Midterm CMSC828J 2006

Assigned, 10/19/06

Due, 10/26/06

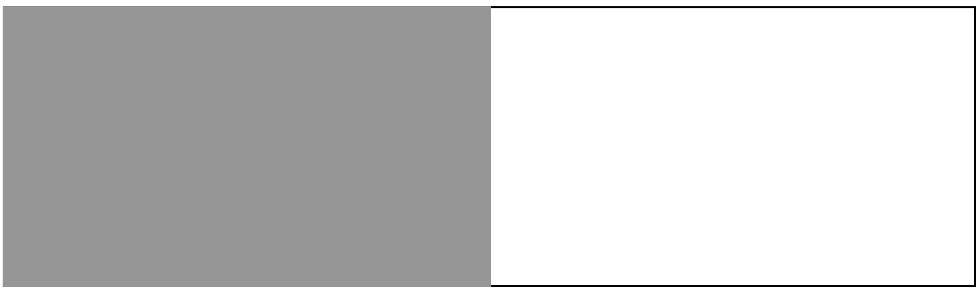
1. **Convolution:** Consider a 1D signal that is a Gaussian, centered at 0, with a standard deviation of σ_1.

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma_1^2}}$$

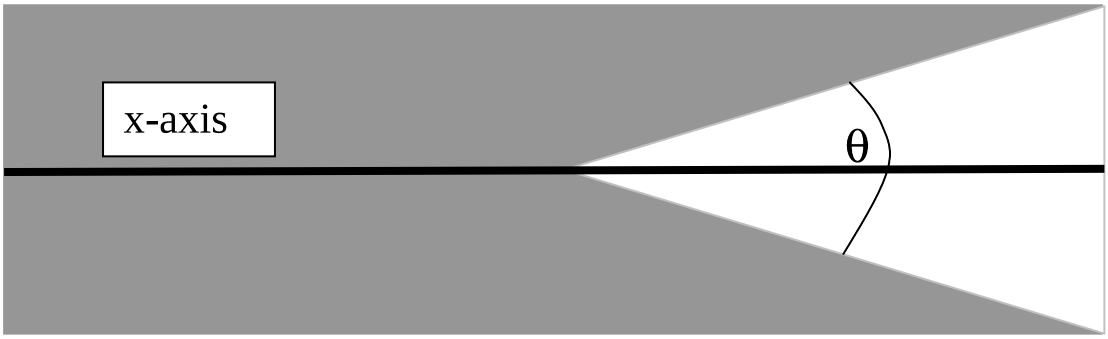
What is the result of convolving this with a Gaussian with a standard deviation of σ_2? Give an analytic expression.

2. **Diffusion:** In this problem we will consider the distribution of particles in 1D among a discrete number of positions, just as when we first learned about diffusion. Suppose we have a very large number of particles at -100 and an equally large number of particles at 100. Every second, 1% of the particles jump 1 to the left, and 1% jump 1 to the right.
   a. Write an expression for the distribution of particles after T seconds. You may approximate this with a continuous distribution, in a way that is only valid for large T.
   b. How much time will pass before the distribution is unimodal?

3. **Edge Detection:** These questions concern the Canny edge detector.
   a. Suppose we have an image that consists of a black region and a white region, separated by a vertical line. That is, imagine the image is black for negative values of *x*, and white for zero or positive values of *x*. Is there any choice of parameter settings that will result in edges consisting of two, parallel vertical lines? Give an example, or explain why this is impossible.



   b. Suppose we have a 2D image that consists of a sharp corner. On the left, the image is black. On the right, it is white, with a region that has a corner symmetric about the x axis, with an angle of theta, centered at the origin. Suppose we run the Canny edge detector, on this image. Consider the edge pixel, if any, that we will find on the x axis. Show quantitatively how the magnitude of this edge and its position will alter as we vary theta. How does the result depend on the parameters of the Canny edge detector? It is ok to answer this question by writing code, but do not use someone else's implementation of the Canny edge detector (eg., don't use matlab's edge detection code). Include any code you have written.



4. **Non-linear Diffusion:** In what significant way, if any will Perona-Malik diffusion differ in its effect from Weickert's Edge-Enhancing Anisotropic Diffusion on an image with intensities that are described by a single Gaussian distribution. By this I mean that the image looks like a Gaussian, bright at a point, and gradually darker moving away from the bright spot.

5. **Shortest path algorithms:** Suppose we have a 1D image that is generated in the following way. We first produce a signal, and then add noise to it. To get the signal, the first pixel has an integer value drawn from a uniform distribution between 0 and 255. Each subsequent pixel, with probability 49/50, has the same value as the previous pixel. With probability 1/50, a new pixel has a value drawn from a uniform distribution from 0 to 255. Next we add noise to the signal. For each pixel, we draw the noise from an i.i.d. Gaussian distribution. We would like to take the noisy signal, and segment the pixels into groups that originally had the same intensity, before noise was added. Explain how to do this using a shortest path algorithm. Be explicit about what the nodes in the graph will be and what edges will exist, along with their weights.

6. **Normalized cut:** We will perform normalized cut on an image, represented as a graph. Suppose the graph contains two groups of vertices, A and B. A contains 20 vertices, B contains 10. All vertices in A are connected to each other with a weight of 1. Likewise, all vertices in B are connected to each other with a weight of 1. No vertices in A are connected to vertices in B except for two special vertices, a in A and b in B. a and b are connected with a weight of N >= 0.

a. Explain what partition of the vertices into two groups will optimize the normalized cut criteria, NCUT. This answer will depend on N, so explain what answers may be obtained for different values of N.

b. Suppose you apply the normalized cut algorithm described in Shi and Malik in order to produce two groups. What answer would you expect to obtain, when N = 0? When N becomes extremely large? Explain your answer.

c. Suppose in addition to the previous vertices and edges, we have another region C. C also contains 10 fully connected vertices, connected with a weight of 1. C contains a single vertex, c, which is connected to a vertex b' in B, with b' different from b. For a value of N that is extremely large, what partition of the graph into two groups will optimize the NCUT criteria? Suppose you apply the normalized cut algorithm described in Shi and Malik in order to produce two groups. What answer would you obtain? Explain your answer.