

Computer vision – HW6

A 、 Source code

```
# WeiWen Wu 2023-10-18
from numpy import float32,ones
import cv2 as cv
from matplotlib import pyplot as plt

class plot_from_matplotlib:
    """
    Make pictures from matplotlib.

    # Example
    ...
    plot = plot_from_matplotlib("title")
    plot("Original",img)
    plot_from_matplotlib.show()
    ...
    """

    n:int=1

    def __init__(self,name:str) -> None:
        fig = plt.figure()
        fig.canvas.manager.window.setWindowTitle(name)

    def __call__(self,name:str,dst,cmap:str='gray',save:bool=1):      # Make pictures.
        n = self.n
        path = __file__.split("\\")[-2]
        plt.subplot(2,3,n), plt.imshow(dst,cmap), plt.title(name)
        plt.xticks([]), plt.yticks([]) # Do not show scale.
        if save:
            cv.imwrite(f"./{path}/{name}.png",dst,[int(cv.IMWRITE_PNG_COMPRESSION),0])
        self.n+=1

    @staticmethod
    def show():
        plt.show()

    def remove_noise(img_file:str):
        plot = plot_from_matplotlib(img_file.split(".")[0])
        img = cv.imread(img_file)
        ### Plot original image. ###
```

```

plot("Original",img)

### Filter2D ####
# Define the kernel, using a N×N NumPy array, where N is odd.
kernel = ones((3, 3), float32)/9
dst = cv.filter2D(img, -1, kernel)
plot("Filter2D",dst)

### Blur ####
img.blur = cv.blur(img, (3, 3))
plot("Blur",img.blur)

### GaussianBlur ####
img_GaussianBlur = cv.GaussianBlur(img, (3, 3),0)
plot("GaussianBlur",img_GaussianBlur)

### Median Filtering ####
img_medianBlur = cv.medianBlur(img, 3)
plot("Median Filtering",img_medianBlur)

### Bilateral Filtering ####
img_bilateralFilter = cv.bilateralFilter(img,15, 75, 75)
plot("Bilateral Filtering",img_bilateralFilter)

# remove_noise('lena_noise2.jpg')
remove_noise('lena_noise3.jpg')

### Show matplotlib. ####
plot_from_matplotlib.show()

