INVERSE HALFTONING OF SCANNED IMAGES

Tsi-Yi Chao and Hsueh-Ming Hang

Dept. of Electronics Eng., National Chiao-Tung Univ.
Hsin-Chu 300, Taiwan, Republic of China
Email: hmhang@cc.nctu.edu.tw

ABSTRACT

Our goal in this research is to find a good inverse halftoning algorithm that recovers gray-scale images from the scanned images. To this aim, we develop the printer and the scanner models and two types of reconstruction methods. In the first method, the reconstruction filter is derived directly from the scanned data and the ideal original gray-scale image. In the second method, the forward halftoning, the printing and the scanning processes are reversed one by one. Either approach seems to produce reasonably good results.

1. INTRODUCTION

Halftoning is a very popular technique for producing printed pictures with only two levels of color. Our goal in this study is to reconstruct gray-level images from the scanned bi-level images. It is rather different from the usual inverse halftoning designed for ideal digital bi-level halftone images [1] because the scanner output images are distorted versions of the original digital bi-level images. In order to reverse the operations done by the printer and the scanner, we develop the printer and the scanner models to represent the printing and the scanning processes. Then, when we apply the inverse operations of the halftoning, the printing and the scanning processes on a printed image, the gray-level images can thereafter be recovered.

2. PRINTER AND SCANNER MODELS

From signal processing viewpoint, a printer is a process that converts a halftone image (input) to a printed image (output). Similarly, a scanner is a signal processing process that converts a printed image (input) to a scanned image (output). In the following derivation and experiments, we assume the resolutions of both printer and scanner are the same (300 dpi). Following the same approach, we could extend these results to unequal resolution cases.

2.1 Printer Model

The printer and scanner block diagram is shown in Fig. 1, where $d(i,j)$ represents the ideal halftone image and has only two values, say 0 and 255, $d'(i,j)$ represents the scanned gray-level image, and $D(i,j)$ is the representative value of ink distribution inside the $(i,j)^{th}$ grid assuming the paper surface has been partitioned into invisible perfect grids. $D(i,j)$ has two values, w and b, representing the *white* and the *black* dots. Thus the printer model is simply a transformation that converts the digital bi-level data into dots of w and b values on paper.

$$D(i,j) \equiv \begin{cases} b & \text{if } d(i,j) = 0 \\ w & \text{if } d(i,j) = 255. \end{cases} \quad (1)$$

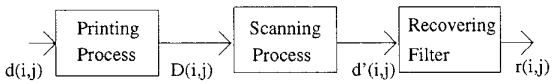

Figure 1: The printing, scanning and recovering system diagram

There are inherit "uncertainties" or "noises" in our printer model. They come from many different sources. For example, the ink density distribution and the dot size of the same printer may vary from one location to another. Similarly, there exists fluorescence variation on the same piece of paper. On the other hand, in scanning, the scanner grid is unlikely aligned perfectly with the printed image grid. Hence, there are grids "rotation" and "displacement" involved in picking up images by the scanner. We are able to compensate part of rotation and displacement when they are large. However, the remaining small residual values would be treated as *noise*.

2.2 Scanner Model

We assume that the scanner model can be decomposed into two parts: Part 1 is a linear system and Part 2 is a memoryless nonlinearity. Part 1 models the transformation from printed image dot ($D(i,j)$) to the brightness received by the scanner sensor. We assume that the brightness received by a sensor from any dot on the paper depends only on the relative distance between the dot and the sensor, and that the brightness generated by all the dots can be linearly superposed. Under the above assumptions, this transformation is represented by a linear filter with impulse response ($h(i,j)$) which behaves like a lowpass filter (Fig. 2). Part 2 models the transformation from the sensor received brightness to its output value. Based on our observations we assume that it behaves like a (nonlinear) clamping function that limit the scanner outputs to a finite range as shown in Fig. 2.

This work was in part supported by a grant from Opto-Electronics and Systems Laboratories, ITRI, Taiwan, ROC.

