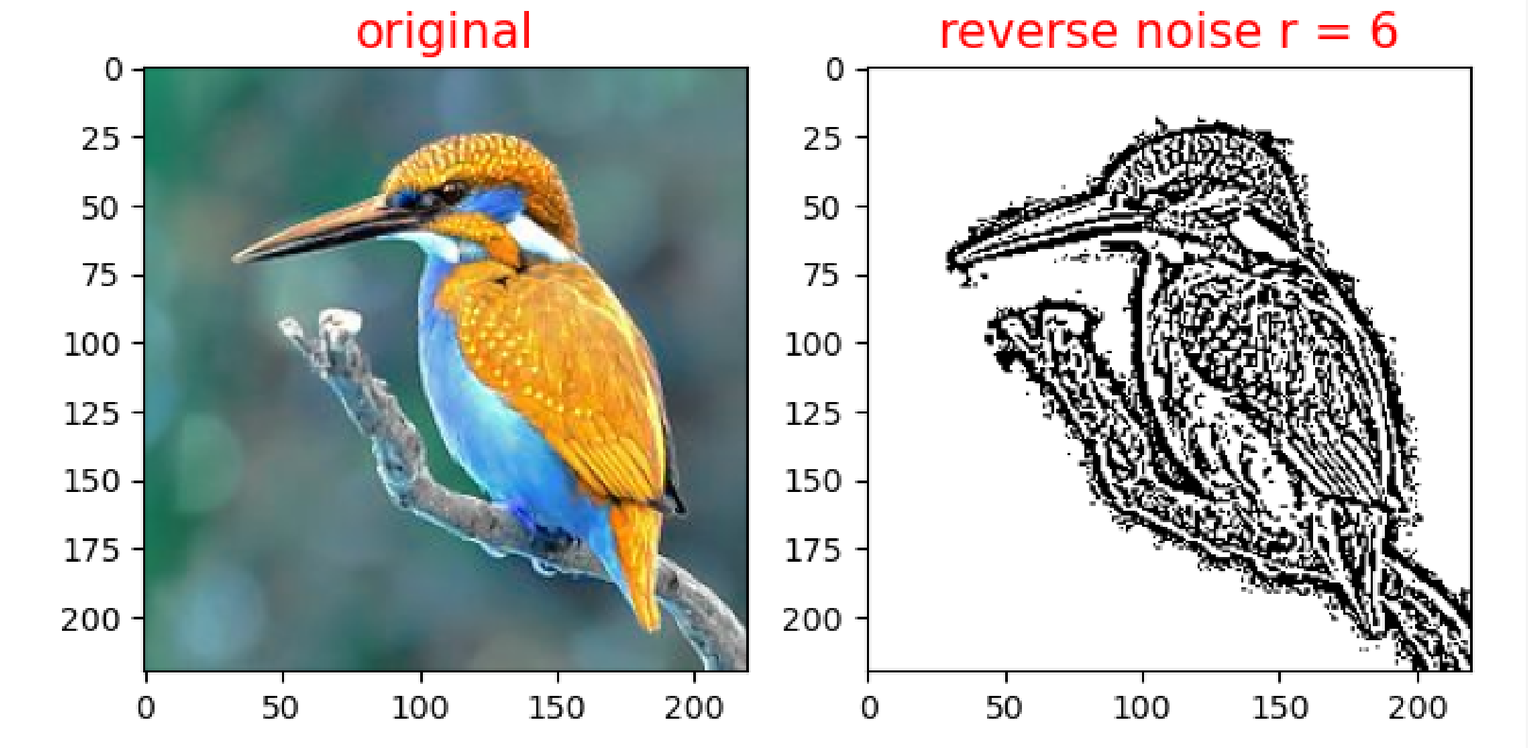
# 多媒體技術概論　期末報告


資工4A　409416541　周騏軍
資工4A　409411369　黃柏熏
資工4A　409416137　趙碩政
資傳4A　409040879　徐慧婷

# 影像轉水墨畫


original
reverse noise r = 6
0
25
50
75
100
125
150
175
200
0
50
100
150
200

# 應用函式與邏輯

1

## cv2.COLOR_BGR2GRAY

將讀取進來的彩色影像轉為灰階影像

2

## cv2.GaussianBlur

使用高斯濾波去除背景雜訊並生成新影像