```

B、Result map

從下圖來看，使用中值濾波器(Median filter)效果較好。

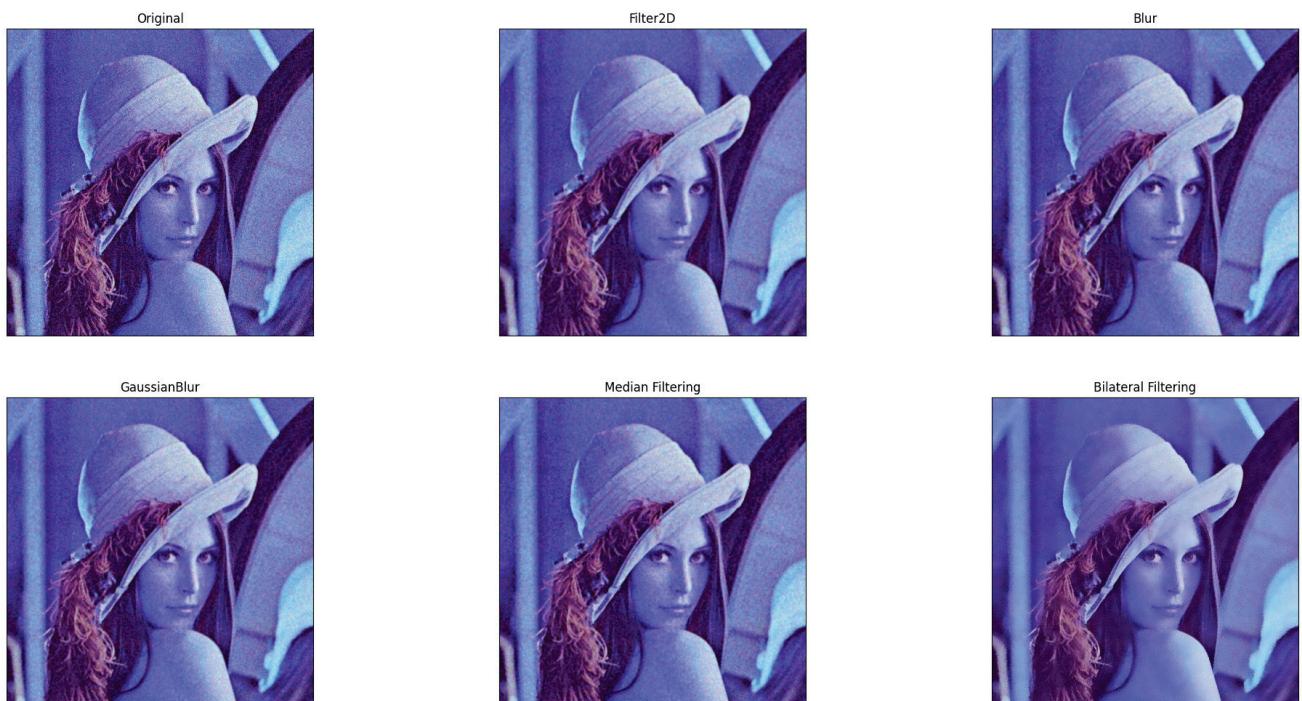


Figure 1 lena_noise2



Figure 2 lena_noise3

C、Appendix (original picture)

i. lena_noise2



Figure 3 Bilateral Filtering



Figure 4 Blur



Figure 5 Filter2D



Figure 6 GaussianBlur



Figure 7 Median Filtering



Figure 8 Original

ii. **lena_noise3**



Figure 9 Bilateral Filtering



Figure 10 Blur



Figure 11 Filter2D

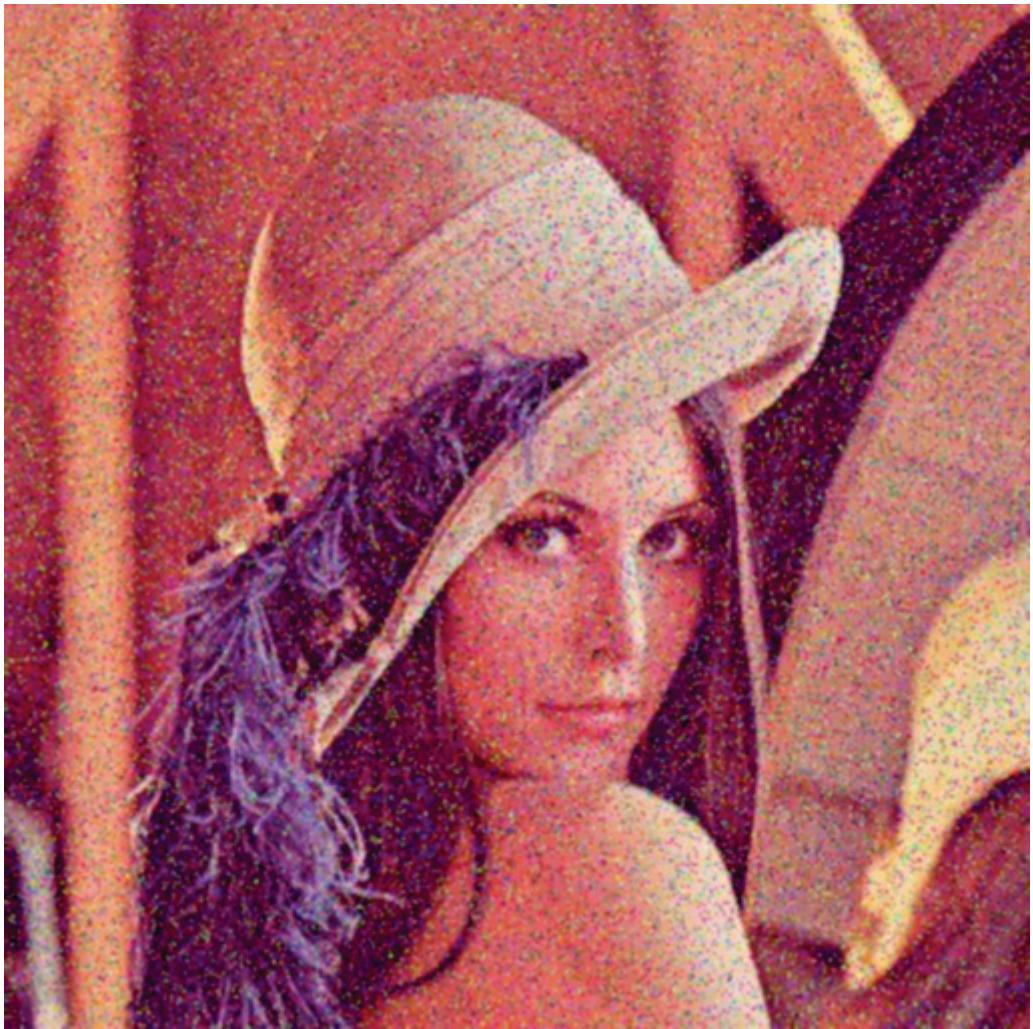


Figure 12 GaussianBlur



Figure 13 Median Filtering



Figure 14 Original