Preserving Features in Multilevel Halftones

Lai-Yan Wong and Yuk-Hee Chan
Center of Multimedia Signal Processing
The Hong Kong Polytechnic University, Hong Kong

***Abstract* — Conventional threshold decomposition (TD) based multilevel halftoning algorithms decompose an input image into layers, halftone them sequentially with a binary halftoning algorithm, and combine their binary halftones to produce the final multilevel halftone. When these algorithms are exploited to produce multilevel halftones, bright spatial features are generally difficult to preserve as darker pixels in the final multilevel output are positioned first. We propose a solution to solve this problem in this paper. Simulation result shows that the proposed method can provide an output of better quality as compared with conventional TD-based algorithms.**

I. INTRODUCTION

The capability of printing devices has been significantly enhanced nowadays such that they can now produce outputs of more than two intensity levels. Accordingly, multilevel halftoning technique is needed to convert an input image to a multilevel output image for being printed. Threshold decomposition is a technique proposed in [1] and is widely used in multilevel halftoning. This technique separates an input image nonlinearly into several energy planes, halftones them sequentially with a binary halftoning algorithm under a constraint, and finally combines the binary halftoning results to get a multilevel output.

To distribute ink dots homogenously in the intermediate-tone regions of an image so as to eliminate the banding artifacts, the decomposed layers are processed from the layer of the highest energy to the layer of the lowest energy one by one subject to a stacking constraint [1]. Specifically, when a location in a higher energy layer is assigned a black pixel in its binary output, black pixels must also be assigned to the corresponding locations in the binary outputs of the lower energy layers. This stacking constraint confines the dot assignment in lower energy layers. The lower the energy of a layer, the more restricted the assignment of a white dot is. As the final multilevel output is a sum of the binary halftoning results of all layers, the locations of brighter dots in the final multilevel result are more restricted than the locations of darker dots. The bright features in an image are difficult to preserve because we may not be able to put the bright dots at the right positions.

In this paper, we propose a multilevel halftoning algorithm to improve the output quality of multilevel halftoning by eliminating this bias.

II. THRESHOLD DECOMPOSITION

This section presents the conventional approach used to handle the layers in threshold decomposition based multilevel halftoning and its weaknesses.

Consider the case that we want to convert a continuous-tone gray–level image $\boldsymbol{A}$ to a multilevel image $\boldsymbol{B}$. The pixel values of both $\boldsymbol{A}$ and $\boldsymbol{B}$ are bounded in [0,1], where 0 and 1 denote the minimum (black) and the maximum (white) intensity levels respectively. Furthermore, the pixel values of $\boldsymbol{B}$ are confined to be a member of $\{n/(m\text{-}1)|n=0,\ldots,m\text{-}1\}$, where m is the number of available output intensity levels. Without losing the generality, we assume that the size of images $\boldsymbol{A}$ and $\boldsymbol{B}$ is $N\times N$, where $N=2^k$ and k is a positive integer. For reference, $\boldsymbol{I}(x,y)$ denotes the pixel value of image $\boldsymbol{I}$ at position (x,y).

When threshold decomposition is applied, the input image $\boldsymbol{A}$ is decomposed into m-1 images (also referred to as layers), each of which is denoted as $\boldsymbol{A_d}$ for d = 1…m-1, such that we have

$$A(x,y)=\sum_{d=1}^{m-1}A_d(x,y)/(m-1) \quad \text{for } x,y=0,1,\ldots,2^k\text{-}1 \qquad (1)$$

As suggested by Suetake [1], A_d can be determined as

$$A_d(x,y)=A_{d-1}(x,y)-A(x,y)^{d-1}\times\left(\frac{(m-1)!}{(d-1)!(m-d)!}\right)(1-A(x,y))^{m-d}$$

for d =1,2, …, m-1, where $A_0(x,y)$ = 1 for all (x,y) (2)

In this paper, a layer of smaller index value is referred to as a higher energy layer as it carries more energy. In contrast, a layer of larger index value carries less energy and is referred to as a lower energy layer.

