EE569 Digital Image Processing

HOMEWORK #0
SAMPLE HOMEWORK – MULTIPLE IMAGE AVERAGING

Issued: 09/01/2010 **Due: 09/02/2010**

Problem 1: Averaging Images to Eliminate Noise (100%)

In Figure 1, you are shown the original color version of *pepper.raw*. Your task is to experiment with the concept that multiple images of the same object which have been distorted by noise with the same characteristics may be used to restore the original.

Figure 1: pepper.raw

(a) Generating Noisy Images (50%)

Corrupt peppers.raw with mean-zero noise such that each time this function is given the same input image, the output is a different instance of noisy image. Since the image is grayscale with values ranging 0-255, you may construct noise in the range of -50 to +50 to provide a suitable level of distortion.

Describe the methods by which you developed your algorithm and any parameters you use to define the noise function.

(b) Averaging Images (50%)

Use the algorithms developed in part (a) and generate 10 and 100 noisy images of each type. Then use an averaging algorithm to estimate the values of the original pixels and reconstruct the input image. Discuss your results.

Thomas Trojan
1234-5678-90
trojan@usc.edu

EE 569 Homework #0
September 1, 2010

I. Abstract and Motivation

At times, as in the example of remote sensing, noise is introduced into images due to environmental conditions or an imperfect transmission channel between the camera and storage medium. In these cases, it is possible to reduce the noise either by means of filtering or by taking advantage of noise averaging. Depending on the nature of the noise, if the noise is of the same nature during each image recording, multiple images may be averaged to reduce or completely eliminate the artifacts present in them. This method uses the Law of Large Numbers, which states that given random variable X with finite mean μ:

$$S_N = \frac{1}{N}\sum_{i=0}^{N} X_i \xrightarrow[N\to\infty]{} \mu$$

In our case, we can obtain a series of images G_i which are the result of the original un-corrupted image F and additive noise. Using the law of large numbers and assuming that noise is zero mean, averaging of the images will result in:

$$\frac{1}{N}\sum_{i=0}^{N} G_i = \frac{1}{N}\sum_{i=0}^{N} F + N_i = \frac{1}{N}\sum_{i=0}^{N} F + \frac{1}{N}\sum_{i=0}^{N} N_i = F + \mu \approx F$$

II. Approach and Procedures

For the first part of this problem, the key first step towards a solution is generating the corrupted images, which requires creation of uniform noise. Luckily, both MATLAB and C++ have a built in function which generates pseudo-random numbers uniformly distributed between 0 and 1. In order to create the required uniformly distributed noise between -50 and 50, we simply apply the following transformation

Noise = [rand() * 100] – 50

Where rand() is the pseudorandom number generator. In order to get unique results for random numbers every time the program is run, we can seed, or initialize, the generator with the time() function, which returns the number of seconds since January 1st, 1970.

Next, we must generate 10 and 100 images which are the combination of pepper.raw with the uniform noise for each pixel in the image, taking care to clip the signal at 0 and 255 where it may potentially exceed those values:

$$G[i,j] = \begin{cases} 0 & \left(F[i,j] + rand()\right) < 0 \\ 255 & \left(F[i,j] + rand()\right) > 255 \\ F[i,j] + rand() & else \end{cases}$$

We then generate a resultant image based on the averaging of the respective pixel values from each of the 10 and 100 noisy images. Finally, we reconstruct the image based on low-pass filtering of one of the corrupted images in order to compare the results.

III.Experimental Results

Shown below are the results for Problem 0:

