Digital Image Processing Laboratory: Image Halftoning

November 30, 1998

1 Introduction

In this lab, we will cover a useful image processing technique called *halftoning*, which is the process of converting a gray scale image into a binary image. The process of halftoning is required in many present day electronic applications such as facsimile (FAX), electronic scanning and copying, laser printing, and low bandwidth remote sensing. This lab will cover the halftoning techniques known as *ordered dithering* and *error diffusion*. All exercises will be performed in Matlab.

2 Binary Images

An 8 bit monochrome image allows 256 distinct gray levels. Such images can be displayed on a computer monitor if the hardware supports the required number intensity levels. However, some output devices print or display images with much fewer gray levels. In this case, the gray scale images must be converted to binary images, where pixels are only black or white.

The simplest way of converting to a binary image is based on *thresholding*, i.e. two-level (one-bit) quantization. Let $f(i,j)$ be a gray scale image, and $b(i,j)$ be the corresponding binary image based on thresholding. For a given threshold T, the binary image is computed as the following:

$$b(i,j) = \begin{cases} 255 & \text{if } f(i,j) > T \\ 0 & \text{else} \end{cases} \tag{1}$$

Figure 1 shows an example of conversion to a binary image via thresholding, using $T = 127$.

It can be seen in Figure 1 that the binary image is not "shaded" properly–an artifact known as *false contouring*. False contouring occurs when quantizing at low bit rates, such as one-bit, because the quantization error is dependent upon the input signal. Therefore if one reduces the signal dependence on the quantization error, the visual quality of the binary image will be enhanced.

One method of reducing the signal dependence on the quantization error is to add uniformly distributed white noise to the input image prior to quantization. To each pixel of the gray scale image $f(i,j)$, a white random number n in the range $[-A, A]$ is added, and then the resultant image is quantized by a one-bit quantizer, as in equation (1). The result of this method is illustrated in Figure 2, where the additive noise is uniform over $[-128, 128]$.

Questions or comments concerning this laboratory should be directed to Prof. Charles A. Bouman, School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907; (765) 494-0340; bouman@ecn.purdue.edu

(a) (b)

Figure 1: (a) Original gray scale image. (b) Binary image produced by simple fixed thresholding.

Notice that even though the resulting binary image is somewhat noisy, the false contouring has been dramatically reduced.

Figure 2: Random noise binarization.

2.1 Exercise

Down load the image *house.tif* from the lab home page. Use *xv* to print a hard copy of this image.

Apply simple thresholding to the image using $T = 127$. Display and print out the result. Label the hard copy with the applied method.

Compute the mean square error (MSE), which is defined by

$$MSE = \frac{1}{NM} \sum_{i,j} \{f(i,j) - b(i,j)\}^2 \tag{2}$$

where NM is the total number of pixels in each image. Note the MSE on the printout of the quantized image.

Section 2.1 Report:
Hand in the original image and the result of thresholding.

3 Ordered Dithering

Halftone images are binary images that appear to have a gray scale rendition. Although the random thresholding technique described in section 2 can be used to produce a halftone image, it is not often used in real applications since it yields very noisy results. In this section, we will describe another halftoning technique known as *ordered dithering*.

The human visual system tends to average a region around a pixel instead of treating each pixel individually, thus it is possible to create the illusion of many gray levels in a binary image, even though there are actually only two gray levels. With 2×2 binary pixel grids, we can represent 5 different "effective" intensity levels, as illustrated in figure 3. Similarly for 3×3 grids, we can represent 10 distinct gray levels. When dithering, we replace blocks of the original image with these types of binary grid patterns.

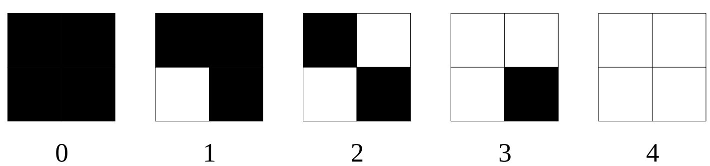

Figure 3: Five different patterns of 2×2 binary pixel grids.

Remember from section 2 that false contouring artifacts can be reduced if we can reduce the signal dependence on the quantization error. We showed that adding uniform noise to the monochrome image can be used to achieve this decorrelation. An alternative method would be to use a variable threshold for the quantization process.

Ordered dithering consists of comparing blocks of the original image to a 2-D grid, known as a *dither pattern*. Each element of the block is then quantized using the corresponding value in the dither pattern as a threshold. The values in the dither matrix are fixed, but are typically different from each other. Because the threshold changes between adjacent pixels, some decorrelation from the quantization error is achieved, which has the effect of reducing false contouring artifacts.

The following is an example of a 2×2 dither matrix,

$$T(i,j) = 255 * \begin{bmatrix} 5/8 & 3/8 \\ 1/8 & 7/8 \end{bmatrix} \tag{3}$$

This is a part of a general class of optimum dither patterns known as *Bayer matrices*. The values of the threshold matrix $T(i,j)$ are determined by the order that pixels turn "ON". This order can be put in the form of an *index matrix*. For example, the index matrix for a

Bayer matrix of size 2 is given by

$$I(i,j) = \begin{bmatrix} 3 & 2 \\ 1 & 4 \end{bmatrix} \tag{4}$$

and the relation between $T(i,j)$ and $I(i,j)$ is given by

$$T(i,j) = 255 \frac{I(i,j) - 0.5}{N^2} \tag{5}$$

where N^2 is the total number of elements in the matrix.