0-8186-7310-9/95 $4.00 © 1995 IEEE

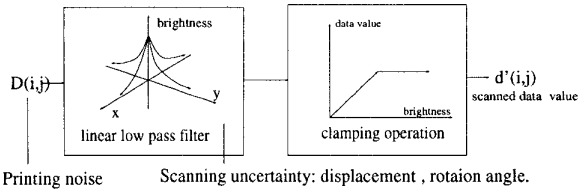

Figure 2: A scanner model with noise

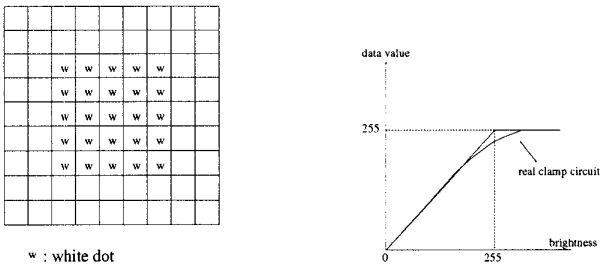

Figure 3: (a) The scanned 9*9 pattern, (b) The clamping operation of Part2

The concatenated printer and scanner model has been shown in Fig. 1. Ideally, we like to compute the parameters in the above printer and scanner models, namely, w and b in the printer model, the impulse response $h(i,j)$ and the shape of clamping function in the scanner model. However, in our experiment setup, we can only measure the data at the printer input, $d(i,j)$ and at the scanner output, $d'(i,j)$. Therefore, we do not have sufficient information to determine the values of w and b directly; we could only compute their scaled version, W and B, as follows.

Imagine that we create an artificial (physically non-exist) printed image with a white dot at position (0,0) and surrounded by the *pure black* background. (Note: "*Pure black* background" means the scanner output of this "color" is "0".) It is denoted as $D(i,j) = w \times \delta(i,j)$. Then, we denote the total sum of the filter outputs as W; that is, $W = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} wh(i,j)$. Let $\alpha = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} h(i,j)$, then $W = \alpha \times w$. Therefore, the W value represents the total brightness received by the scanner due to a white dot on paper. Similarly, for a black dot (b), $B = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} bh(i,j) = \alpha \times b$. Based on the above definitions of W and B, $D'(i,j)$ is defined as a scaled version of $D(i,j)$; namely, $D'(i,j) = D(i,j) \times \alpha$. Thus, $D'(i,j)$ has values of B and W, which can be computed from the scanned values of a known printed pattern as discussed in the next subsection.

We next look for $h(i,j)$. Assuming the clamping operation in the scanner model (Fig. 2) is inactive (all the scanned values are within the linear region), then the operation of the entire scanner is simply a linear filter. That is, $d'(i,j) = h(i,j) \otimes D(i,j)$. Let

$$h'(i,j) = \frac{h(i,j)}{\alpha}, \quad \text{then} \quad d'(i,j) = h'(i,j) \otimes D'(i,j).$$

Hence, $h'(i,j)$, the normalized version of $h(i,j)$, will be used in conjunction with $D'(i,j)$ in the rest of this paper. Now, this $h'(i,j)$ can be estimated from the measured data of known input signals using minimum mean square estimation (MMSE).

2.3 Simulation Results

A known 5×5 printed white pattern shown in Fig. 3(a) is used to identify the printer and the scanner models described in the above.

1. Find B value: Scan a 512×512 picture and use the 400×400 black area inside to compute an average value of $B = 20$.

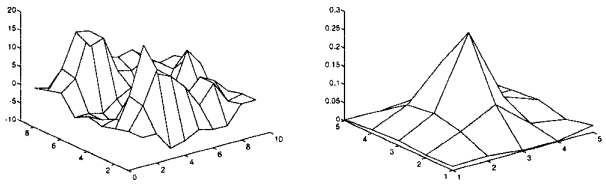

Figure 4: (a) The error pattern: $\hat{d}'(i,j) - d'(i,j)$, (b) The filter $h'(i,j)$

2. Find W value: The total sum of a 13×13 area surrounding the 5×5 pattern is $SUM = 3197404$. This SUM is the overall brightness of 25 white dots and $(13^2 - 5^2)$ black dots. Hence, $W = \frac{3197404 - (169-25)\times 20}{25} = 240$.
3. Estimate $h'()$: Use the standard minimum mean square error (MMSE) method to estimate $h'(i,j)$, $-2 \le i,j \le 2$ based upon the 9×9 region surrounding the 5×5 white dots. The resultant $h'(i,j)$, and the identification error $\varepsilon(i,j) = \hat{d}'(i,j) - d'(i,j)$ are shown in Fig. 4, where $\hat{d}'(.,.)$ is the estimated value of $d'(.,.)$.

3. FURTHER INVESTIGATION ON PRINTER AND SCANNER MODELS

In addition to the scanned pattern shown in Figure 3(a) (Pattern A), we create a "7×7" data pattern shown in Figure 5(a) (Pattern B) and a "four concatenated 3×3" pattern shown in Figure 5(b) (Pattern C). The Pattern B is created to produce a large white area and thus the clamping effect may become evident, assuming that the clamping operation is the nonlinear curve shown in Figure 3(b). In contrast, the Pattern C is created to avoid clamping operation. The above three patterns are used to identify the scanner model using either the "MMSE" method or the "sequential training" method.

