National Tsing Hua University
Department of Electrical Engineering
EE3662 Digital Signal Processing Laboratory, Fall 2022

Lab 11 Image Filtering and Corner Detection

Assigned on Dec 5, 2022

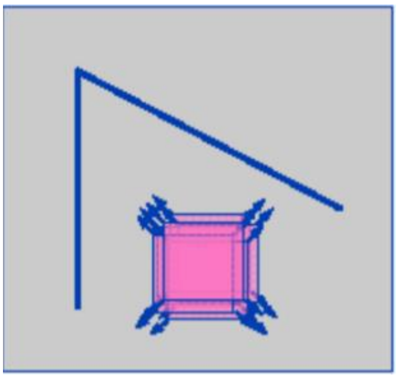
Due by Dec 12, 2022

Overview

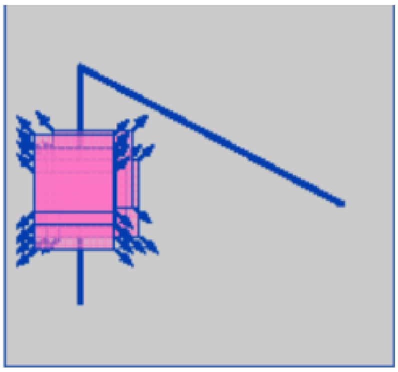
The goal of this lab is to use image filters to calculate image gradients. Then, we combine image gradients and apply Gaussian filter to compute second moment matrix at each pixel. Next, we can compute the corner response function. Finally, we apply a threshold and non-maximum suppression to obtain distinctive corner locations.

(Left: original image. Right: image overlapped with corners.)

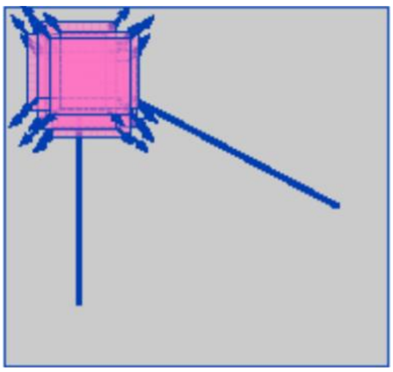
Ideas and Derivation



Flat region → no change in all directions



Edge → no change along the edge direction.



Corner → significant change in all directions

The change of intensity for the shift E(u,v) is given by

$$E(u,\mathrm{v}) = \sum_{x.y} w(x,y)|I(x+u,y+v) - I(x,y)|^2,$$

where w(x,y) is window function. By the first order approximation of Taylor Series for 2D functions, we have

$$I(x+u,y+v) \cong I(x,y) + uI_x(x,y) + vI_y(x,y),$$

and then we can get the equation as follow

$$E(u,\mathrm{v}) = \sum_{x,y} w(x,y)(u^2I_x^2 + 2uvI_xI_y + v^2I_y^2).$$

Rewrite it as a matrix equation

$$E(u,\mathrm{v}) = \sum_{x,y} (u \quad v) w(x,y) \begin{pmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$