After threshold decomposition, the layers are halftoned with a binary halftoning algorithm one by one and the final multilevel halftone is obtained by summing up the binary halftones of all layers. Figure 1 shows how a 3-level halftone is produced with this conventional approach.

As layers are possessed to produce binary halftones one by one subject to a stacking constraint, the positions of the black dots in the final multilevel halftone are determined first in the conventional approach. This bias makes conventional algorithms difficult to preserve bright features in the image and hence lowers the output quality. In order to remove this bias, we suggest handling the layer of the maximum energy and the layer of the minimum energy simultaneously to locate the brightest and the darkest dots in the final multilevel halftone in an interleaving manner.

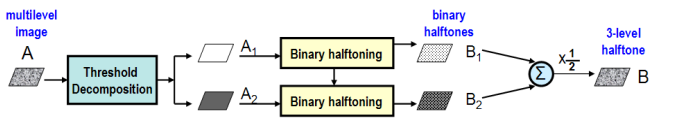

Fig. 1 Flow of a conventional multilevel halftoning algorithm based on threshold decomposition

III. PROPOSED ALGORITHM

In theory, one can apply any binary halftoning algorithm to halftone a layer after threshold decomposition. In our proposed algorithm, feature-persevering multiscale error diffusion (FMED)[2] and its modified version are exploited to halftone the layers since FMED is more flexible to introduce dots of different nature to the outputs of individual layers and it is superior to other conventional binary halftoning algorithms in preserving spatial features.

FMED is a two-step iterative method developed based on MED[3]. In the first step it searches for the most critical pixel location to assign a minority dot in a region and in the second step it diffuses the quantized error of the selected pixel to its neighbors. We adopt the general idea of FMED and modify it to halftone the layers.

Figure 2 shows the operation flow of the proposed algorithm for producing a 3-level output. Instead of processing layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ one by one sequentially, the proposed algorithm processes the two layers simultaneously to locate the darkest and the brightest pixels in the final multilevel halftone in an interleaving manner.

At the beginning of the halftoning process, we estimate the black dot budget for the higher energy layer $\boldsymbol{A_1}$ and the white dot budget for the lower energy layer $\boldsymbol{A_2}$ as follows.

$$Budget0 = N_o - \sum_{(x,y)} A_1(x,y)$$

$$Budget1 = \sum_{(x,y)} A_2(x,y) \tag{3}$$

where $N_o = N \times N$ is the total number of pixels in the image. Accordingly, a budget ratio is determined as

$$Budget\ \ Ratio = \frac{\sum_{(x,y)} A_2(x,y)}{N_o - \sum_{(x,y)} A_1(x,y)} \tag{4}$$

Then we alternately locate a pixel in layer $\boldsymbol{A_1}$ to assign a '0' and locate a pixel in layer $\boldsymbol{A_2}$ to assign a '1' under the guidance of the budget ratio. Under the stacking constraint, this is equivalent to alternately positioning the darkest and the brightest dots in the final multilevel halftoning output of the input image. By doing so, the bias in either white or black dots is removed and a balance on preserving the dark and bright spatial features in an image can be achieved.

When dots are put in the layers, the black dot budget for layer $\boldsymbol{A_1}$ and the white dot budget for layer $\boldsymbol{A_2}$ are consumed accordingly. At the time when we determine which type of dots should be position next, we pick the one which can make the ratio of the remained white dot budget to the remained black dot budget closer to the budget ratio.

