CS 663 : Digital Image Processing : Mid-Sem Examination

Instructor : Suyash P. Awate

6 Sep 2015, Tuesday, 3 pm – 5 pm
1 Cheat Sheet Allowed. Maximum Points 100

1. [30 points]

(a) (4 points) • For any discrete (spatially sampled and intensity quantized) grayscale image with integer-valued intensities, will histogram equalization make the intensity distribution uniform ? Prove or disprove.
• Can such histogram equalization change the number of unique intensities in the original image ? Briefly explain why or why not.

Yes. Please see the lecture slides.

(b) (4 points) For a denoising application, among the filters that we've covered in class, name the filter best suited for removing (i) salt-and-pepper noise and (ii) additive zero-mean Gaussian noise. Briefly justify your answer.

(i) Median filter.

(ii) Bilateral filter or patch-based filter.

(c) (6 points) Suppose we have two filter masks, say, A and B, which we want to use for convolution. Suppose we have an image X. Suppose we define image $Y := (X * A) * B$, where $*$ denotes convolution. Suppose we define image $Z := (X * B) * A$. Assume <u>full</u> 2D convolution, where the convolved image will be of a size greater than either of the input images. Assume the intensities outside the image / mask boundaries to be zero.
• Will image Y and image Z be the same for all masks A and B ? Why or why not ?
• Indicate all subsets of the following set of images: $(A * X) * B$, $(A * B) * X$, $(B * X) * A$, $(B * A) * X$, $(X * A) * B$, $(X * B) * A$, where all images in that subset are identical ? Explain why.

All these images exactly the same, because convolution is commutative and associative.

(d) (6 points) Suppose you want to perform template matching over an image to detect a certain face in a photograph (similar to the example we covered in class, in the slides). You have access to a function, namely, conv(), for convolving an image with a filter.
• Can you use the conv() function if you want to perform template matching by computing correlations ? Why or why not ?
• Can you use the conv() function if you want to perform template matching by computing <u>normalized</u> correlations ? Why or why not ?

Yes, conv() can be used instead of corr() by reflecting the filter mask about the X and Y axis.

conv() cannot replace normalized correlation, because the latter isn't a linear function of the inputs.

(e) (3 points) • Describe the filtering mask that produces "bokeh" effects in photography.
• When do we call a filter mask as "separable" ? Define this notion.
• Is the bokeh mask separable ? If so, show how.

Please see the lecture slides.

(f) (3 points) Suppose you need to blur an image, but aren't allowed to use any averaging directly in the spatial domain. Instead, you are restricted to the following operations (a) Fourier transform, (b) inverse Fourier transform, and (c) any operation on the Fourier transform in the frequency domain. How will you perform such a blurring using concepts of Fourier analysis ?

Take the Fourier transform. Multiple it with a Gaussian. Take the inverse Fourier transform.
Please see the lecture slides.

(g) (4 points) Does the Fourier transform describe a linear shift-invariant system ? Prove or disprove.

Please see the lecture slides.

2. [15 points] Explain the effects of the following algorithms / processes on the histogram of the image: (i) introduction of salt-and-pepper noise, (ii) introduction of zero-mean additive Gaussian noise, (iii) mean-shift filtering, (iv) global histogram equalization, and (v) filtering via Gaussian convolution. For each case, describe, for images in general, what happens to the (a) range of the intensities, (b) number of peaks, and (c) height of the peaks.

(1) salt-and-pepper noise: increase of 2 peaks at 0 (min) and 1 (max); other peaks lower and fewer; same intensity range.

(2) additive i.i.d. zero-mean Gaussian noise: PDF gets convolved with Gaussian. So, lower and fewer peaks; range increases.

(3) mean-shift: fewer and taller peaks; range decreases.

(4) global histeq: range cannot decrease, number of peaks cannot increase, peak heights cannot increase.

(5) Gaussian convolution: range decreases; number of peaks cannot decrease, peak heights can increase or decrease.

