

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
Department of Electrical and Computer Engineering

ECE 418 IMAGE PROCESSING

Problem Set 9
Spring 2008

Issued: Thursday, March 27, 2008

Due: Thursday, April 3, 2008

Problem 9.1

Suppose that you have an old, noisy image that you want to clean up. The image appears to contain an old man, pointing at a sign, outside a German-looking tavern, but you can't see the details clearly enough to know who the man is, or what the sign says.

- (a) As a first step, you decide to estimate the autocorrelation function. How would you estimate the autocorrelation function?
- (b) The autocorrelation function turns out to be

$$R_{yy}[\vec{n}] = 0.25 + (0.9)^{-(n_1+n_2)}$$

What is the power spectrum of the image?

- (c) Design a filter $G(\vec{\omega})$ such that under reasonable assumptions, $\hat{x}[\vec{n}] = g * y$ is an MMSE approximation of the original clean image $x[\vec{n}]$. Clearly state your assumptions.
- (d) The filter you designed in part (c) gives you a reasonable image, but not a perfect image. You decide to design an interactive image enhancement algorithm with just one slider, so that by tweaking the slider, you can search interactively for the best-looking restored image. What parameter should the slider control?
- (e) Suppose you decide to scrap everything that you've done in parts (a)-(d), because you become convinced that the image is corrupted by multiplicative rather than additive noise. What can you do to restore an image corrupted by multiplicative noise?

Problem 9.2

You have noticed that there is no company specializing in the needs of incompetent photographers, so you decide to start one. As your first product roll-out, you propose to develop a lossless compression standard for saturated images. A saturated image is a grayscale image whose pixel values, $x[\vec{n}]$, obey the following probability distribution:

$$p(x) = \begin{cases} \frac{1}{8} & x = 0 \text{ or } x = 255 \\ \frac{3}{1016} & \text{otherwise} \end{cases}$$

- (a) Compute the entropy, $H(p)$, of the distribution above.
- (b) How many bits/pixel are required by the fixed-length binary code?
- (c) Design a Huffman code for $p(x)$. How many bits/pixel are required for your code?
- (d) Why is your answer to part (c) different from your answer to part (a)?
- (e) What might you do in order to get a Huffman code whose average bit rate per pixel approaches $H(p)$?