3

## cv2.Canny

取得影像的邊緣資訊

4

## change(img)

將canny的邊緣改成黑色

5

## cv2.adaptiveThreshold

將影像透過高斯法的適應性閥值化來將灰階影像的灰階值變為只有 0 和 255

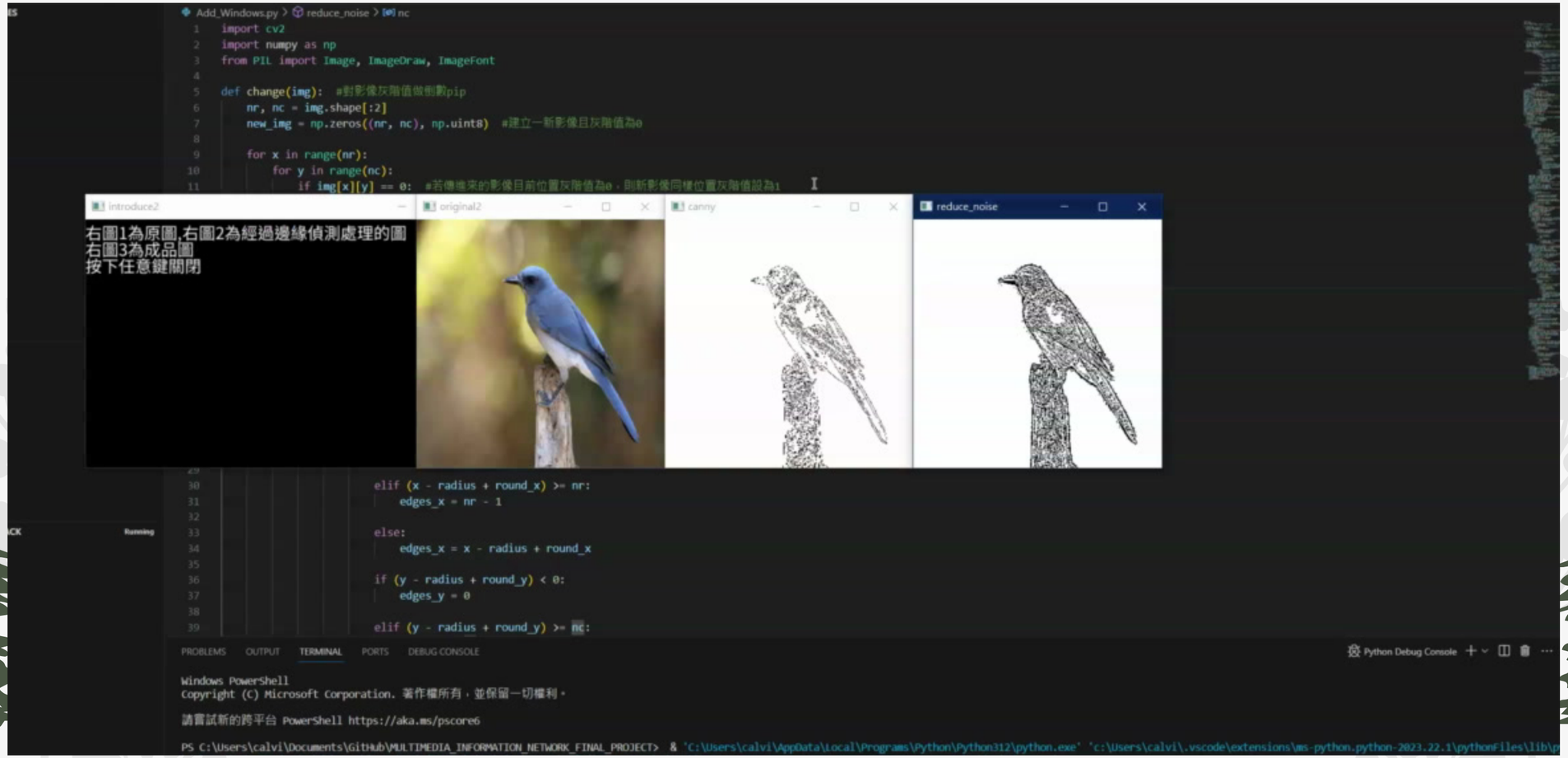
6

## reduce_noise(img, edges, radius)

在各像素點的半徑(radius)周圍的邊緣資訊(edges)是否有值，有值則保留，沒值則將灰階值設為255(白色)，以此來確認哪些像素點的灰階值應該被保留下來，其餘的灰階值為 255，以此來生成成品

