Improving the Efficiency in Halftone Image Generation Based on Structure Similarity Index Measurement

Aroba KHAN¹, Nonmember, Hernan AGUIRRE¹, and Kiyoshi TANAKA²†, Members

SUMMARY This paper presents two halftoning methods to improve efficiency in generating structurally similar halftone images using Structure Similarity Index Measurement (SSIM). Proposed Method I reduces the pixel evaluation area by applying pixel-swapping algorithm within inter-correlated blocks followed by phase block-shifting. The effect of various initial pixel arrangements is also investigated. Proposed Method II further improves efficiency by applying bit-climbing algorithm within inter-correlated blocks of the image. Simulation results show that proposed Method I improves efficiency as well as image quality by using an appropriate initial pixel arrangement. Proposed Method II reaches a better image quality with fewer evaluations than pixel-swapping algorithm used in Method I and the conventional structure aware halftone methods.

key words: structure aware halftoning, SSIM, pixel-swapping, objective function, phase block-shifting, bit-climbing algorithm

1. Introduction

In the field of halftoning, the conventional methods such as ordered dithering, error diffusion, and so on [1]–[6] mainly focus on preserving tone similarity with their original grayscale images. These methods use various threshold matrixes to generate halftone images and are easy to implement. The images generated by error diffusion methods preserve good texture contents, especially for less-structured smooth images. However, error diffusion has the tendency to enhance edges in an image at the expense of gray level reproduction accuracy. It also suppresses the appearance of artifacts at the cost of over-blurring. Hence, they fail to represent the detailed structures of an image.

In the halftoning problem, an N-gray tone image must be portrayed as an n-gray tone image, where n < N. There are generally two approaches for halftoning. One approach uses filters which are directly applied to the N-gray tone image to generate a halftone image [1]–[6]. Hence, their pixel allocation locks into a regular stable pattern and these generated images fail to simultaneously maintain tone and detailed structure similarity with their original images. The second approach searches directly for the optimum halftoning image having the input N-gray tone image as reference. Our work adopts the latter approach.

W. Pang et al. [7] first proposed a direct search structure aware halftoning technique that uses Structure Similarity Index Measurement (SSIM) [8] in the objective function. This technique uses random black-white pixel pair swapping algorithm which minimizes an objective function that accounts for tone as well as structure similarity. It is applied to an arbitrary binary image. This method’s major advantage is that it can generate high-quality images preserving tone and structure similarity even in the fine detailed regions of an image. Compared to the standard ordered dither and the state-of-the-art error diffusion, it preserves better structure content which is sensitive to the Human Visual System (HVS). However, the paper does not clearly mention the computational efficiency or the MSSIM (mean SSIM) values of the resultant images. Moreover, as the random pixel pairs are picked and swapped from within the whole area of the image, it uses a substantial number of evaluations to generate images with high quality. Also, as it uses an arbitrary binary image, the generated images vary in quality based on the binary image used.

In this paper, we proposed two halftoning methods that overcome the above mentioned drawbacks by improving the image quality as well as efficiency. In proposed Method I, we reduced the search area of random pixel pair selection by applying pixel swapping algorithm within inter-correlated blocks. We also applied phase block shifting in order to use pixels from the adjacent blocks and thus accelerated the image quality improvement. The same objective function as the conventional structure aware halftoning method [7] is used for pixel evaluation. We investigated the effectiveness of reducing search area by block division and phase block shifting. We also investigated the role of an appropriate initial pixel arrangement in generating structurally similar halftone images.

In proposed Method II, to further improve efficiency, we used bit-climbing algorithm within inter-correlated blocks. Each pixel of the block is flipped to the opposite color and evaluated to find an improved objective function value. This ensures that all the pixels in a block get a fair chance to be evaluated and also allows to make appropriate change in grayness where required. We investigate the effectiveness of pixel flipping over pixel swapping and also the role of initial pixel arrangements.

In this paper, first we describe about the relevant background information, followed by two proposed halftoning methods. Then we show some simulation results for each method and include discussions within the sections. Finally, we conclude the main findings of this work followed by future works.

Manuscript received December 30, 2011.
Manuscript revised June 1, 2012.
The authors are with the Faculty of Engineering, Shinshu University, Nagano-shi, 380–8553 Japan.
a) E-mail: ktanaka@shinshu-u.ac.jp
DOI: 10.1587/transinf.E95.D.2495

Copyright © 2012 The Institute of Electronics, Information and Communication Engineers
2. Structure Aware Halftoning Method

2.1 Basic Concept

Recently, based on the hypothesis that the HVS is highly adapted for extracting structural information from a viewing field, Structure Similarity Index Measure (SSIM) [8] has come into consideration for image quality assessment to compare structure similarity between images. Halftoning methods using SSIM in the objective function can compare structure similarity between the generated image and the original image. However, Mean Square Error (MSE) [9] still plays an important role in image quality assessment regarding tone preservation. The halftoning method proposed by W. Pang et al. [7] is an optimization based halftoning technique that minimizes an objective function consisting of both tone and structure components. The optimization can start with any bi-tonal image with global grayness (ratio of black to white pixels) equivalent to that of the original grayscale image. Such initialization can be done by randomly distributing the black/white pixels such that the overall grayness is maintained. For faster convergence, optimization may start with the halftoning result of an existing method. The generated halftone images preserve visually sensitive texture details as well as the local tone, but at the expense of longer execution time.