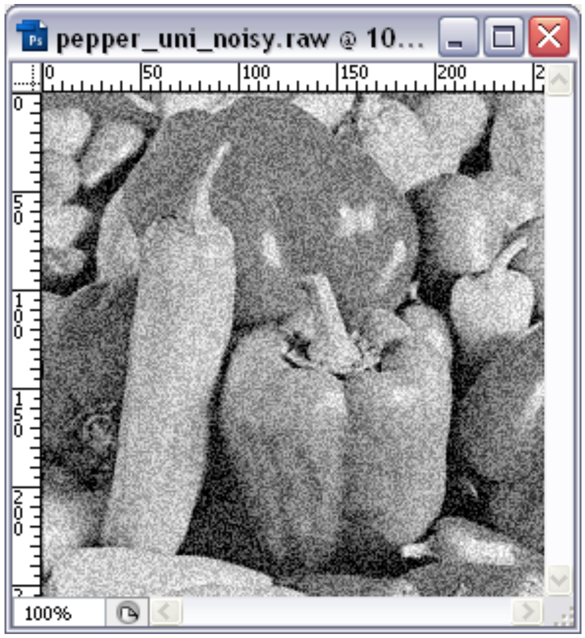

Figure 1: Noisy version of pepper.raw

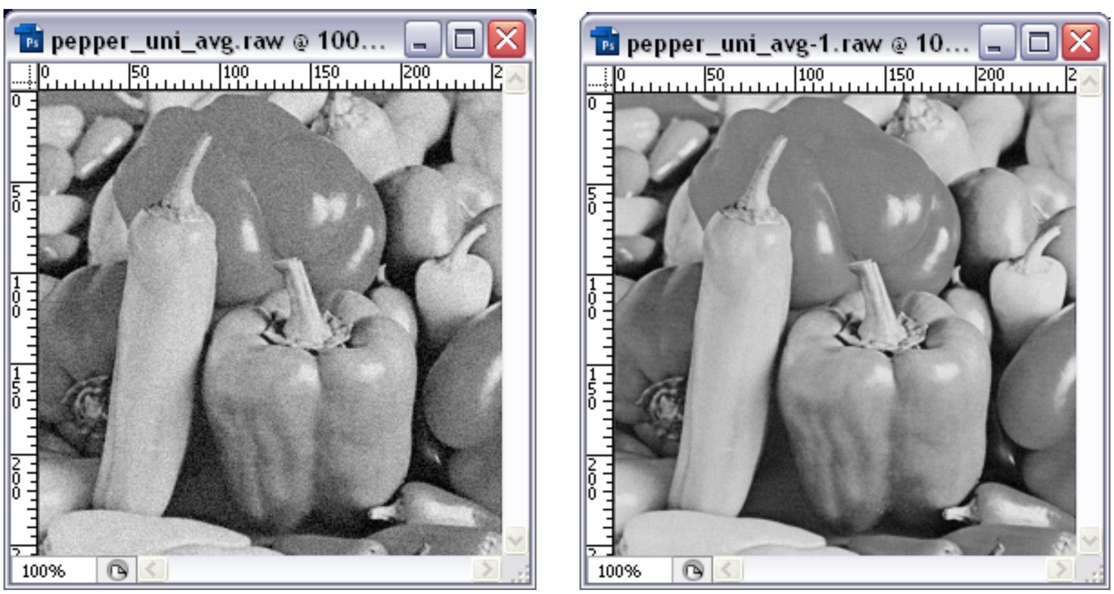

Figure 2: Averaged pepper.raw of 10 (left) and 100 (right) noisy images

IV. Discussion

Figure 1 above shows a sample of an image corrupted by the given uniformly distributed noise and impulse noise. Comparing this to the original, it is clear that there are ample distortions in an image corrupted with noise whose maximum value is just 20% of the full range of the image.

Figure 2 shows the comparison of the input image in Figure 1 to an averaged image with 10 and 100 iterations. At 10 iterations there is already ample improvement over the noisy image, but an extension of 100 iterations was used to show that given “infinitely many” corrupted samples, the original image can be perfectly restored.

From the images in Figures 1 and 2, it is clear that the averaging process, even with just 10 images, provides an amazing improvement over standard image enhancement techniques. The image generated from 100 averages is almost identical to the original. In the real world, this means that applications like remote sensing or even satellite communications could benefit from more clear and accurate images if many samples are taken and averaged together, rather than simply relying on processing algorithms to be able to recover the original scene.

The high rate of improvement implies that even 2 or 3 images could be adequately used to greatly improve output quality, provided that the resources available permit such redundant transmissions.