**電腦視覺**

**Homework 10**

**Contents:**

1. Description
2. Methodology
3. Code fragment
4. Images

Description

This is the report for the tenth homework for Computer Vision 2019. It is completed in full.

The environment used is Windows, with the code being written in Python 3 with the cv2 module.

Methodology

The method described in the slides (on the homework website) is followed.

The resulting Python program, **hw10.py**, saves 5 images: "laplace1", "laplace2", "minvarLaplace", "LaplaceGaussian", "DiffOfGaussian" and then a ".bmp" at the end of each.

"laplace1" and "laplace2" refer to the 0, 1, 0, 1, -4, 1, 0, 1, 0 and 1, 1, 1, 1, -8, 1, 1, 1, 1 masks respectively. The other names are self-explanatory.

The thresholds follow the suggested values on the homework webpage.

Each border is extended using BORDER\_REPLICATE, i.e. the method in the slides.

Code Fragment

The fragment shows the operation for the first three pictures; the last two are minor variations.

1. **def** laplace(img, mask, threshold: int):
2. **def** has\_neg\_1(gradient, i, j):
3. neighbours = np.zeros((3, 3))
4. **for** delta\_i **in** range(-1, 1+1):
5. **for** delta\_j **in** range(-1, 1+1):
6. neighbours[1+delta\_i, 1+delta\_j] = gradient[i+delta\_i, j+delta\_j] **if** (delta\_i, delta\_j) != (0,0) **else** 87
7. **return** np.any(neighbours == -1)
9. extended = cv2.copyMakeBorder(img, 1, 1, 1, 1, cv2.BORDER\_REPLICATE).astype(np.float)
10. gradient = img.copy().astype(np.float); new\_img = img.copy(); h, w = new\_img.shape[:2]
11. **for** i **in** range(h):
12. **for** j **in** range(w):
13. n, m = i+1, j+1
14. total = np.sum(extended[n-1:n+2,m-1:m+2] \* mask)
15. gradient[i,j] = 1 **if** total >= threshold **else** -1 **if** total <= -threshold **else** 0
16. gradient = cv2.copyMakeBorder(gradient, 1, 1, 1, 1, cv2.BORDER\_REPLICATE)
17. **for** i **in** range(h):
18. **for** j **in** range(w):
19. n, m = i+1, j+1
20. new\_img[i,j] = 0 **if** gradient[n,m] == 1 **and** has\_neg\_1(gradient, n, m) **else** 255
21. **return** new\_img

Images

Each image is identified via its edge detection operator and the threshold used:

|  |  |  |
| --- | --- | --- |
| Laplace 1 (15) | Laplace 2 (15) | Minimum Variance Laplacian (20) |
|  |  |  |

|  |  |
| --- | --- |
| Laplace of Gaussian (3000) | Difference of Gaussian (1) |
|  |  |