# 修改部分

## 視窗化


Add_Windows.py > reduce_noise > nc
import cv2
import numpy as np
from PIL import Image, ImageDraw, ImageFont
def change(img):  #對影像灰階值做倒數pip
nr, nc = img.shape[:2]
new_img = np.zeros((nr, nc), np.uint8)  #建立一新影像且灰階值為0
for x in range(nr):
for y in range(nc):
if img[x][y] == 0:  #若傳進來的影像目前位置灰階值為0，則新影像同樣位置灰階值設為1
introduce2
右圖1為原圖,右圖2為經過邊緣偵測處理的圖
右圖3為成品圖
按下任意鍵關閉
original2
canny
reduce_noise
elif (x - radius + round_x) >= nr:
edges_x = nr - 1
else:
edges_x = x - radius + round_x
if (y - radius + round_y) < 0:
edges_y = 0
elif (y - radius + round_y) >= nc:
PROBLEMS
OUTPUT
TERMINAL
PORTS
DEBUG CONSOLE
Python Debug Console
Windows PowerShell
Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。
請嘗試新的跨平台 PowerShell https://aka.ms/pscore6
PS C:\Users\calvi\Documents\GitHub\MULTIMEDIA_INFORMATION_NETWORK_FINAL_PROJECT> & 'C:\Users\calvi\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\calvi\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\p

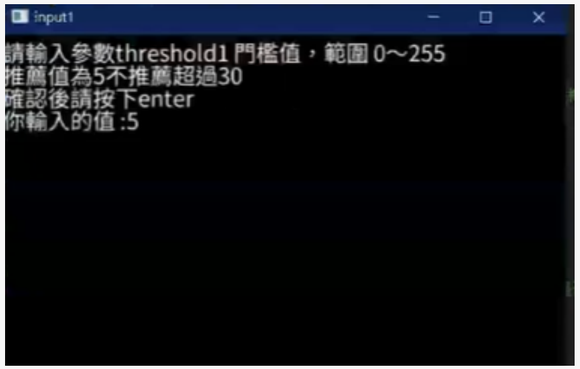
# 程式碼

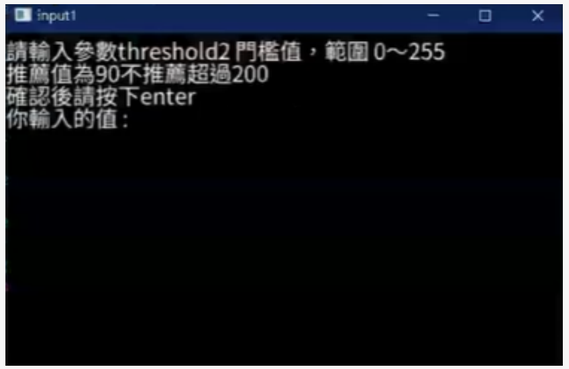
```
windowname = ['start', 'input1']
recommendation = [1, 5, 30, 2, 90, 200, 6, 10]
img = cv2.imread("images.jpg")  #讀取影像

gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)  #將讀取的影像轉換為灰階影像
gaussian_img = cv2.GaussianBlur(gray_img, (5, 5), 0)  #將讀取的影像透過高斯濾波將影像平滑化，以此來減少背景躁點

menu = 1  #控制畫面
new_img1 = np.zeros((300, 500), np.uint8)  #製作全黑圖片
fontFace = cv2.FONT_HERSHEY_TRIPLEX  #設定字形
fontpath = 'NotoSansTC-Regular.ttf'  #字形路徑

imgPil_wrong = Image.fromarray(np.zeros((300, 500), np.uint8))   #輸入錯誤時顯示的畫面
draw_wrong = ImageDraw.Draw(imgPil_wrong)
draw_wrong.text((0, 0),"輸入錯誤請重試", fill=(255), font=ImageFont.truetype(fontpath, 20))
wrong_picture = np.array(imgPil_wrong)

imgPil1 = Image.fromarray(new_img1)   #初始畫面
draw1 = ImageDraw.Draw(imgPil1)
word=['此程式可以把圖片轉成山水畫', '以下為控制邊緣偵測的數值', '使用的是Canny來進行邊緣偵測']
cv2.namedWindow('start',0)
cv2.moveWindow('start',500,300)

for i in range(3):
    draw1.text((0, i*20),word[i], fill=(255), font=ImageFont.truetype(fontpath, 20) )
    wordcontrol = np.array(imgPil1)
    cv2.imshow('start',wordcontrol)
    # 按下任意鍵則關閉所有視窗
    cv2.waitKey(2000)
    cv2.destroyAllWindows()
```