Figure 4 shows the halftone image produced by Bayer dithering of size 4. It is clear from the figure that the halftone image provides good detail rendition. However the inherent square grid patterns are visible in the halftone image.

Figure 4: The halftone image produced by Bayer dithering of size 4.

3.1 Exercise

Create a 4×4 Bayer matrix using the index matrix and equation (5). The index matrix for this dither pattern is given by

$$I(i,j) = \begin{bmatrix} 12 & 8 & 10 & 6 \\ 4 & 16 & 2 & 14 \\ 9 & 5 & 11 & 7 \\ 1 & 13 & 3 & 15 \end{bmatrix} . \tag{6}$$

Apply Bayer dithering to *house.tif* using the 4×4 dither pattern. Display and print out the result. Label the hard copy with the applied method.

Compute the mean square error using equation (2). Note the MSE on the printout of the quantized image.

Since the human visual system tends to smooth the halftoned image, compute the mean square error between the original and a filtered version the binary image. Use a 5×5

Gaussian filter defined by the following point spread function

$$h(i,j) = C\ \exp(-\frac{i^2+j^2}{2}) \tag{7}$$

where C is a normalizing constant such that $\sum_{i,j} h(i,j) = 1$. Note the weighted MSE on the result.

Section 3.1 Report:
Hand in the result of Bayer dithering.

4 Error Diffusion

Another method for halftoning is random dithering by *error diffusion.* In this case, the pixels are quantized in a specific order (raster ordering[1] is commonly used), and the residual quantization error for the current pixel is propagated (diffused) forward to unquantized pixels. This keeps the overall intensity of the output binary image closer to the input gray scale intensity.

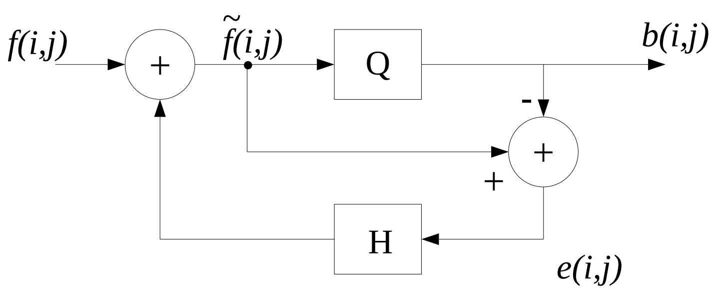

Figure 5: Block diagram of the error diffusion method.

Figure 5 is a block diagram that illustrates the method of error diffusion. The current input pixel $f(i,j)$ is modified by means of past quantization errors to give a modified input $\tilde{f}(i,j)$. This pixel is then quantized to a binary value by Q, using some threshold T. The error $e(i,j)$ is defined as

$$e(i,j) = \tilde{f}(i,j) - b(i,j) \tag{8}$$

where $b(i,j)$ is the quantized binary image.

The error $e(i,j)$ of quantizing the current pixel is is diffused to "future" pixels by means of a two-dimensional weighting filter $h(i,j)$, known as the *diffusion filter*. The process of modifying an input pixel by past errors can be represented by the following recursive relationship.

$$\tilde{f}(i,j) = f(i,j) + \sum_{k,l \in S} h(k,l) e(i-k, j-l) \tag{9}$$

[1]Raster ordering of an image orients the pixels from left to right, and then top to bottom. This is similar to the order that a CRT scans the electron beam across the screen.

The most popular error diffusion method, proposed by Floyd and Steinberg, uses the diffusion filter shown in Figure 6. Since the filter coefficients sum to one, the local average value of the quantized image is equal to the local average gray scale value. Figure 7 shows the halftone image produced by Floyd and Steinberg error diffusion. Compared to the ordered dither halftoning, the error diffusion method can be seen to have better contrast performance. However, it can be seen in Figure 7 that error diffusion tends to create "streaking" artifacts, known as *worm* patterns.

	•	7/16
3/16	5/16	1/16

Figure 6: Point Spread Function of the error diffusion filter proposed by Floyd and Steinberg.

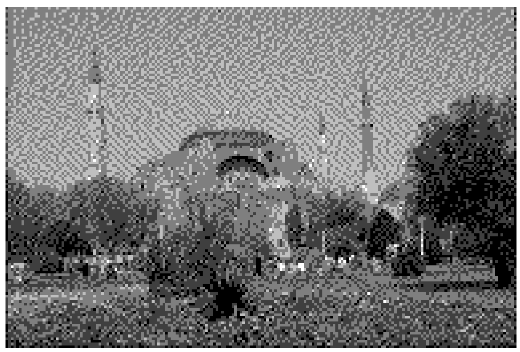

Figure 7: A halftone image produced by Floyd and Steinberg error diffusion method.

4.1 Exercise

Apply the error diffusion technique to *house.tif*. Use a threshold $T = 127$ and the diffusion filter in Figure (6). It is most straightforward implement this by performing the following steps on each pixel in raster order:

1. Initialize an output image matrix with zeros.

2. Quantize the current pixel using using the threshold T, and place the result in the output matrix.

3. Compute the quantization error by subtracting the binary pixel from the gray scale pixel.

4. Add scaled versions of this error to "future" pixels of the original image, as depicted by the diffusion filter of Figure 6.

5. Move on to the next pixel.

Display and print out the result. Label the hard copy with the applied method.

Compute the mean square error using equation (2). Note the MSE on the printout of the quantized image.

Compute the weighted mean square error using the same technique described in section 3.1. Note the weighted MSE on the result.

Section 4.1 Report: Hand in the result of error diffusion.