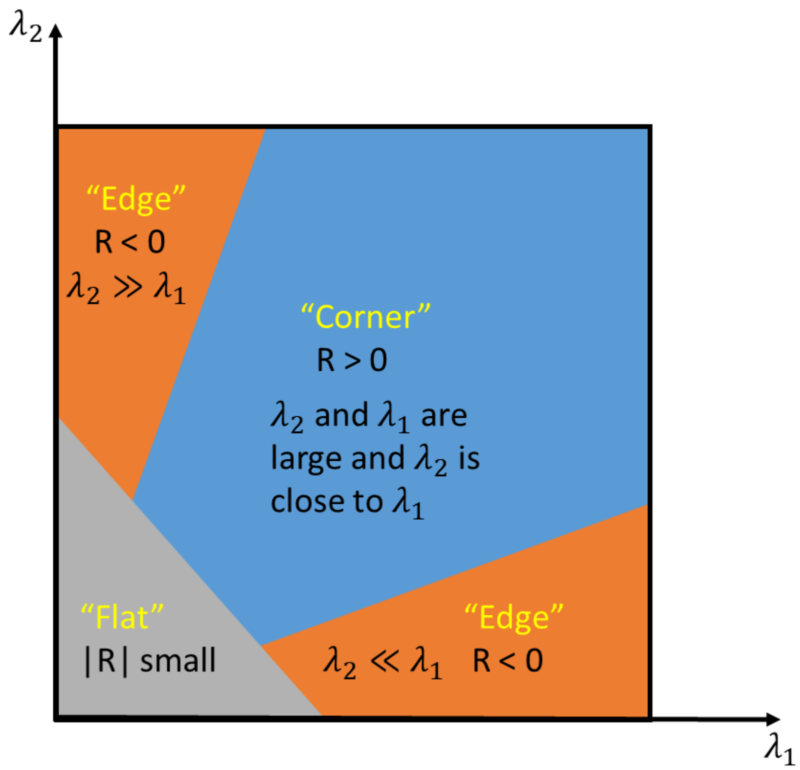
where $A = \begin{pmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{pmatrix}$ is called Harris Matrix

Measurement of corner response is given by

$$\mathrm{R} = \det(\mathrm{A}) - \mathrm{k}(\mathrm{trace}(\mathrm{A}))^2 = \lambda_1\lambda_2 - \alpha(\lambda_1+\lambda_2)^2,$$

where λ_1, λ_2 are the eigenvalues of $\begin{pmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{pmatrix}$.

We use eigenvalues and R's value to classify image points as below



Procedure

I. Grayscale

Change the RGB to Grayscale by $R * 0.299 + G * 0.587 + B * 0.114$.

II. Get Image gradient

Use horizontal and vertical gradient filter to get I_x, I_y

III. Get Gaussian smoothed I_x^2, I_y^2, I_xI_y

Use Gaussian filter to get Gaussian smoothed I_x^2, I_y^2, I_xI_y.

IV. Calculate corner response R, and map R to 0 ~ 1000

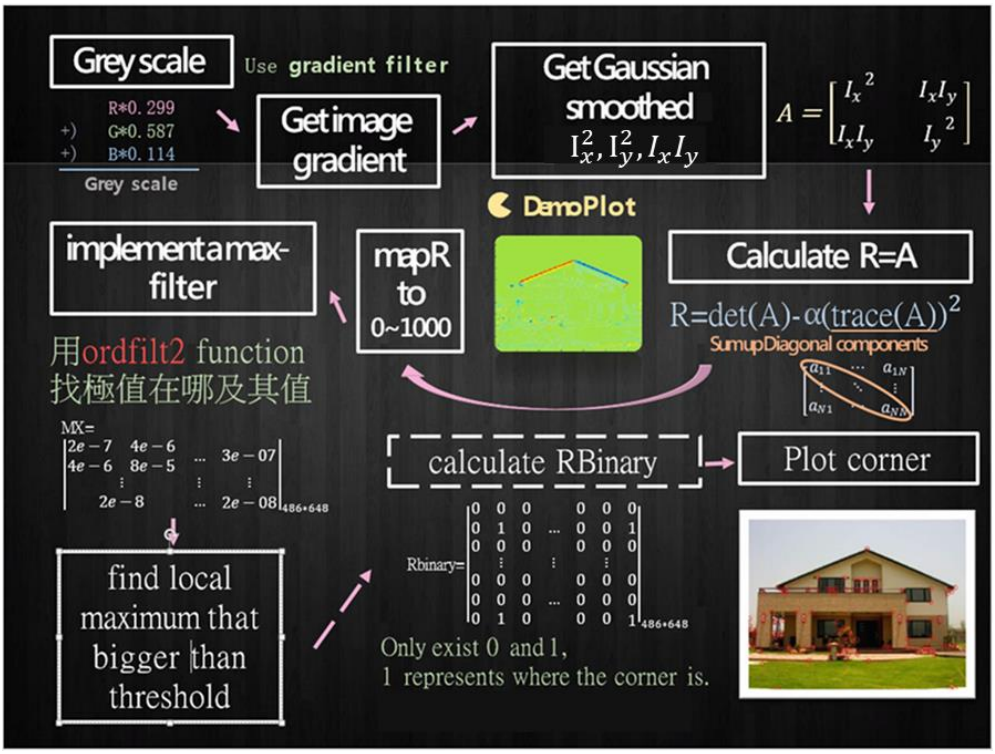
Calculate $R = det(A) - \alpha\left(trace(A)\right)^2$, where α is 0.04 empirically and trace(A) is the sum of diagonal components of A.

