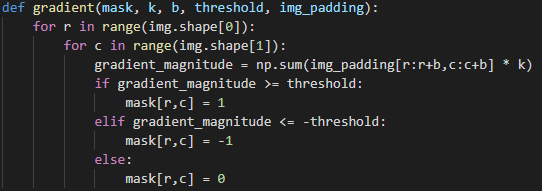
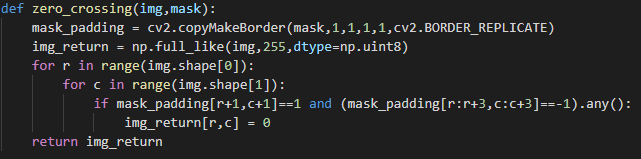
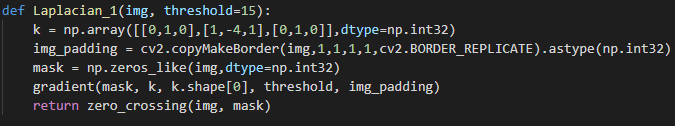
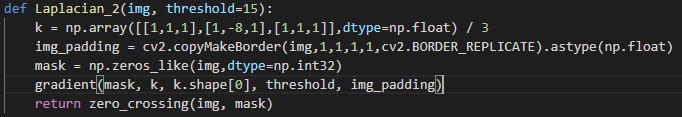
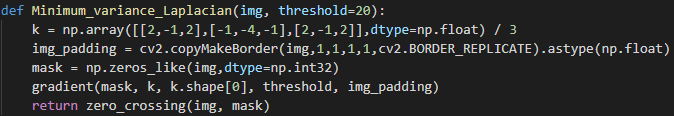
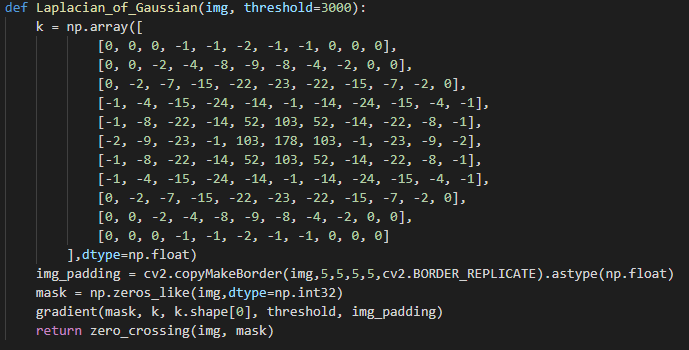
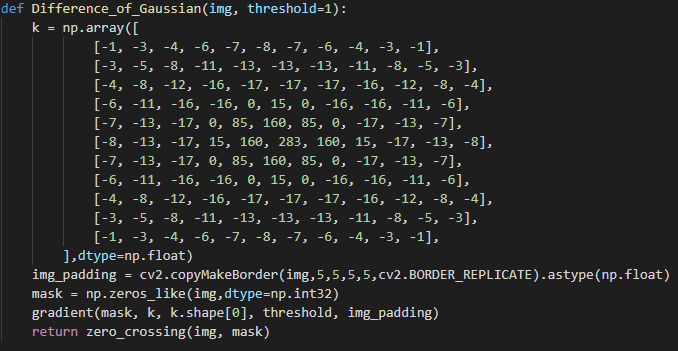
1. Problem Statement  
   Implement 2 Laplacian Mask, Minimum Variance Laplacian, Laplacian of Gaussian, and Difference of Gaussian(inhibitory sigma=3, excitatory sigma=1, kernel size 11x11). Please list the kernels and the thresholds(for zero crossing) you used.
2. Programming Tools

* Programming language: Python 3.8.5
* Library: Numpy 1.19.1, OpenCV 4.0.1

1. Problem-Solving Process  
   對image的由上到下、由左到右計算使用不同kernel去pixel gradient，若gradient大於等於threshold，則mask填1，若gradient小於等於-threshold，則mask填-1，其餘mask填0，之後再對mask做zero crossing檢測邊緣。  
   以下為計算gradient與mask的程式碼。  
     
   以下為zero crossing的程式碼，先判斷mask的value是否為1，若等於1再判斷他的八個鄰居是否有值為-1的，若有則為edge。  
     
   (a) Laplace Mask1 (0, 1, 0, 1, -4, 1, 0, 1, 0): 15  
   使用k作為kernel，並設定threshold為15。  
     
   (b) Laplace Mask2 (1, 1, 1, 1, -8, 1, 1, 1, 1): 15  
   使用k作為kernel，並設定threshold為15。  
     
   (c) Minimum variance Laplacian: 20  
   使用k作為kernel，並設定threshold為20。  
     
   (d) Laplace of Gaussian: 3000  
   使用k作為kernel，並設定threshold為3000。  
     
   (e) Difference of Gaussian: 1  
   使用k作為kernel，並設定threshold為1。  
   
2. Results  
   (a) Laplace Mask1 (0, 1, 0, 1, -4, 1, 0, 1, 0): 15  
     
     
   (b) Laplace Mask2 (1, 1, 1, 1, -8, 1, 1, 1, 1): 15  
     
     
     
   (c) Minimum variance Laplacian: 20  
     
     
   (d) Laplace of Gaussian: 3000  
     
     
     
     
   (e) Difference of Gaussian: 1