The sequential training method we use is the conventional least mean square (LMS) method. Its training procedure is similar to the SWF method stated in [1]. Because the training data size is small, it takes a number of iterations

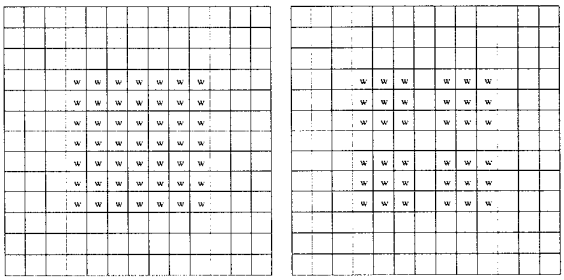

Figure 5: (a) Pattern B, (b) Pattern C

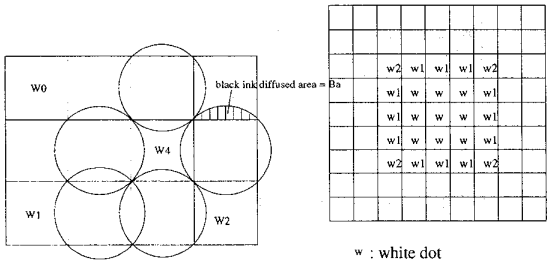

Figure 6: (a) Black ink diffusion (b) Pattern A1

Ba	A1 pattern	B1 pattern	C1 pattern
0.00	30.2381	97.9906	30.1768
0.09	22.9688	62.9885	24.0144
0.18	29.5860	41.0079	19.5240
0.21	35.6747	37.5706	18.4498
0.24	44.1206	36.5753	17.6650
0.27	55.3422	38.4525	17.3353
0.30	69.9794	43.7021	18.1123

Table 1: Ba vs MSE for Patterns A1, B1 and C1

Figure 7: (a) Lenna1 (b) The image recovered from lenna1 by DIM method

(40000 iterations) to reach convergence. Both methods produce similar results. Typically, the identification errors are significantly larger on the edges of white block in the patterns. We suspect that the black dot ink diffuses to the adjacent dots and thus the brightness generated by the white dots is less than expect. This phenomenon is illustrated by Figure 6(a). We assume that the amount of black ink of a black dot diffuses into a nearby white dot is Ba. Then there are five types of white dots ($W0, W1, W2, W3, W4$) in our test patterns, where $Wk = W - k \times Ba$ approximately [2]. Based on the above definitions of Wk and Ba, we can use the Wk to substitute for the W in the Patterns A, B and C to obtain the corresponding corrected Patterns $A1$, $B1$and $C1$ as illustrated by Figure 6(b). Now, we try different Ba values (Ba = 0, 0.03, 0.06, ...0.3) to find the scanner model using the same sensed data but based on assumption of Patterns $A1$, $B1$, and $C1$. From Table 1, we observe that for Pattern $A1$ the MSE of $Ba = 0.09$ is the smallest. In the cases of Patterns $B1$ and $C1$, the smallest MSE occurs at $Ba = 0.24$ and $Ba = 0.27$ respectively. In addition to the above results, we find that the filter shape are very similar when $Ba = 0.18$ for Patterns $A1$, $B1$, $C1$ patterns. Hence, we choose $Ba = 0.18$ to identify the scanner model. Because that the MSE of Pattern $C1$ is the smallest, the scanner filter derived based on Pattern $C1$ assumption may be the most accurate.

4. INVERSE HALFTONING SCHEMES

Our goal is to reconstruct gray-level images from the scanned images. The entire printing, scanning, and reconstructing system is illustrated by Figure 1, where $r(i,j)$ represents the reconstructed gray-level images. There are two approaches to use the printer and the scanner models we just described. 1) When we inverse the printing and scanning processes for a printed image, we can recover the halftone images. Then, the ordinary inverse halftoning technique designed for ideal bi-level images can be applied. This will be called *scanner model based (SMB)* method. Or 2) we could combine the inverse printer and scanner model together with the usual inverse halftoning procedure to produce the gray-level images in one step. This will be called *direct inverse modeling (DIM)* method .