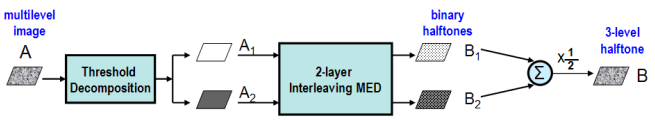

Fig. 2 Flow of the proposed algorithm for producing a 3-level halftone

The search of the pixel to put a dot is based on an energy plane $\boldsymbol{E}$. When locating a pixel to put a black dot, the energy plane $\boldsymbol{E}$ is initialized to be layer $\boldsymbol{A_1}$. Otherwise it is initialized to be layer $\boldsymbol{A_2}$. The search is carried out as follows. Starting with the selected energy plane $\boldsymbol{E}$ as the region of interest, we repeatedly divide the region of interest into nine overlapped sub-regions of equal size and select one of these sub-regions to be the new region of interest based on a selection criterion. When we are looking for a pixel in layers $\boldsymbol{A_1}$ to put a black dot, we follow the minimum intensity guidance and pick the sub-region having the smallest sum of its pixel values. Otherwise, we follow the maximum intensity guidance and pick the one of the largest sum. We repeat the above steps to update the region of interest until a pixel location is reached.

Without loss of generality, let $\boldsymbol{A_k}$ be the layer that we select to locate a pixel to put a dot and the selected location of the pixel is (p,q). According to the stacking constraint, when a pixel in the brighter layer ($\boldsymbol{A_1}$) is selected to put a black dot in its binary halftone, a black dot must also be assigned to the corresponding pixel in the binary halftone of the darker layer ($\boldsymbol{A_2}$) to maintain the consistency. Based on the same philosophy, when a pixel in the darker layer ($\boldsymbol{A_2}$) is selected to put a white dot in its binary halftone, a white dot must also be assigned to the corresponding pixel in the binary halftone of the brighter layer ($\boldsymbol{A_1}$). In other words, once pixel (p,q) is selected, no matter whether the search is based on layer $\boldsymbol{A_1}$ or $\boldsymbol{A_2}$, both layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ are affected and dots of identical intensity level are put at $\boldsymbol{B_1}(p,q)$ and $\boldsymbol{B_2}(p,q)$ simultaneously as follows.

$$B_1(p,q) = B_2(p,q) = \begin{cases} 1 & if\ \ searching\ is\ based\ on\ A_2 \\ 0 & if\ \ searching\ is\ based\ on\ A_1 \end{cases} \tag{5}$$

where $\boldsymbol{B_k}$ is the binary output plane of layer $\boldsymbol{A_k}$.

For each layer $\boldsymbol{A_i}$, where $i \in \{1,2\}$, the difference between $\boldsymbol{B_i}(p,q)$ and $\boldsymbol{A_i}(p,q)$ is then diffused to $\boldsymbol{A_i}(p,q)$'s neighbors to update layer $\boldsymbol{A_i}$ as follows.

$$A_i^{'}(x,y) = \begin{cases} 0 & if\ \ (x,y) = (p,q) \\ A_i\ (x,y) - w_{x-p,y-q} \cdot \left(B_i(p,q) - A_i\ (x,y)\right) \cdot \dfrac{R(x,y)}{S} & else \end{cases} \tag{6}$$

where $\boldsymbol{A'_i}(x,y)$ and $\boldsymbol{A_i}(x,y)$ are, respectively, the values of pixel (x,y) in layer i after and before the error diffusion process, $\boldsymbol{B_i}(p,q)$ is the value assigned to pixel (p,q), $R(x,y)$ is a mask defined as

$$R(x,y) = \begin{cases} 0 & if\ \ B_i(x,y)\ has\ been\ assigned\ a\ dot \\ 1 & else \end{cases} \tag{7},$$

$w_{s,t}$ for $(s,t) \in \Omega$ is a filter weight of a non-causal diffusion filter with support Ω, and

$$S = \sum_{(x-p,y-q)\in\Omega} \left(w_{x-p,y-q} \cdot R(x,y)\right) \tag{8}$$

The aforementioned procedures are repeated until we use up all black dot budget for the higher energy layer $\boldsymbol{A_1}$ and all white dot budget for the lower energy layer $\boldsymbol{A_2}$. Values of some pixels in $\boldsymbol{B_2}$ and $\boldsymbol{B_1}$ may be left unassigned at this point. They are backfilled as follows to complete the construction of the binary halftones of layers $\boldsymbol{A_2}$ and $\boldsymbol{A_1}$.