3. [15 points] An amateur photographer wants to take a picture of the entrance to the KReSIT building. She fixes the camera on a tripod stand and places the stand on the footpath in front of the KReSIT building. The problem is that a few students always keep passing (walking) between the camera and the KReSIT entrance, thereby obstructing the view of the camera. After a few minutes of trying, she manages to take 20 pictures without changing the camera location, trying to time them right to avoid students coming into the photograph frame. However, none of the pictures is "perfect", i.e., there is always one or two students somewhere in the frame. Also, the locations of the students are randomly distributed over the spatial coordinates of the photograph.
Given this set of "imperfect" pictures, she manages to generate a perfect picture of the entrance such that no passing student appears in the frame. Her algorithm is fully automatic (avoiding any manual intervention) and relies only on the image-processing concepts that we have discussed in the class so far.
• What was her algorithm ? Precisely specify the assumptions on acquired images the under which such a generation of the perfect picture is possible.

Algorithm: Median filtering on the vector of intensities at each pixel, where the vector comprises the intensities in each of the photographs.

Assumption: At each pixel, the number of photographs in which an obstruction appears doesn't exceed 9. At each pixel where there isn't any obstruction, the intensity remains the same across multiple photographs.

4. [10 points] Suppose you take a photograph of your friend in a playground such that the photograph comprises the face of your friend in the foreground and green grass and trees in the background — these are the only entities present in the scene. In the photograph, all entities, in both foreground and background, are in focus and exhibit sharp edges. Now, you decide to apply an artistic effect to this photograph such that only the background appears blurred in such a way that the amount of blur increases gradually with distance of the background pixel from the outline of the face. Inside the face, no blurring should occur.
   • Design and describe an algorithm to process the image to achieve this effect.

   Algorithm: Segment the face via mean-shift, assuming background (grass + trees) is green and foreground (face) is non-green. At each pixel outside the face, compute the distance between its location and the nearest pixel on the face outline. At each pixel, perform Gaussian smoothing where the sigma for the Gaussian is controlled using the distances computed before.

5. [10 points] • Explain how the algorithm of mean-shift filtering can be used for denoising ? In addition, explain how to choose the kernel bandwidth parameter and how many iterations to run ?
   • Will this method work well for texture images or piecewise-constant (textureless / cartoon-like) images or both ? Explain why.

   Mean-shift filtering can be used for piecewise smooth (texture less) images. Kernel bandwidth should be around the noise standard deviation. Number of iterations shouldn't be infinite (rather, some small number) to make sure that all intensities in an object don't converge to a single intensity (this will be bad if the object itself comprises some variation of color shades).

6. [20 points] In geology, a ridge is a line of high ground, with the land dropping away on either side. Geometrically, this is a function that looks similar like a $\_\cap\_$ (on a 1D domain) or an inverted half-cylinder with its flat side placed on the ground (on a 2D domain). If you negate / invert this function, then we get a negative ridge that is shaped like a $^{-}\cup^{-}$.
   In an image, interpreted as a function of pixel intensities over space, positive and negative ridges are important features; just like edges and corners. Consider that you want to detect ridges in an real-world image, where some amount of noise is present.
   • Design and describe an algorithm for detecting ridges in an image. Specifically, the algorithm needs to find the (a) locations of the center ridgeline, i.e., the top of the $\cap$ and the bottom of the $\cup$, and (b) direction of the ridge.

   Akin to edge location that is characterized by the first derivative being large, ridges (similar to double edges) location is characterized by the second derivatives being large.

   Just as we have talked about a direction of an edge (given by the direction of the gradient vector), we can talk about the direction of a ridge.

   For a 1D image / function, the top of a ridge is a point of a local maximum (2nd derivative is large negative). The bottom of a ridge is a point of a local minimum (2nd derivative is large positive) of the image-intensity function.

   For a real-world grayscale image, which is a function on a 2D domain, the 2nd derivatives are characterized by the Hessian matrix, following a Taylor-series expansion upto 2nd order (quadratic form).

   The Hessian is a 2x2 symmetric matrix, with orthogonal eigenvectors. For a ridge, one eigenvector will be along the ridge and another across the ridge. For the eigenvector along the ridge, the eigenvalue will be of small magnitude (close to zero). For the eigenvector across the ridge, the eigenvalue will be of large magnitude.

Another way to interpret the eigenvalues and eigenvectors is to note that the (positive) ridge function looks similar to the Mahalanobis distance function associated with a bivariate Gaussian. The Hessian is then essentially equivalent to the inverse covariance matrix, whose eigenvalues and eigenvectors will have the same properties as described earlier.