# 程式碼

```
while True:
    if menu==1:
        k = 0
        while(k < 2):
            cv2.namedWindow('input1',0)
            cv2.moveWindow('input1',500,300)

            control=0
            imgPil2 = Image.fromarray(np.zeros((300, 500), np.uint8))
            draw2 = ImageDraw.Draw(imgPil2)
            word1=['請輸入參數threshold'+str(recommendation[3*k])+\
                   ' 門檻值，範圍 0～255','推薦值為'+str(recommendation[3*k+1])+\
                   '不推薦超過'+str(recommendation[3*k+2]),'確認後請按下enter','你輸入的值 : ']

            for j in range (2):  #一次輸出兩段文字
                draw2.text((0, j*40), word1[j*2], fill=(255), font=ImageFont.truetype(fontpath, 20) )
                draw2.text((0, (2*j+1)*20), word1[2*j+1], fill=(255), font=ImageFont.truetype(fontpath, 20) )
                wordcontrol = np.array(imgPil2)

                if j == 0:
                    cv2.imshow('input1',wordcontrol)
                    # 自動關閉所有視窗
                    cv2.waitKey(2000)
                    cv2.destroyAllWindows()

                elif j == 1:
                    location = 0  #控制數字位置
                    number = 0  #紀錄數字

                    while(1):
                        cv2.imshow('input1',wordcontrol)
                        # 按下enter鍵則關閉所有視窗
                        value = cv2.waitKey(0)
                        cv2.destroyAllWindows()
                        if value == 13: #enter鍵
                            break

                        elif 57 >= value >= 48: #數字鍵
                            draw2.text((110+location*10, 60),str(value-48), fill=(255), font=ImageFont.truetype(fontpath, 20) )
                            wordcontrol = np.array(imgPil2)
                            number = number*10+value-48
                            location += 1
```

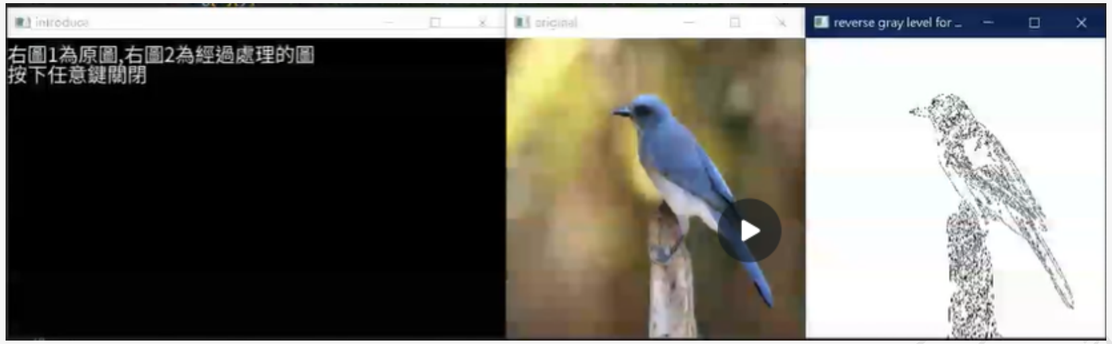
```
            if k == 0:
                if(0 > number or number > 255):
                    cv2.imshow('reset', wrong_picture)
                    cv2.waitKey(2000)
                    cv2.destroyAllWindows()
                    continue

                threshold1 = number
                k += 1

            elif k == 1:
                if(0 > number or number > 255):
                    cv2.imshow('reset', wrong_picture)
                    cv2.waitKey(2000)
                    cv2.destroyAllWindows()
                    continue

                threshold2 = number
                k += 1
```



