

CMSC 426, Image Processing
HomeWorks 2 And 3: Math/Computer Vision?
Due on: 11:59:59PM on Saturday, Nov 05 2016

Prof. Yiannis Aloimonos,
Nitin J. Sanket and Kiran Yakkala

October 15, 2016

Questions

1. Why is a filter bank used instead of a single filter? Explain this in as much detail as you can. Feel free to use synthetic or real world examples of images or just conceptual examples if you want. **10 Pts**

2. A 2-D filter is called a ‘separable filter’ when it can be split into 2 1-D filters. For eg. The 2-D sobel kernel given by

$$S = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

This 2-D sobel kernel can be split into 2 1-D kernels given below:

$$S_v = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

$$S_h = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

Answer the following questions:

(a) How do you know if a given 2-D matrix is a separable filter. Provide a mathematical formulation to do this. **5 Pts**

(b) Consider a $m \times n$ image which has to be filtered using a $p \times q$ kernel. What are the number of multiplications and additions required to do this? Now, let us say the filter is separable and can be split into 2 1-D filters of size $p \times 1$ and $1 \times q$. We know that convolution is associative, instead of convolving the image directly with the 2-D kernel, we can convolve the image with 2 1-D kernels and obtain the same result. How many multiplications and additions required to do this? (i.e., using 2 1-D kernels instead of a 2-D

kernel). What is the advantage/disadvantage (advantage if using 2 1-D kernels is better and disadvantage if using 2-D kernel is better) of using 2 1-D kernels instead of a 2-D kernel? **5 Pts**

(c) Split the following matrix into 2 1-D kernels:

$$F = \begin{bmatrix} -4 & 20 & -12 \\ 5 & -25 & 15 \\ -8 & 40 & -24 \end{bmatrix}$$

Do not use MATLAB to do this. Solve it by hand. **5 Pts**

(d) Can a 2-D separable filter be split into different pairs of 1-D filters, i.e., does the separable operation always result in a unique output. Reason why or why not. **5 Pts**

3. In computer vision, we generally say “Difference of Gaussian (DoG) is a good approximation to Laplacian of Gaussian (LoG)”. Can you come up with an intuition why it is so? Feel free to use math/plots to support your answer. **10 Pts**

4. A circle has known radius ρ and can be assumed to be in the $Z = 0$ plane centered at $(0, 0, 0)$ origin of a world coordinate system (assume any right-handed coordinate system which will make the problem easier for you to solve, but Z should point vertically up). A drone observes the circle as an ellipse in the image plane (this is not always the case but we assume that the drone is sufficiently far). Compute the altitude of the drone in the world coordinate system as well as its distance to the circle center. The camera matrix and all the things in the image are known. Can you compute the attitude (roll, pitch and yaw angles) in this problem setup, why or why not? *Hints: Look at projective transforms, homography, equation of circle in matrix form and definition of matrix determinant in terms of matrix columns.* **40 Pts**

5. A 1-D gaussian distribution is represented as

$$\mathcal{N}(\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x - \mu)^2}{2\sigma^2}\right)$$

Now, consider that you have 2 gaussian distributions $\mathcal{N}(\mu_1, \sigma_1)$ and $\mathcal{N}(\mu_2, \sigma_2)$. Let $\mathcal{N}(\mu, \sigma) = \mathcal{N}(\mu_1, \sigma_1) \otimes \mathcal{N}(\mu_2, \sigma_2)$. Here \otimes represents the convolution operation. Represent μ as a combination of μ_1, μ_2 and σ as a combination of σ_1, σ_2 . What does this physically signify? **10 Pts**

6. Consider the image of a white square on a black background as shown in Fig. 1.

(a) Design a 2D kernel or a set of 1D kernels (a kernel is a filter you convolve with) which can detect any/all corners of the white square without detecting any other corners in the image. **15 Pts**

(b) Design a 2D kernel or a set of 1D kernels (a kernel is a filter you convolve with) which can detect only the top left and right bottom corners of the white square without detecting any other corners in the image. **15 Pts**

Feel free to use MATLAB to check if your kernel works correctly.

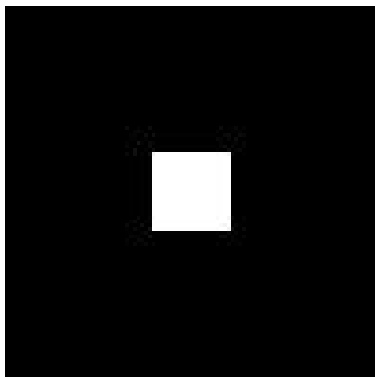


Figure 1: Figure used in Problem 6.