In this paper, lenna, peppers and baboon of size 512 × 512, 8 bits each pixel are used as the test pictures and the processing window size of the scanned data is 718 × 716. These pictures are halftoned using a 5 × 5 clustered-dot dither matrix and are printed using an NEC Printer (Model: NT3234). The term "lenna" denotes the original gray-scale lenna image. The first scanned lenna data is called "lenna1". The same notation is used for baboon and peppers. The scanned image (lenna1) is shown in Figures 7(a). It can be seen that the scanned image is rather blocking.

4.1 Direct Inverse Modeling Method

The direct inverse modeling (DIM) method is simply deriving the reconstruction filter based on the scanned data (input) and the original images (ideal output). But this

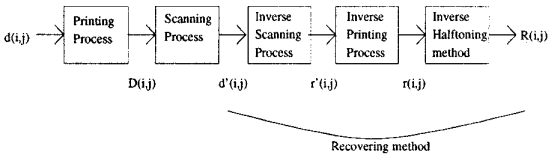

Figure 8: The scanner model based recovering system diagram

method can be used only when the dimensions of the input and the output are identical. Hence, we need to convert the expanded (or reduced) scanned image to the original size. In our experiment, a 512 × 512 original image is expanded to a 519 × 515 scanned image. A simple approach is to interpolate and then trim the scanned image. If we ignore the interpolation operation, the trimming operation alone would produce incorrect reconstruction filters. The recovered image from lenna1 is shown in Figure 7(b). We can observe that the recovered images is somewhat blurred. The low frequency components of the original image seem to be recovered well but the high frequency components are lost.

4.2 Scanner Model Based Method

In this section, the inverse halftoning method based on the scanner model (Scanner Model Based (SMB) Method) is introduced. The system diagram of converting the halftoning images to gray-scale images is shown in Figure 8. The complete inverse halftoning system includes the inverse scanning process, the inverse printing process and the inverse halftoning process. The inverse scanner model is composed of the inverse clamping operation and the inverse scanner linear filter. The inverse scanner filter is found by finding the inverse model of the scanner described earlier. The printing process maps each pixel value from (255, 0) to (W, B). Hence, the inverse printing process should maps each pixel value from (W, B) to (255, 0). However, because that the inverse scanned image $r'(i, j)$ has more than two values due to various noises (nonideal factors) in the printing and the scanning process, the inverse mapping becomes less intuitive. Two approaches have been tested.

(A) Scaling Approach

Because the B/W is small. Hence, we simply treat (W, B) as $(W, 0)$. As a consequence of this mapping is that the resultant filter is a scaled (larger) version of the ideally inverted filter. However, this method can compensate for the brightness loss due to black ink diffusion. Therefore, it produces rather good result. These new filters (scaled up) are named lennasa, pepperssa, and baboonsa.

(B) Combined Approach

The reconstruction filter is trained by using (W, B) as input (where $W = 240$ and $B = 20$) and the perfect original image as output. This filter produce higher MSE. The filters obtained by "Combined Approach" are named lennaca, peppersca and baboonca. The gray-scale images recovered from lenna1 using lennasa and lennaca are shown in Figures 9(a) and 9(b). We can find that there are visible noises in the smooth regions. Because the inverse scanner linear filter is calculated from the "designed patterns", this filter can recover the high frequency portion well but can not eliminate the displacement noise. The modified inverse halftoning filter is a low-pass filter and can reduce some displacement noises. Hence, the "SMR" method can recover images with higher frequencies at the price of visible displacement noise in recovered images. From Table 2, we can observe that lennasa filter is better than lennaca filter and DIM method.

Figure 9: (a) The image recovered from lenna1 using lennasa filter (b) The image recovered from lenna1 using lennaca filter

method		DIM	SMB		
			lennaca	lennasa	baboonsa
lenna1	MSE	427.87	374.28	298.38	
	AVE	112.39	113.87	123.43	
peppers1	MSE	632.28	640.85	615.88	648.83
	AVE	112.43	113.69	123.12	123.73
baboon1	MSE	1073.35	1054.88	856.88	883.33
	AVE	112.42	113.18	122.70	120.34
	AVE of lenna =123.54				
	AVE of peppers =120.22				
	AVE of baboon =129.62				

Table 2: Comparison of various reconstruction methods

5. REFERENCES

[1] L.-M. Chen and H.-M. Hang, "Inverse halftoning for monochrome pictures," *IEEE International Conf. on Image Processing '94*, pp.1022–1026, Austin, Texas, Nov. 1994.

[2] T.N. Pappas and D.L.Neuhoff,"Printer Models and Error Diffusion," *IEEE Trans. on Image Proc., Vol. 4, No. 1, Jan. 1995*