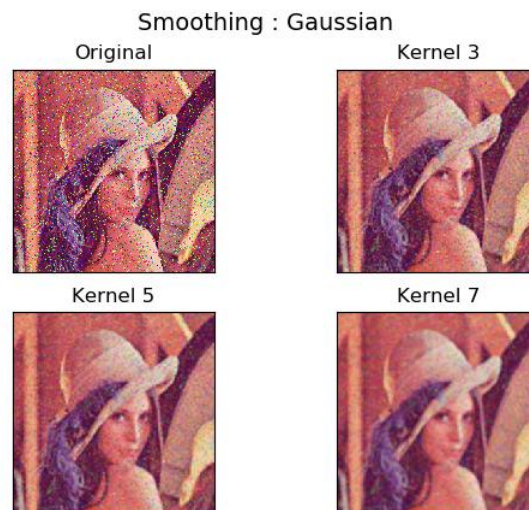
$$K = \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

**Gambar 2.** Kernel pada Averaging Technique



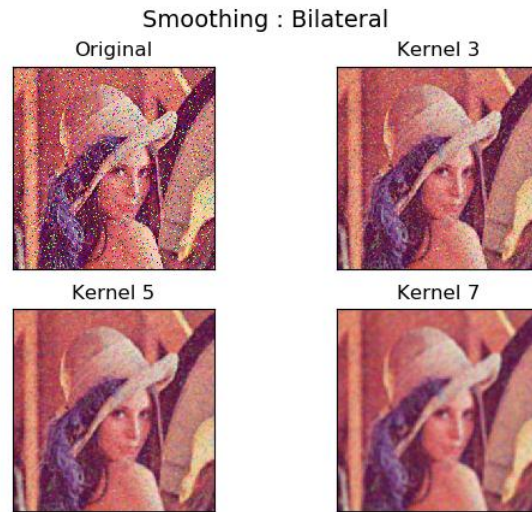
**Gambar 3.** Smoothing menggunakan Median Technique

Pada Teknik Median, nilai piksel yang digunakan adalah nilai median dari kumpulan piksel di sekitarnya. Teknik Median untuk Kernel berukuran 7x7 tidak dilakukan.



**Gambar 4.** Smoothing menggunakan Gaussian Technique

Pada teknik ini, setiap piksel menggunakan nilai hasil distribusi gauss dari piksel-piksel disekitarnya.



**Gambar 5.** Smoothing menggunakan Bilateral Technique

Teknik Bilateral ini juga menggunakan Filter Gaussian, dengan ditambahkan sebuah filter lagi untuk mendeteksi pinggiran. Sehingga gambar hasil dari teknik ini tetap memiliki kualitas yang baik di pinggiran (*edge*). Pada laporan ini, dipilih nilai **sigmaColor** dan **sigmaSpace** 75.

### C. Questions And Jawaban

- What is kernel?

Kernel adalah jarak pandang dari sebuah piksel. Kernel adalah matriks yang digunakan sebagai pengali pada proses konvolusi. **Gambar 1** adalah contoh kernel untuk mengambil rata-rata dari matriks lain yang berukuran 5x5.

- Which smoothing technique produce the best result? Why?

Teknik yang menghasilkan gambar paling baik adalah Teknik Bilateral. Karena pada teknik ini, *edges* (pinggiran/daerah perubahan warna) tidak menjadi blur, tidak seperti teknik-teknik lainnya.

Kekurangan teknik ini adalah proses komputasinya relatif lambat.

- What is correlation between kernel and smoothing result?

Semakin besar ukuran kernel, gambar akan semakin blur karena diproses dengan banyak sekali piksel, padahal piksel tersebut hanya berpengaruh dengan piksel-piksel di sekitarnya saja. Selain itu, untuk ukuran kernel yang besar, proses komputasi semakin banyak.

Pada ukuran kernel yang kecil, bisa jadi noise tidak hilang karena semakin sedikit piksel yang diproses.

### Referensi

1. [https://docs.opencv.org/4.0.1/d4/d13/tutorial\\_py\\_filtering.html](https://docs.opencv.org/4.0.1/d4/d13/tutorial_py_filtering.html)
2. [https://docs.opencv.org/4.0.1/d4/d86/group\\_\\_imgproc\\_\\_filter.html](https://docs.opencv.org/4.0.1/d4/d86/group__imgproc__filter.html)

## D. Kode Program

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

img = plt.imread('lena_noise.png')
noimg = plt.imread('no_image.png')

kernel3 = np.ones((3,3),np.float32)/9
kernel5 = np.ones((5,5),np.float32)/25
kernel7 = np.ones((7,7),np.float32)/49

avg3 = cv2.filter2D(img,-1,kernel3)
avg5 = cv2.filter2D(img,-1,kernel5)
avg7 = cv2.filter2D(img,-1,kernel7)

gauss3 = cv2.GaussianBlur(img, (3,3), 0)
gauss5 = cv2.GaussianBlur(img, (5,5), 0)
gauss7 = cv2.GaussianBlur(img, (7,7), 0)

med3 = cv2.medianBlur(img, 3)
med5 = cv2.medianBlur(img, 5)
med7 = noimg

sigma = 75
bil3 = cv2.bilateralFilter(img,3,sigma,sigma)
bil5 = cv2.bilateralFilter(img,5,sigma,sigma)
bil7 = cv2.bilateralFilter(img,7,sigma,sigma)

def mergeFigure(fIndex, methodName, img1, img3, img5, img7):
    plt.figure(fIndex)
    plt.subplot(221), plt.imshow(img1), plt.title('Original')
    plt.xticks([], plt.yticks([]))
    plt.subplot(222), plt.imshow(img3), plt.title('Kernel 3')
    plt.xticks([], plt.yticks([]))
    plt.subplot(223), plt.imshow(img5), plt.title('Kernel 5')
    plt.xticks([], plt.yticks([]))
    plt.subplot(224), plt.imshow(img7), plt.title('Kernel 7')
    plt.xticks([], plt.yticks([]))
    plt.suptitle('Smoothing : ' + methodName, size=14)
    plt.savefig(methodName)

mergeFigure(1, 'Averaging', img, avg3, avg5, avg7)
mergeFigure(2, 'Gaussian', img, gauss3, gauss5, gauss7)
mergeFigure(3, 'Median', img, med3, med5, med7)
mergeFigure(4, 'Bilateral', img, bil3, bil5, bil7)

plt.show()
```