7. We know that RANSAC is a non-deterministic algorithm, hence knowing some statistics about RANSAC is advantageous. We are mainly concerned about the probability of success of RANSAC.

(a) Consider that the probability of picking a good set of points (we pick 4 points in panorama stitching) is p_g . What is the probability of success after N number of iterations?

3.33 Pts

(b) Now, consider that only half of the point matches are good and you pick 4 points from this to estimate homography (p_g is not half, think carefully what the value of p_g will be). How many iterations are needed to have a 95% probability of success? **3.33 Pts**

(c) What are the number of all possible combinations of 4 point pairs one can pick from the image (assume you have 100 points)? By what order of magnitude is this (Number of iterations you found in previous problem) better than trying all possible combinations?

3.33 Pts

8. What are the minimum number of points required to estimate homography transformation between 2 images? What does this physically signify? *Hints: Think of this mathematically by writing projection equation or think of it as losing a degree of freedom when you add each constraint.* **10 Pts**

9. You want a drone to go through the window shown in Fig. 2 (This figure is also given to you and is called `Window.jpg`), for this to happen you first need to detect the window. Come up with an algorithm to detect the window. You can use any built-in or third party function to do this. But please do cite the necessary source. Also, you **DO NOT** have to submit the code for this part. Explain in detail the steps you followed. If you use a third party code, you need to explain what that code does in detail instead of superficially saying 'it does something'. If you have math behind the algorithm you need to write all the equations necessary. **20 Pts**



Figure 2: Figure used in Problem 9. Taken from <https://www.youtube.com/watch?v=meSItatXQ7M&t=37s>.

10. We have defined an edge point as a point where the gradient magnitude of the image $\|\nabla I\|$ reaches a local maximum along the gradient direction. This means that the derivative of $\|\nabla I\|$ along the gradient direction $\frac{\nabla I}{\|\nabla I\|}$ has a zero crossing. Compute

$$\nabla_n \|\nabla I\| \text{ where } n = \frac{\nabla I}{\|\nabla I\|}$$

*Hint: Look up how to differentiate the magnitude of a vector $\|v\|$ with respect to the vector v , the gradient operator ∇ is a vector in this case, say $[u, v]^T$. **20 Pts***

11. An April Tag (AT) is a visual marker (like a QR code) used in computer vision and robotics extensively to estimate the 3D pose (position and orientation) of the camera with respect to the april tag (Refer to sample AT in Fig. 3). It was developed at University of Michigan and is used extensively throughout the world. My (Nitin) master's thesis was on estimating extremely accurate 3D pose of the camera using multiple april tags and a my research can be seen here: <https://daniilidis-group.github.io/penncosyvio/>. The way the AT library works is as follows, you know the size of the tag and you can easily extract the 4 extreme corner points in the image. Using this the homography is estimated from which we obtain the camera pose. Now, say the AT is at a distance of d from the camera and you estimate $d - \delta d$ due to error. The amount of error depends on your error in corner detection. Let us say the true location of the corner is p and you detect the corner in $p + \delta p$. Also assume that the focal length is f and the size of the AT as l (square tag). Find δd in terms of $\delta p, d, l, f$. What does this equation physically signify? **15 Pts**

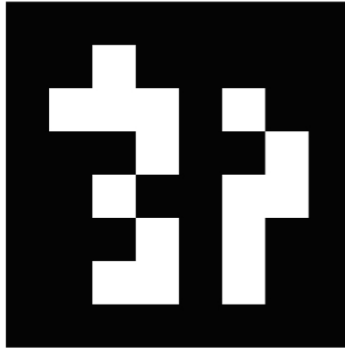


Figure 3: Figure used in Problem 11.

12. Refer to Fig. 4, is the camera above/below the parachute? Why? **5 Pts**



Figure 4: Figure used in Problem 12.

Submission Guidelines

Answer the questions in a pdf file with the naming convention `YourDirectoryID_hw2.pdf` and submit them ELMS/Canvas. **YOU NEED TO TYPESET THE ANSWERS IN**

L^AT_EX or Word, HANDWRITTEN ANSWERS WILL BE GIVEN ZERO CREDIT! FEEL FREE TO DRAW DIAGRAMS BY HAND IF YOU WANT!

Collaboration Policy

You are restricted to discuss the ideas with at most two other people. But the code you turn-in should be your own and if you **DO USE** (try not it and it is not permitted) other external codes/codes from other students - do cite them. For other honor code refer to the CMSC426 Fall 2016 website.

DON'T FORGET TO HAVE FUN AND PLAY AROUND WITH IMAGES!.