

Computer Vision Homework 5

R08922079 資工所一 洪浩翔

Part 0

```
def readImg(filename='lena.bmp'):
    #read img
    image = cv2.imread(filename, cv2.IMREAD_GRAYSCALE)
    print('shape:', image.shape)
    return image

def ImgPreProcess(image):
    #padding
    w, h = image.shape
    gray = np.pad(image, ((2,2),(2,2)), 'constant', constant_values=0)
    print('padding shape:', gray.shape)

    #kernel
    Dilkernel = np.array([[0,1,1,1,0],[1,1,1,1,1],[1,1,1,1,1],[1,1,1,1,1],[0,1,1,1,0]])
    Erokernel = np.array([[255,1,1,1,255],[1,1,1,1,1],[1,1,1,1,1],[1,1,1,1,1],[255,1,1,1,255]])

    return gray, Dilkernel, Erokernel, image.shape
```

Image reading and pre-processing

Part 1

(a) Dilation



Lena



Dilation

```
def dilation(image, kernel, shape):
    #dilation on gray image
    dilImg = np.zeros(shape)
    w,h = image.shape
    for i in range(2,w-2):
        for j in range(2,h-2):
            dilImg[i-2, j-2] = np.amax(image[i-2:i+3, j-2:j+3]*kernel)
    return dilImg
```

Code for dilation

(b) Erosion



Lena



Erosion

```
def erosion(image, kernel, shape):
    #erosion on gray image
    eroImg = np.zeros(shape)
    w,h = image.shape
    for i in range(2,w-2):
        for j in range(2,h-2):
            eroImg[i-2, j-2] = np.amin(image[i-2:i+3, j-2:j+3]*kernel)
    return eroImg
```

Code for erosion

(c) Opening



Lena



Opening

```
def opening(image, Dilkernel, Erokernel, shape):
    #erosion then dilation
    eroImg = erosion(image, Erokernel, shape)
    eroImg = np.pad(eroImg, ((2,2),(2,2)), 'constant', constant_values=0)
    openImg = dilation(eroImg, Dilkernel, shape)
    return openImg
```

Code for opening

(d) Closing



Lena



Closing

```
def closing(image, Dilkernel, Erokernel, shape):
    #dilation than erosion
    dilImg = dilation(image, Dilkernel, shape)
    dilImg = np.pad(dilImg, ((2,2),(2,2)), 'constant', constant_values=0)
    closImg = erosion(dilImg, Erokernel, shape)
    return closImg
```

Code for closing

```
if __name__ == "__main__":
    image = readImg()
    gray, Dilkernel, Erokernel, shape = ImgPreProcess(image)
    dilImg = dilation(gray, Dilkernel, shape)
    print('dilation shape:', dilImg.shape)
    cv2.imwrite("dilation.jpg", dilImg)
    eroImg = erosion(gray, Erokernel, shape)
    print('erosion shape:', eroImg.shape)
    cv2.imwrite("erosion.jpg", eroImg)
    openImg = opening(gray, Dilkernel, Erokernel, shape)
    cv2.imwrite("opening.jpg", openImg)
    closImg = closing(gray, Dilkernel, Erokernel, shape)
    cv2.imwrite("closing.jpg", closImg)
```

Code for main