Due: 16^{th} **March 2016 11:55 PM**

Objective

At the end of this assignment you should have a function that takes an image as input and returns the location of corners in the image. The function could have other input parameters that affects the accuracy of your Harris Corner detector.

Harris Corner Detector Re-cap

The Harris matrix at a pixel is given by:

$$H = \sum_u \sum_v w(u,v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{1}$$

Where u and v span the neighbors of the pixel and $w(u,v,)$ is a Gaussian weighting function. I_x and I_y are the x- and y- gradients of the image. A measure of "cornerness" of a pixel is given by:

$$M = \frac{determinant(H)}{trace(H) + \epsilon} \tag{2}$$

ϵ is a small positive number to avoid division by zero. A pixel is a corner if the measure M for the pixel is high.

Implementation

1. **"Cornerness" measure** **3 points**

   (a) Convert the input image to gray scale.

   (b) Compute I_x and I_y using central difference x-gradient and y-gradient filters respectively. You had implemented these filters in Assignment 1.

   (c) Compute I_x^2, I_y^2 and $I_x I_y$.

   (d) Create a Gaussian smoothing filter (w in Equation (1)) using `fspecial` with a chosen standard deviation (σ) and size 4σ.

   (e) Apply the Gaussian filter to I_x^2, I_y^2 and $I_x I_y$ using `imfilter`.

   (f) Compute the cornerness measure M. Recall that $determinant([a,b;b,c]) = ac - b^2$ and $trace([a,b;b,c]) = a + c$. M should be map of cornerness, the same size as the image.

2. **Corner extraction** **5 points**

(a) Perform non-maximal suppression on M to find local maximas. You could implement the local maxima detection by various means. A few suggested Matlab functions are: `ordfilt2`, `imdilate`, `imregionalmax`.

(b) Find the coordinates of the corner points.

(c) Display the image and superimpose the corners.

Show results by applying your function on the attached 'chessboard.jpg' image. Adjust your parameters (Gaussian filter σ and any parameter involved in the non-maximal suppression step) so that all the corners of the checkerboard pattern are detected.

Note: Avoid for-loops at all costs. The above steps can be performed without any for-loop.

Rotation and Scaling

2 points

1. Rotate the chessboard image by 30 deg and apply your function.
2. Resize the chessboard image by 4 times on both axes and apply your function.

Comment on the above two outputs.

Submission Instructions

Every student must submit following 2 files:

- An organized report submitted as a PDF document. The report should describe the implementation, issues (problems encountered, surprises), and an analysis of the test results (interpretation of effects of varying parameters, different image results). Intermediate and final results must be provided.
- A ZIP file containing the necessary codes.

The heading of the PDF file should contain the assignment number and topic. Also, attach a photo of yourself at top-left of the PDF along with your name and department.

Late Submission Policy

Assignments are expected to be submitted on the due date. Each student gets a total of 3 late days that can be used however you wish. For examples, all 3 days can be used towards 1 assignment or 1 day late for 3 assignments or other combinations. Late submissions beyond that will be penalized as below:

- One day late will be penalized 25% of the credit.
- Two Days late will be penalized 50%.
- Submissions more than 2 days late will not be considered for credit.

I will be ruthless in enforcing this policy. There will be no exceptions

Collaboration Policy

I encourage collaboration both inside and outside class. You may talk to other students for general ideas and concepts but the programming must be done independently. For mid-term and final examination there will be no collaboration permitted.

Plagiarism

Plagiarism of any form will not be tolerated. You are expected to credit all sources explicitly. If you have any doubts regarding what is and is not plagiarism, talk to me.