

Computer Vision I

Homework 10 - Zero Crossing Edge Detection

B04902090 資工四 施長元

Usage of the full code:

```
python3 main.py [Image_path]
```

After the code exit, output file will be in the directory where you execute the code.

Environment: Python3.7 on Windows Linux Subsystem (Ubuntu 18.04.1)

Contents:

Write the following programs to detect edge:

Code for function do all the Laplace mask:

```
def Laplace(img_o, kernel, thres):  
    pad = len(kernel) // 2  
    img_t = np.ones(shape, dtype=np.int16)  
    temp_val = np.zeros(img_o.shape, dtype=np.int16)  
    for i in range(shape[0]):  
        for j in range(shape[1]):  
            accum = np.sum(img_o[i+5-pad:i+6+pad, j+5-pad:j+6+pad] * kernel)  
            temp_val[i+5, j+5] = 1 if (accum > thres) else -1 if (-accum > thres) else 0  
    for i in range(shape[0]):  
        for j in range(shape[1]):  
            if temp_val[i+5, j+5] == 1 and np.isin(-1, temp_val[i+5-pad:i+6+pad, j+5-pad:j+6+pad]):  
                img_t[i, j] = 0  
    return img_t
```

1. Laplace Mask1



```
Mask1 = [[0, 1, 0], [1, -4, 1], [0, 1, 0]]  
cv2.imwrite("hw10_LaplaceMask1.bmp", Laplace(img, Mask1, 15)*255)
```

Kernel

Threshold

2. Laplace Mask2



```
Mask2 = [[1, 1, 1], [1, -8, 1], [1, 1, 1]]  
cv2.imwrite("hw10_LaplaceMask2.bmp", Laplace(img, Mask2, 45)*255)
```

Kernel

Threshold*3

3. Minimum variance Laplacian: 20



```
Mini = [[2, -1, 2], [-1, -4, -1], [2, -1, 2]]  
cv2.imwrite("hw10_MinimumVarianceLaplacian.bmp", Laplace(img, Mini, 60)*255)
```

Kernel

Threshold*3

4. Laplace of Gaussian: 3000



Kernel

```
LG = [ [0, 0, 0, -1, -1, -2, -1, -1, 0, 0, 0] ,  
        [0, 0, -2, -4, -8, -9, -8, -4, -2, 0, 0] ,  
        [0, -2, -7, -15, -22, -23, -22, -15, -7, -2, 0] ,  
        [-1, -4, -15, -24, -14, -1, -14, -24, -15, -4, -1] ,  
        [-1, -8, -22, -14, 52, 103, 52, -14, -22, -8, -1] ,  
        [-2, -9, -23, -1, 103, 178, 103, -1, -23, -9, -2] ,  
        [-1, -8, -22, -14, 52, 103, 52, -14, -22, -8, -1] ,  
        [-1, -4, -15, -24, -14, -1, -14, -24, -15, -4, -1] ,  
        [0, -2, -7, -15, -22, -23, -22, -15, -7, -2, 0] ,  
        [0, 0, -2, -4, -8, -9, -8, -4, -2, 0, 0] ,  
        [0, 0, 0, -1, -1, -2, -1, -1, 0, 0, 0] ]  
cv2.imwrite("hw10_LaplaceOfGaussian.bmp", Laplace(img, LG, 3000)*255)
```

Threshold

5. Difference of Gaussian: 1

