

นายสิทธิพงศ์ จำรัสกุฑธิรังค์ 6410301019
วันที่เริ่มทำ 2023-07-05 y-m-d

รายงานวิชา image processing งาน classwork ครั้งที่ 3 หัวข้อเรื่อง denoised

code

```
import cv2 as cv
import numpy as np
import random as rd

part = "./22962.jpg"

img = cv.imread(part, cv.IMREAD_GRAYSCALE)
img = np.array(img, dtype="uint8")

density_salt = 0.2
density_pepper = 0.2

number_of_white_pexel = int(density_salt * img.shape[0] * img.shape[1])
number_of_black_pexel = int(density_pepper * img.shape[0] * img.shape[1])
img_noise=np.array(img, dtype="uint8")

#เติม noise ขาว
for i in range(number_of_white_pexel):
    y=rd.randint(0,img.shape[0]-1)
    x=rd.randint(0,img.shape[1]-1)
    img_noise[y][x]=255

#เติม noise ดำ
for i in range(number_of_black_pexel):
    y=rd.randint(0,img.shape[0]-1)
    x=rd.randint(0,img.shape[1]-1)
```

```
img_noise[y][x]=0

# ขั้น denoise
denoised_img = cv.medianBlur(img_noise, 1)

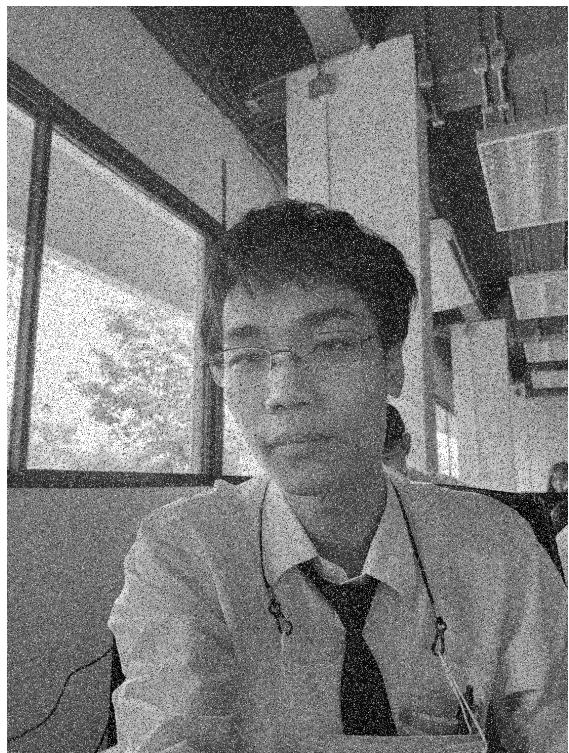
cv.imwrite("./denoised.png", denoised_img)

#noise ที่ไม่ถูกลบหาได้จาก out-in แบบ อ.แจ็ค สอน
out_vs_in = denoised_img-img
cv.imwrite("./noised.png", img_noise)
cv.imwrite("./original.png", img)
cv.imwrite("./out_vs_in.png", out_vs_in)
```

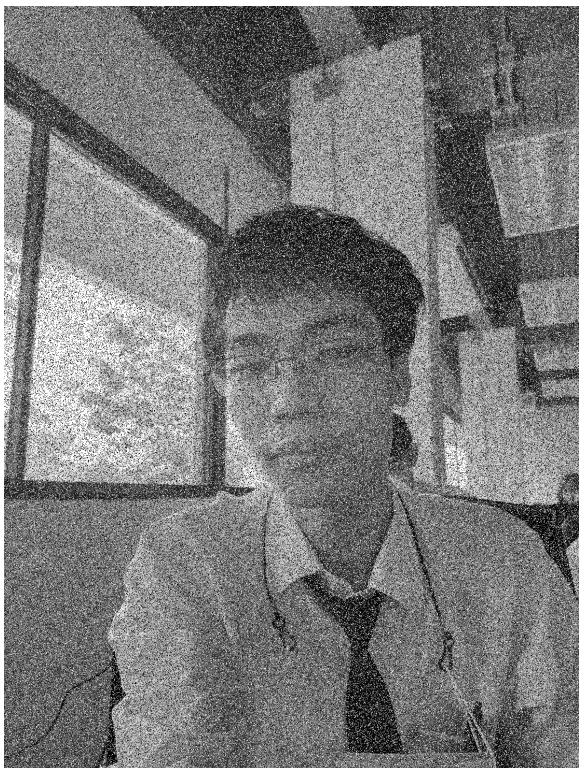
จาก code ดังกล่าว

ผมรับภาพหน้าผม弄มาใส่ noise สีขาวสุด(salt) สีดำสุด(pepper) แบบสุ่มๆ





หลังใส่ noise density 0.1



noise 0.2

median blur	ภาพที่ได้	output - input
1		

3



5



7



25



สรุป

วิธีการวัดของผม ผมจะใช้หลัก out - in เบบ อ.แจ็ค เพื่อหาความแตกต่าง
ระหว่าง ภาพซ่อม กับต้นฉบับ

ถ้าต้องการซ่อมแซมภาพให้กลับมาใกล้เคียงกับแบบเดิมที่สุดต้องใส่ median blur เป็น 1

ถ้าต้องการลบ noise ออกทั้งหมด โดยที่ภาพใกล้เคียงกับของเดิมมากที่สุด ต้อง^{ใช้ 7}