$$B_1(x,y)=1 \qquad \text{for all } (x,y) \text{ in } \boldsymbol{B_1} \text{ not assigned } 0 \tag{9}$$

and

$$B_2(x,y)=0 \qquad \text{for all } (x,y) \text{ in } \boldsymbol{B_2} \text{ not assigned } 1 \tag{10}$$

Finally, the multilevel halftone is obtained by combining the two binary halftones as follows.

$$B(x,y)=\frac{1}{2}\sum_{n=1,2} B_n(x,y) \tag{11}$$

Figure 3 summaries the operation flow of the proposed multilevel halftoning algorithm in a form of pseudo code.

IV. SIMULATION

Simulation was carried out to evaluate the performance of the proposed method with a set of fifteen 256 gray-level testing images of size 256×256 each. A 3×3 noncausal filter with filter coefficients w = [1 2 1; 2 0 2; 1 2 1] was exploited in the simulation. For comparison, the performance of two conventional threshold decomposition based multilevel halftoning algorithms ([1] and [4]) was also evaluated in the simulation.

Figure 4 shows a part of an original testing image "Barbara" and its 3-level halftoning results obtained with different evaluated algorithms. Figure 5 shows enlarged portions of Figure 4 for better comparison. One can see that the proposed algorithm can render the texture in a way better than [1] and [4].

The performance of the evaluated algorithms was also evaluated in terms of Universal Objective Image Quality Index (UQI) [5] and Mean Square Error through virtual human visual system (MSEv) [6] for objective comparison. Specifically, the averages of the UQI performances of the 3-level halftoning results obtained with [1], [4] and the proposed algorithm are 0.4330, 0.4608 and 0.4643 respectively. Note that UQI is used to measure the information loss after halftoning. Its value is bounded in the range of [-1,1] and a higher value means an output of higher quality.

Figure 6 shows the performance of various algorithms in terms of average MSEv. The MSEv performances of the simulation results are measured under the assumption that the simulation results are viewed at an observation distance which varies from 20 to 80 inches. MSEv measurement is sensitive to energy error which reflects the difference between the original image and its halftoning result. The higher the value of MSEv, the poorer the performance is. The proposed algorithm performs better than [1] and [4] in terms of both UQI and MSEv.

The complexity of FMED is proportional to the number of the pixels in an image and most of its effort is for searching locations to put dots. When FMED works with the conventional TD framework to produce a multilevel halftone, the total searching effort is roughly proportional to the number of layers. However, when it works with the proposed framework, the total number of pixel locations that we have to locate is bounded by the number of pixels in the image, which is independent of the number of layers. For the production of a 3-level halftone, a reduction ratio of 2 to 1 can be achieved.

```
Initialize Budget1 and Budget0 with eqns. (3)
Compute BudgetRatio = Budget1/Budget0;
WHILE Budget1+Budget0>0
   Determine if white or black dot should be put in this round
   IF white dot should be put
      Search for a pixel location in A2 via maximum intensity guidance
      B1(xo,yo) = B2(xo,yo) = 1     % Assume that the location is (xo,yo)
      Diffuse error of Ai(xo,yo) to its neighbors in Ai for i=1,2
      Budget1 = Budget1 – 1
   ELSE
      Search for a pixel location in A1 via minimum intensity guidance
      B1(xo,yo) = B2(xo,yo) = 0   % Assume that the location is (xo,yo)
      Diffuse error of Ai(xo,yo) to its neighbors in Ai for i=1,2
      Budget0 = Budget0 - 1
   END
END
B1(x,y) = 1      for all (x,y) in B1 not assigned 0
B2(x,y) = 0      for all (x,y) in B2 not assigned 1
Multilevel halftone is B(x,y) = 1/2 Σ_{n=1,2} Bn(x,y)
```