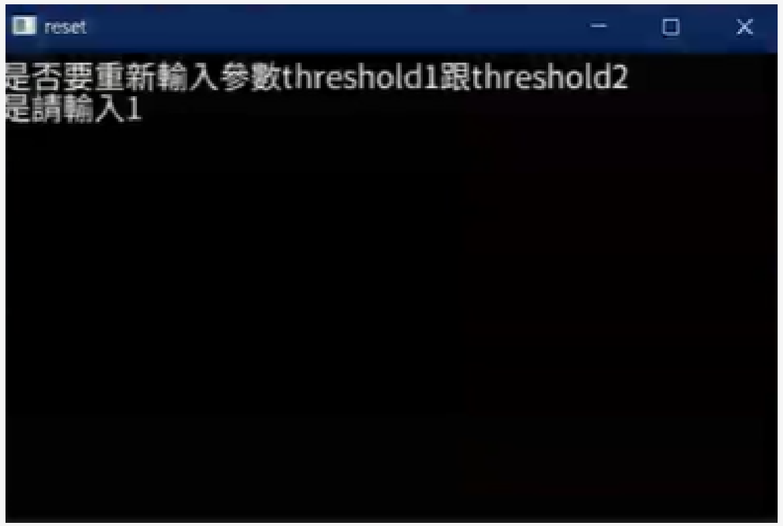
# 程式碼

```
edges_img = cv2.Canny(gaussian_img, threshold1, threshold2)#影像邊緣偵測
edges_img = change(edges_img)  #因為Canny所生成之邊緣影像的邊緣為白色背景為黑色，故將其黑白顛倒
imgPil3 = Image.fromarray(np.zeros((300, 500), np.uint8))
draw3 = ImageDraw.Draw(imgPil3)
word2=['右圖1為原圖,右圖2為經過處理的圖', '按下任意鍵關閉']

for i in range(2):
    draw3.text((0, i*20), word2[i], fill=(255), font=ImageFont.truetype(fontpath, 20))

wordcontrol2 = np.array(imgPil3)
imgs=[wordcontrol2, img, edges_img]
titles=['introduce','original', 'reverse gray level for edges']
# 顯示圖片
for i in range (3):
    cv2.namedWindow(titles[i], 0)

    if i == 0:
        cv2.resizeWindow(titles[i], 500, 300)
        cv2.moveWindow(titles[i], 300, 300)
    else:
        cv2.resizeWindow(titles[i], 300, 300)
        cv2.moveWindow(titles[i], 300*i+500, 300)
    # 按下任意鍵則關閉所有視窗
    cv2.imshow(titles[i], imgs[i])

cv2.waitKey(0)
cv2.destroyAllWindows()
```



