

Computer Vision Homework 10

How to run it

```
python image.py lena.bmp
```

Principal code fragment

```
def calculateKernel(neighbors, mask, sizes, threshold, alpha):
    result = 0
    for y in xrange(sizes[1]):
        for x in xrange(sizes[0]):
            result += (neighbors[x][y] * mask[x][y])
    result *= alpha
    if result > threshold:
        return 1
    elif result < -threshold:
        return -1
    else:
        return 0

def checkNeighbors(position, labels, sizes):
    x = position[0]
    y = position[1]
    half = sizes[0]/2

    if labels[x][y] == 1:
        for row in xrange(-half, half+1):
            for col in xrange(-half, half+1):
                if labels[x+col][y+row] == -1:
                    return 0

    return WHITE
```

Description

implement Laplacian, Minimum Variance Laplacian, Laplacian of Gaussian, and Difference of Gaussian(inhibitory sigma=1, excitatory sigma=3, kernel size 11x11 [1][1]) with these threshold values:

- Laplace Mask Type1: 15
- Laplace Mask Type2: 15
- Minimum variance Laplacian: 20
- Laplace of Gaussian: 3000
- Difference of Gaussian: 1

Result



Laplace Mask Type 1, threshold = 15



Laplace Mask Type2, threshold = 15



Minimum Variance Laplacian, threshold = 20



Laplace of Gaussian, threshold = 3000



Difference of Gaussian, threshold = 1