Fig. 3 Pseudo code of the proposed halftoning algorithm

V. CONCLUSION

Threshold decomposition is widely used in multilevel halftoning. When it is exploited, a given input image is decomposed into layers such that the layers can be processed with a binary halftoning algorithm and then combined to produce the final multilevel halftone. In conventional approaches, layers of higher energy are handled first. It implies that the pixels of lower intensity levels in the multilevel halftoning result are positioned first. Bright spatial features are difficult to preserve because of this bias. We propose an approach which changes the processing order of the layers such that the pixels of extreme intensity levels in the multilevel halftoning output are always positioned first. The bias is then automatically removed and the visual quality of the output can be improved. The suggested approach can be easily realized with FMED [2]. Simulation results showed that the proposed method can provide a better result than the conventional TD-based algorithms such as [1] and [4] in terms of both subjective and objective criteria.

ACKNOWLEDGMENT

This work was supported by a grant from The Hong Kong Polytechnic University (PolyU Grant G-YJ64).

REFERENCES

[1] N. Suetake, M. Sakano, E. Nakashima and E. Uchino, "Simple multilevel halftoningsxcelled in gradation reproducibility," *Optics Letters*, Vol. 33, No. 4, pp.339-341, 2008.

[2] Y. H. Chan and S. M. Cheung, 'Feature-preserving multiscale error diffusion for digital halftoning,' *Journal of Electronic Imaging*, Vol.13, Issue 3, 2004.

[3] Y. H. Chan, "A modified multiscale error diffusion technique for digital halftoning," *IEEE Signal Process. Lett.*, Vol.5, No.11, pp.277-280, 1998.

[4] Y. H. Fung and Y. H. Chan, "Multilevel Halftoning using Multiscale

Error Diffusion," *Journal of Electronic Imaging*, Vol.19, Issue 3, 2010

[5] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, Vol. 9, No. 3, pp. 81–84, 2002.

[6] D. L. Lau and G. R. Arce, *Modern Digital Halftoning*. New York:Marcel Dekker, 2008.

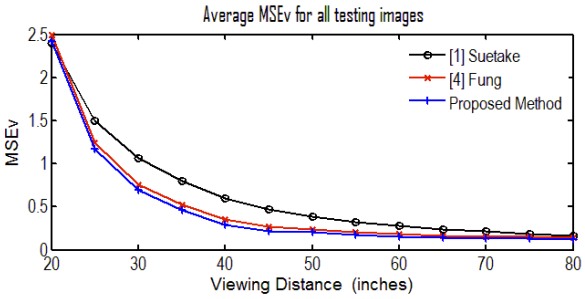

Fig. 6 Average MSEv performance at various viewing distances

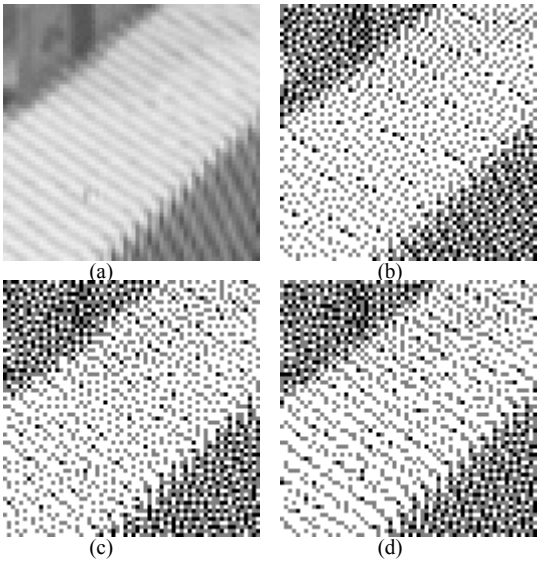

Fig. 5 Regions clipped from Fig. 4: (a) Original, (b) [1], (c) [4] and (d) the proposed algorithm