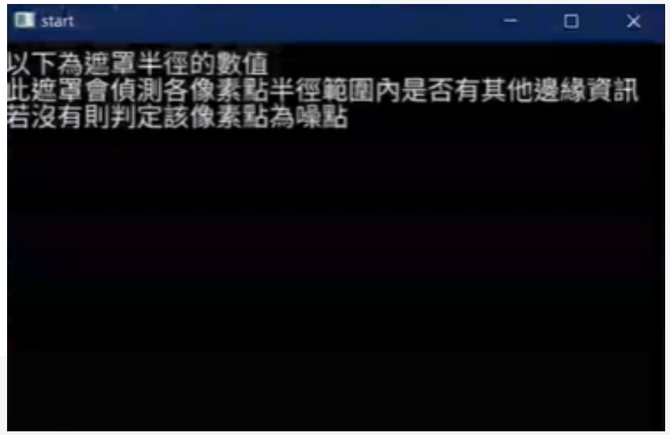
# 程式碼

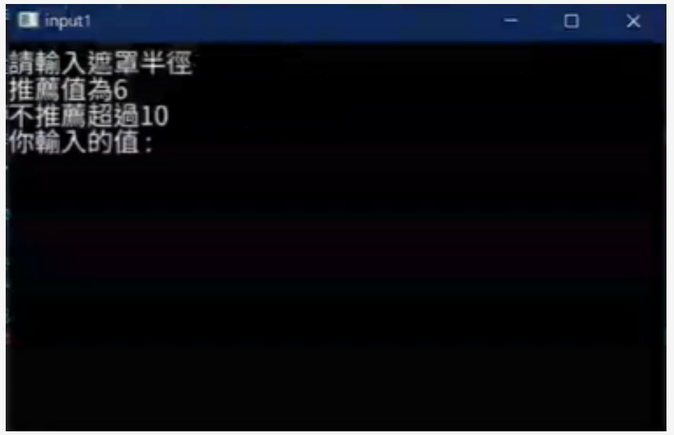
```
word3 = ['是否要重新輸入參數threshold1跟threshold2', '是請輸入1', '否請輸入2', '你輸入的值 :  ']
imgPil4 = Image.fromarray(np.zeros((300, 500), np.uint8))
draw4 = ImageDraw.Draw(imgPil4)
cv2.namedWindow('reset', 0)
cv2.resizeWindow('reset', 500, 300)
cv2.moveWindow('reset', 500, 300)

for i in range(2):
    draw4.text((0, i*40), word3[i*2], fill=(255), font=ImageFont.truetype(fontpath, 20))
    draw4.text((0, (2*i+1)*20), word3[2*i+1], fill=(255), font=ImageFont.truetype(fontpath, 20))
    wordcontrol1 = np.array(imgPil4)

    if i == 0:
        cv2.imshow('reset', wordcontrol1)
        cv2.waitKey(2000)
        cv2.destroyAllWindows()

    elif i == 1:
        while(1):
            cv2.imshow('reset',wordcontrol1)
            control1 = cv2.waitKey(0)-48
            cv2.destroyAllWindows()

            if 1 <= control1 <= 2 :
                menu = control1
                break

            else:
                cv2.imshow('reset', wrong_picture)
                cv2.waitKey(2000)
                cv2.destroyAllWindows()
```



