School of Computing
National University of Singapore
CS4243 Computer Vision and Pattern Recognition
Semester 1, AY 2016/17

Lab 3 – Edge Detection

Objective:

To understand the materials covered in the lecture through

- Implementing edge detection and thinning using python codes

Preparation:

- Download the file lab3.zip from IVLE into your working directory. Uncompress the file and you should find the following document and pictures: example.jpg, prewit_result.jpg, sobel_result.jpg, thinned_sobel_result.jpg, test1.jpg, test2.jpg, test3.jpg.

Edge Detection and thinning

This is an exercise to make sure you understand edge detection and thinning. You must write python code to do edge detection and thinning on images. Specific instructions are:

- You can only use OpenCV for the following, and only for the following:
  - read an image using cv2.imread
  - write an image using cv2.imwrite

- You are not allowed to use any other methods in OpenCV or any other packages other than python and the associated Numpy.

- You must implement edge detection and thinning by writing the python codes by yourself (i.e. you cannot get the codes from elsewhere).

  Note the following:

  - You are required to define a 2D convolution function MyConvolve as follows:

```
def  MyConvolve(img, ff):
        result = np.zeros(img.shape)
        :
        :
        :
return result
```

- For edge detection, implement both Sobel edge detector and Prewit edge detector. You are required to make use of the MyConvolve function defined above. Examples of result are as shown in Fig.1 (original image), Fig.2 (Sobel edge detection result), and Fig.3 (Prewit edge detection result).

- For edge thinning, you should implement a non-maximal suppression technique. You may do a simplified version of edge thinning: an edge pixel will not be suppressed if its edge strength is the maximum **either** along the horizontal direction **or** the vertical direction (i.e. not necessary to be maximum along both directions). Example of thinning result is as shown in Fig.4.

- You need to run your codes to do edge detection and thinning on all the 3 test pictures (test1.jpg, test2.jpg, test3.jpg). Note that for this lab, you need to read these color pictures into grayscale before doing edge detection and thinning.

Submission Instruction

Submit the following:

1. Print-out of your Python codes.

2. Save the edge detection and thinning results into files.

3. Submit the softcopy of your Python codes and the resultant image files to IVLE.
   - Please put your python codes and images in a folder and submit the folder. Use the following convention to name your folder: *StudentNumber_yourName_Lab3*. For example, if your matriculation number is A1234567B, and your name is Chow Yuen Fatt, for this lab, your file name should be *A1234567B_ChowYuenFatt_Lab3*.

Please remember to write your name on the hardcopy print-out.

Fig.1 Input Image.

Fig.2 Sobel edge detector result

Fig.3 Prewit edge detector result

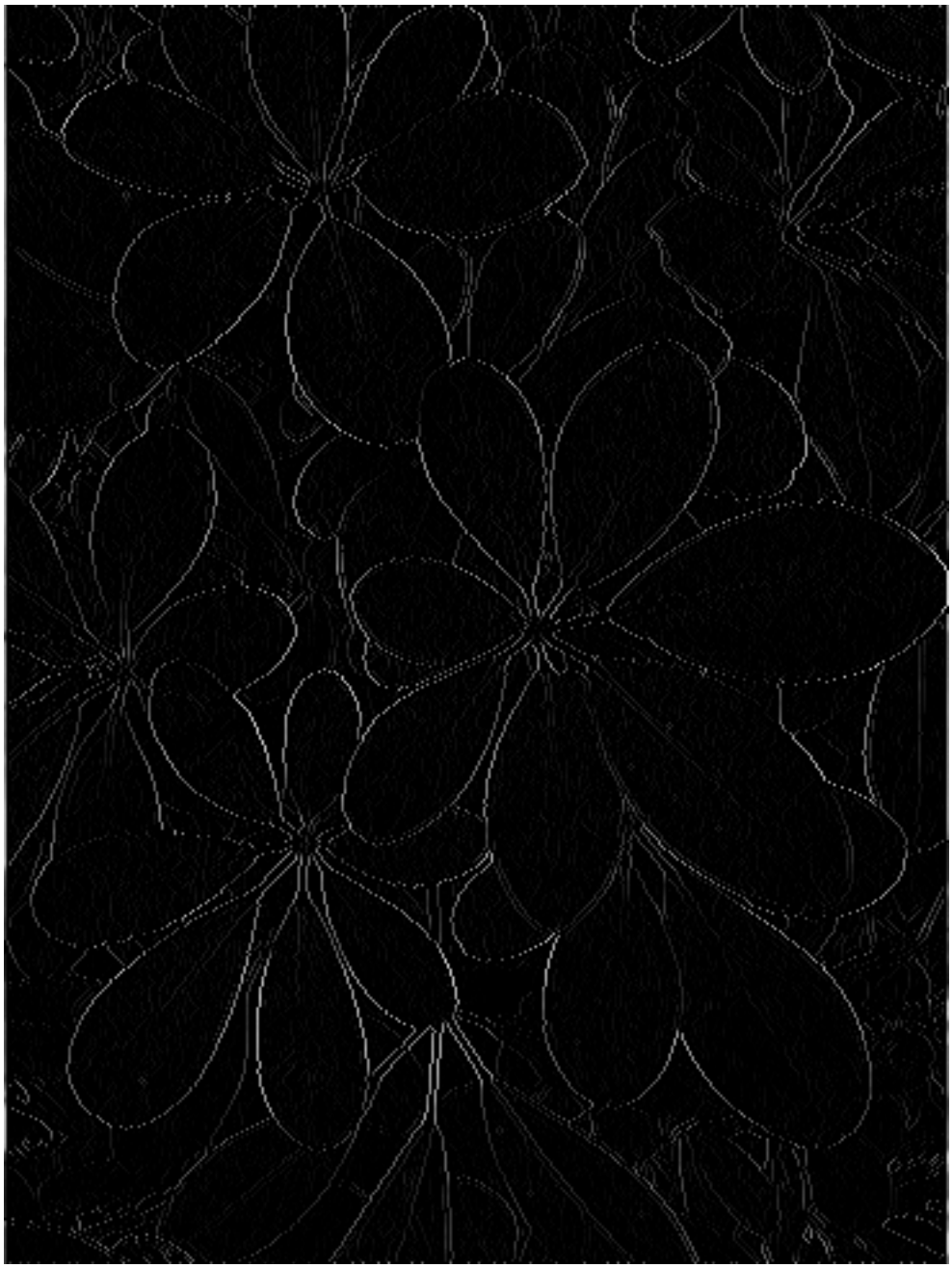

Fig.4 Result of edge thinning