V. Find local maximum

Use `ordfilt2()` function to find the local maximum.

VI. Calculate RBinary and plot corners

RBinary = $\begin{bmatrix} 0 & 0 & 0 & & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & & 0 & 0 & 0 \\ & \vdots & & \vdots & & \vdots & \\ 0 & 0 & 0 & & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & & 0 & 0 & 1 \end{bmatrix}$, and 1 represents where the response is larger than the threshold, which stands for where the corner is.



(Overall procedure)

In-class Demo

1. (15%) Calculate and plot Gaussian smoothed $I_x I_y$.
2. (20%) Calculate RBinary and generate the image with corner detection.
3. (15%) Implement in **`CalculateRotate.m`**. Given an image pair (**`data/img_1.jpg and data/img_2.jpg`**), please use Harris corner detector to find corners and estimate yellow object's rotation **degree (i.e. θ_2-θ_1 in the figure below)**.

Note: You can directly use built-in functions (e.g. **`Imfilter(), filter2()`**) to perform filtering.

Report

1. (6%) In addition to provided image (**`data/im.jpg`**), show at least **two** different corner image generation results and each needs to contain the input image, Gaussian smoothed $I_x I_y$ and corner detection results.
2. (10%) Please refer to the definition of **"edge"** in the figure above (2nd page) to find edge in the given image, Im.jpg (put your code in **`Lab11/code/FindEdge.m`**). An example output is given below.

3. (12%) Analyze the corner detection results using Rectangular window function and try to discuss

the differences between Rectangular window function and Gaussian window function. Besides, please try your own window function and explain the reason why you choose it.

Rectangular window function (use the same size as the provided Gaussian filter):

1	1	…	1
1	1	…	1
…	…	…	…
1	1	…	1

4. (7%) What if we don't use any window functions to smooth I_x, I_y, I_xI_y?
5. (12%) Analyze the corner detection results using Prewitt and Scharr gradient filter. Besides, please try to discuss the differences and similarities between Prewitt and Scharr gradient filter. Below shows these two gradient filters.

Prewitt

dx

1	0	-1
1	0	-1
1	0	-1

dy

1	1	1
0	0	0
-1	-1	-1

Scharr

dx

3	0	-3
10	0	-10
3	0	-3

dy

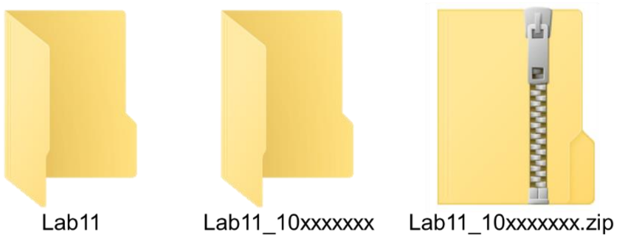
3	10	3
0	0	0
-3	-10	-3

6. (3%) Conclusion

Deliverable and file organization

Directory	Filename	Description
Lab11/code/	*.m	Matlab code
Lab11/data/	*.png / *.jpg	Your own source images
Lab11/results/	*.png / *.jpg	Your results
Lab11/report/	report_10xxxxxxx.pdf	Your report

Please organize your files according to the above table, and change the root name from Lab11 to Lab11_10xxxxxxx. Then compress it as Lab11_10xxxxxxx.zip in ZIP format.
(Note: 10xxxxxxx is your student ID)



Wrong file delivery, wrong file organization or wrong file naming will get up to 5% punishment.

Reference

[1] http://www.cse.psu.edu/~rtc12/CSE486/lecture06.pdf