# 程式碼

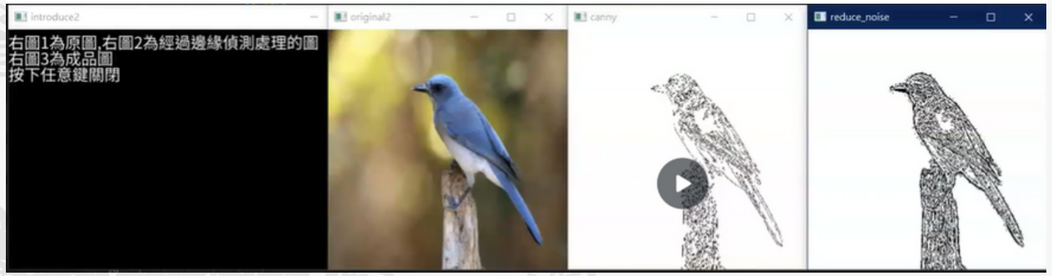
```
elif menu == 2:
    word4=['以下為遮罩半徑的數值', '此遮罩會偵測各像素點半徑範圍內是否有其他邊緣資訊', '若沒有則判定該像素點為噪點']
    adaptive_theshold_gaussian = cv2.adaptiveThreshold(gray_img, 255, cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY, 11, 2)#將影像透過高斯法的邊應性閾值化來將灰階影像變為只有黑白兩色
    imgPil_start2 = Image.fromarray(np.zeros((300, 500), np.uint8))
    draw_start2 = ImageDraw.Draw(imgPil_start2)

    for i in range(3):
        draw_start2.text((0, i*20),word4[i], fill=(255), font=ImageFont.truetype(fontpath, 20) )
        start2_control = np.array(imgPil_start2)
        cv2.imshow('start',start2_control)
        # 按下任意鍵則關閉所有視窗
        cv2.waitKey(2500)
        cv2.destroyAllWindows()

    word5 = ['請輸入遮罩半徑', '推薦值為6', '不推薦超過10', '你輸入的值 : ']
    imgPil_input2 = Image.fromarray(np.zeros((300, 500), np.uint8))
    draw_input2 = ImageDraw.Draw(imgPil_input2)

    for j in range (2):#一次輸出兩段文字
        draw_input2.text((0, j*40), word5[j*2], fill=(255), font=ImageFont.truetype(fontpath, 20) )
        draw_input2.text((0, (2*j+1)*20), word5[2*j+1], fill=(255), font=ImageFont.truetype(fontpath, 20) )
        wordcontrol2 = np.array(imgPil_input2)

        if j == 0:
                cv2.imshow('input1',wordcontrol2)
                # 自動關閉所有視窗
                cv2.waitKey(2000)
                cv2.destroyAllWindows()

        elif j==1:
                location=0 #控制數字位置
                number=0 #紀錄數字

                while(1):
                    cv2.imshow('input1',wordcontrol2)
                    # 按下enter鍵則關閉所有視窗
                    value = cv2.waitKey(0)
                    cv2.destroyAllWindows()

                    if value==13: #enter鍵
                        break
                    elif 57>=value>=48: #數字鍵
                        draw_input2.text((110+location*10, 60),str(value-48), fill=(255), font=ImageFont.truetype(fontpath, 20) )
                        wordcontrol2 = np.array(imgPil_input2)
                        number = number*10+value-48
                        location+=1

            radius = number

reduce_noise_img = reduce_noise(adaptive_theshold_gaussian, edges_img, radius)  #透過邊緣資訊來將不必要的地方刪除
```



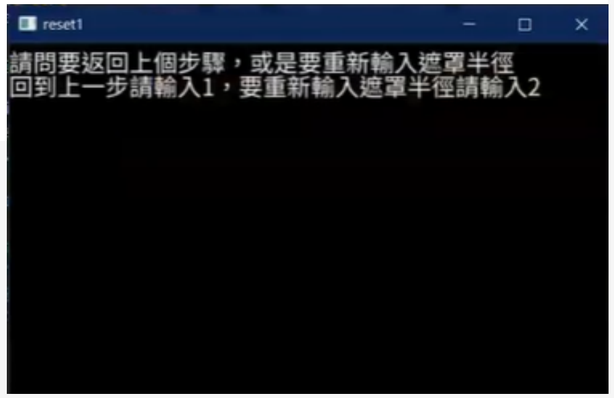



# 程式碼

```
word6 = ['右圖1為原圖,右圖2為經過邊緣偵測處理的圖', '右圖3為成品圖', '按下任意鍵關閉']
finish_imgPil = Image.fromarray(np.zeros((300, 500), np.uint8))
finish_draw = ImageDraw.Draw(finish_imgPil)

for i in range(3):
    finish_draw.text((0, i*20),word6[i], fill=(255), font=ImageFont.truetype(fontpath, 20) )
finish_control = np.array(finish_imgPil)
imgs = [finish_control, img, edges_img, reduce_noise_img]
titles2 = ['introduce2', 'original2', 'canny', 'reduce_noise']

for i in range (4):
    cv2.namedWindow(titles2[i], 0)

    if i == 0:
        cv2.resizeWindow(titles2[i], 500, 300)
        cv2.moveWindow(titles2[i], 200, 300)
    else:
        cv2.resizeWindow(titles2[i], 300, 300)
        cv2.moveWindow(titles2[i], 300*i+300, 300)
    # 按下任意鍵則關閉所有視窗
    cv2.imshow(titles2[i], imgs[i])

cv2.waitKey(0)
cv2.destroyAllWindows()
```