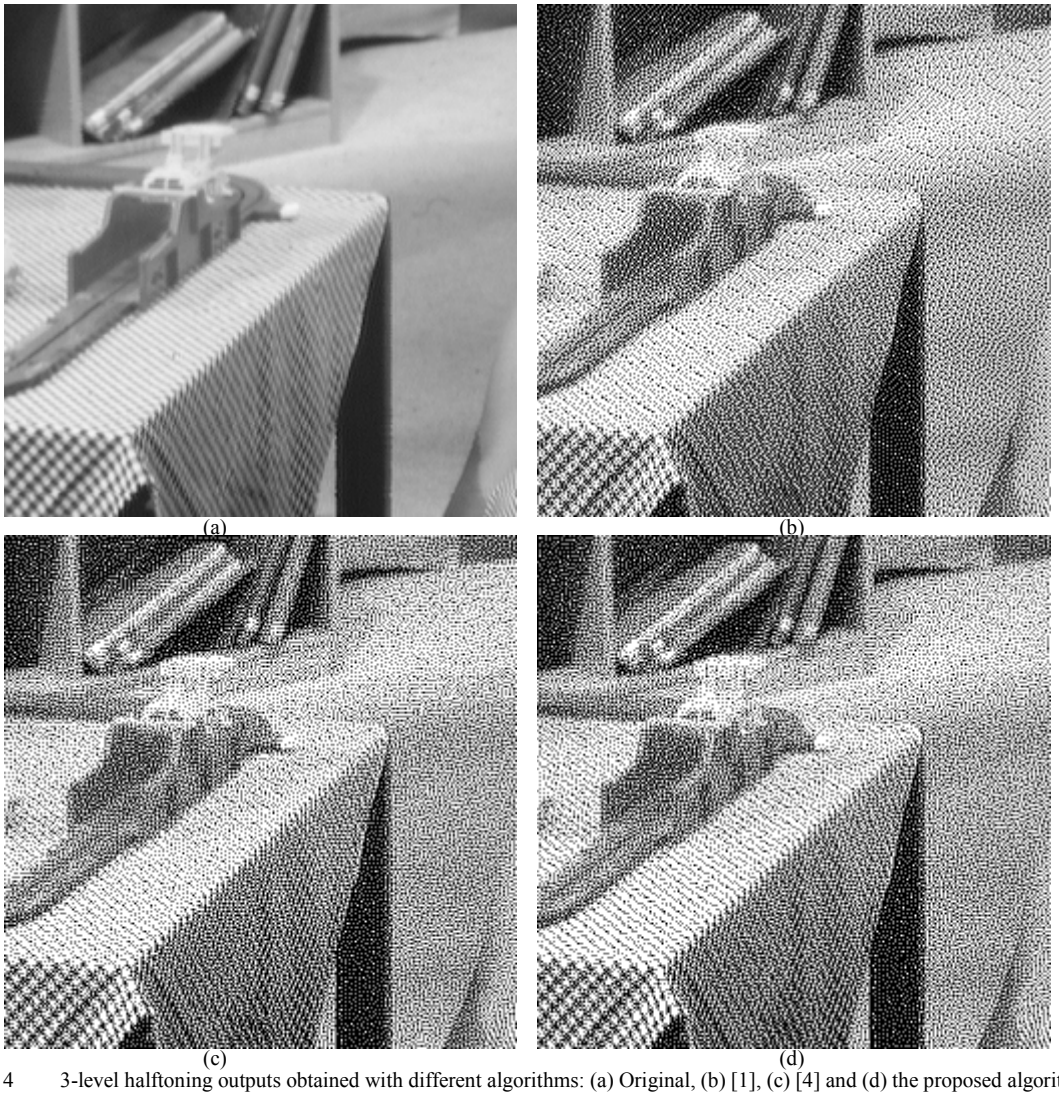

Fig. 4 3-level halftoning outputs obtained with different algorithms: (a) Original, (b) [1], (c) [4] and (d) the proposed algorithm