School of Computing
National University of Singapore
CS4243 Computer Vision and Pattern Recognition
Semester 1, AY 2017/18

Lab 4 Corner Detector

Due date: 22nd Sep 2017, 2359hrs

Objective:

To understand and implement the corner detector that we learned in class.

Preparation:

- Download the file pic.zip from IVLE into your working directory. Uncompress the file and you should find the following pictures: leaves_gray.jpg, flower_pot_gray.jpg, orchid_gray.jpg

Steps

The following shows the detailed steps of the corner detector that you will implement in this lab:

- Get the horizontal edge strength of every point in the image and store in *gx*.
  - You can use the approximation *gx = pic(x+1,y) – pic(x,y)*

- Get the vertical edge strength of every point in image and store in *gy*.
  - You can use the approximation *gy = pic(x, y+1) – pic(x,y)*

- Compute the product of derivatives
  - *I_xx = gx * gx*
  - *I_xy = gx * gy*
  - *I_yy = gy * gy*

- Define a Gaussian kernel of size 13. You may use the following code:
  fullwin = 13;
  *gkern = gausswin(fullwin) * gausswin(fullwin).';*

- Convolve *I_xx, I_xy, I_yy* with the above Gaussian kernel. Place the results in *W_xx, W_xy, W_yy* respectively. You may use the Matlab built-in convolution function *conv2.*
  - *Note: please check the size of the output matrix after the convolution. It will be larger than the input matrices. State how much larger is the size of the output matrix compared to the input matrices. What is the relationship of this increase compared with the Gaussian window size?*

- For each of the pixels in *W_xx, W_xy, W_yy*, perform the following:

  - Obtain the W matrix as follows:

$$W = \begin{pmatrix} W_{xx} & W_{xy} \\ W_{xy} & W_{yy} \end{pmatrix}$$

  - Compute the eigenvalues of W (using the Matlab function *eig*) and store the smaller of the eigenvalues into a matrix called eig_min.

- In the matrix *eig_min* containing all the smaller eigenvalues, divide this matrix into a mosaic of *13x13* regions. For each region, keep only the highest value and set all other values to *0.*

- Find the top 200 values in the matrix *eig_min*, and mark these locations on the picture. You may use the following Matlab commands:

  - *reshape*
    - *for arranging a 2D matrix into a 1D vector*
  - *sort*
    - *for sorting the eigenvalues*
  - *find*
    - *for finding the locations in a matrix where the values are above the cut-off eigenvalues.*
  - *cat*
    - *for constructing a 3 channel RGB image from a single gray scale image, eg. cat(3, grayPic, grayPic, grayPic). This allows you to use color to mark the feature locations later (see rectangle below).*
  - *rectangle*
    - *for marking the location of good features (i.e. corners)*
    - *please mark the corners using red rectangles*

Question *(This question carries 2 marks).*

In the lecture, we formed the following matrix before we do eigen-decomposition to obtain the eigenvalues:

$$\sum_x \sum_y \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

In our experiment steps above, we seemed to be missing the double summation. Did we? Explain your answer.

Instructions

- Please perform corner detection on all the 3 pictures
- Please answer the last question and append your answer to your matlab codes using comment lines.
- You must implement the corner detector by writing the Matlab codes all by yourself (i.e. you cannot get the codes from elsewhere).

<u>Submission Instruction</u>

At the end of your lab session, submit the softcopy of your Matlab codes and the pictures marked with good features (i.e. corners) to IVLE.

Please put your files in a folder and submit the folder. Use the following convention to name your folder:

> StudentNumber_yourName_Lab#. For example, if your student number is A1234567B, and your name is Chow Yuen Fatt, for this lab, your file name should be A1234567B_ChowYuenFatt_Lab4.