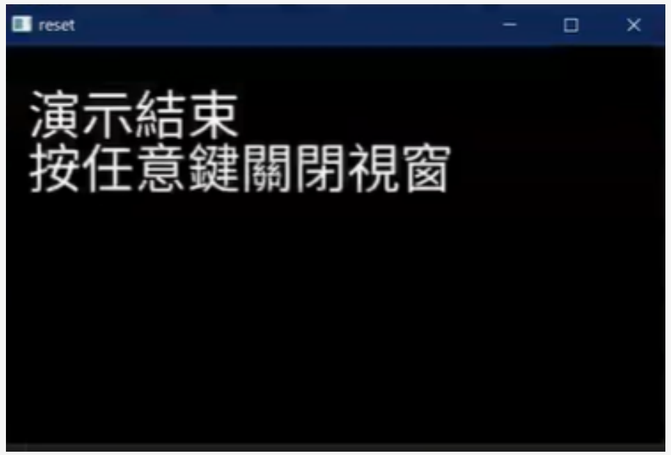



```
word7 = ['請問要返回上個步驟，或是要重新輸入遮罩半徑', '回到上一步請輸入1', '要重新輸入遮罩半徑請輸入2', '不重新輸入遮罩半徑請輸入3', '你輸入的值 :  ']
finish_imgPil2 = Image.fromarray(np.zeros((300, 500), np.uint8))
finish_draw2 = ImageDraw.Draw(finish_imgPil2)
cv2.namedWindow('reset1', 0)
cv2.resizeWindow('reset1', 500, 300)
cv2.moveWindow('reset1', 500, 300)

for i in range(2):
    finish_draw2.text((0, i*40), word7[i*2], fill=(255), font=ImageFont.truetype(fontpath, 20))
    finish_draw2.text((0, (2*i+1)*20), word7[2*i+1], fill=(255), font=ImageFont.truetype(fontpath, 20))
    if(i):
        finish_draw2.text((0, 80), word7[4], fill=(255), font=ImageFont.truetype(fontpath, 20))
    finish_control2 = np.array(finish_imgPil2)

    if i == 0:
        cv2.imshow('reset1', finish_control2)
        cv2.waitKey(2000)
        cv2.destroyAllWindows()

    elif i == 1:
        while(1):
            cv2.imshow('reset', finish_control2)
            control2=cv2.waitKey(0)-48
            cv2.destroyAllWindows()

            if 1 <= control2 <= 3 :
                menu = control2
                break

            else:
                cv2.imshow('reset',wrong_picture)
                cv2.waitKey(2000)
                cv2.destroyAllWindows()
```



# 程式碼

```
elif menu == 3:
    finish_imgPil3 = Image.fromarray(np.zeros((300, 500), np.uint8))
    finish_draw3 = ImageDraw.Draw(finish_imgPil3)
    finish_draw3.text((20, 20), '演示結束', fill=(255), font=ImageFont.truetype(fontpath, 40) )
    finish_draw3.text((20, 60), '按任意鍵關閉視窗', fill=(255), font=ImageFont.truetype(fontpath, 40) )
    finish_control3 = np.array(finish_imgPil3)
    cv2.imshow('reset', finish_control3)
    cv2.waitKey(0)
    cv2.destroyAllWindows()
    break
```



# Github



**return step when wrong threshold**
409416541 committed 2 weeks ago

Commits on Dec 11, 2023

**Create test2.py**
kiritotototo committed 2 weeks ago

**Create NotoSansTC-Regular.ttf**
kiritotototo committed 2 weeks ago

**Create images.jpg**
kiritotototo committed 2 weeks ago

**Delete test2.py**
kiritotototo committed 2 weeks ago

**Delete NotoSansTC-Regular.ttf**
kiritotototo committed 2 weeks ago

**Delete images.jpg**
kiritotototo committed 2 weeks ago

**Create test2.py**
409416137 committed 2 weeks ago

**Create NotoSansTC-Regular.ttf**
409416137 committed 2 weeks ago

**Create images.jpg**
409416137 committed 2 weeks ago

Commits on Dec 7, 2023

**update**
Lia1526 committed 3 weeks ago

409416541

409416541 周騏軍
大部分程式碼

kiritotototo

409411369 黃柏熏
大部分程式碼

Lia1526 Ting

409040879 徐慧婷
簡報製作、報告

409416137 calvinchiu

409416137 趙碩政
少部分程式碼


Thank you!