2.2 Structure Similarity Index Measure (SSIM)

Here we describe the calculation process for SSIM [8]. This image quality metric separates the task of similarity measurement between two images into three comparisons: luminance, contrast and structure. Suppose, \( x_{i,j} \) and \( y_{i,j} \) \( (i, j = 1, 2, 3, \ldots, N) \) are two image pixels, which have been aligned with each other from reference and distorted image. If one of the pixels has perfect quality, then the similarity measure can serve as a quantitative measurement of quality for the second pixel.

First, the luminance comparison function is given by the following equation:

\[
 l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{1}
\]

Here, \( \mu_x \) and \( \mu_y \) are the mean intensities of \( x \) and \( y \), \( C_1 = (K_1L)^2 \) where \( L \) is the dynamic range of the pixel values (255 for 8-bit grayscale images) and \( K_1 = 0.01 \).

To calculate contrast, first the mean intensity is removed from each pixel to obtain the standard deviation \( \sigma_x \) and \( \sigma_y \). The contrast comparison function is obtained as follows:

\[
 c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{2}
\]

\[
 \sigma_x = \left( \sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j}(x_{i,j} - \mu_x)^2 \right)^{\frac{1}{2}} \tag{3}
\]

where \( C_2 = (K_2L)^2 \), \( K_2 = 0.03 \), \( N \) is the number of pixels to be calculated and \( w_{ij} \) is a gaussian function. A Gaussian kernel of size \( 11 \times 11 \) is employed in the implementations. The equation for two-dimensional Gaussian filter is as follows:

\[
 g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}
\]

Here, the standard deviation is taken 1.5 for the Gaussian filter and normalized to unit sum.

The correlation (inner product) between \( (x-\mu_x)/\sigma_x \) and \( (y-\mu_y)/\sigma_y \) is a simple and effective measure to quantify the structural similarity between \( x \) and \( y \). Thus, the structure comparison function is defined by the following equation:

\[
 s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{5}
\]

Here, constant \( C_3 = C_2/2 \) and \( \sigma_{xy} \) is the cross correlation coefficient between \( x \) and \( y \).

\[
 \sigma_{xy} = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j}(x_{i,j} - \mu_x)(y_{i,j} - \mu_y) \tag{6}
\]

Finally, combining the three comparisons of \( l(x, y) \), \( c(x, y) \) and \( s(x, y) \), the resulting similarity measure index SSIM between pixels \( x \) and \( y \) is as follows:

\[
 SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{\mu_x^2 + \mu_y^2 + C_1(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{7}
\]

In order to calculate a single overall quality measure for the entire image, a mean SSIM (MSSIM) is obtained by taking average over all the SSIM values of pixels.

\[
 MSSIM(X, Y) = \frac{1}{N \times N} \sum_{i=1}^{N} \sum_{j=1}^{N} SSIM(x_{i,j}, y_{i,j}) \tag{8}
\]

where \( X \), \( Y \) are the reference and distorted images and \( N \times N \) is the total number of pixels present in an image. MSSIM is valid in the range of \([0, 1]\), with higher values indicating higher similarity.

2.3 Objective Function

Given a grayscale image \( I \), the corresponding halftone image \( I_h \) is obtained by minimizing the following objective function [7] consisting of two terms:

\[
 Objective(I, I_h) = w_g G(I, I_h) + w_c (1 - MSSIM(I, I_h)) \tag{9}
\]

where \( G(I, I_h) \) measures tone similarity and \( MSSIM(I, I_h) \) measures structural similarity between the original and halftone images. These two terms are described in detail in the following subsections. Here, \( w_g \) and \( w_c \) are the weighting factors, such that \( w_g + w_c = 1 \). The weighting factors are set to \( w_g = w_c = 0.5 \) for all experiments.

This objective function is minimized using a simulated
annealing strategy. In each iteration, a random pair of black and white pixels are picked from an arbitrary binary image and swapped. If the swapping decreases the objective function value, the swap is kept. Otherwise, the swapping is undone. This process is continued for many iterations. Since no extra black or white pixel is introduced, the overall grayness should be maintained.

Since the generated halftone image is made up of only black and white pixels, it cannot be compared directly with any grayscale image. So, a Gaussian filter is applied to convert the halftone image to its estimated graylevel image \( g(I_h) \) before comparison.

2.3.1 Tone Similarity Measurement

The term \( G(I, I_h) \) measures MSE [9] between the Gaussian-blurred grayscale input \( g(I) \) and Gaussian-blurred halftone image \( g(I_h) \) in order to evaluate tone similarity. MSE is calculated using the following equation:

\[
G(I, I_h) = \frac{1}{M} \sum (g(I) - g(I_h))^2
\]

This value is normalized between \([0,1]\) to be similar with MSSIM.

2.3.2 Structure Similarity Measurement

A mean SSIM (MSSIM) is used to quantify the structure similarity between Gaussian-filtered halftone result \( g(I_h) \) and original grayscale image \( I \). Since the objective function is a minimization problem, \((1 - MSSIM)\) is considered in Eq. (9).

3. Proposed Halftoning Methods

3.1 Method I

Here we describe the process of Method I using the following sub-sections:

3.1.1 Pixel Swapping within Blocks

In order to reduce the search area of the pixel selection and swapping process, we proposed to divide an existing halftone image into equal sized non-overlapping blocks before applying the pixel swapping algorithm [10].

At first, we took a halftone image generated from any existing halftone method and divided the image into non-overlapping equal sized blocks consisting of \( r \times r \) pixels as shown in Fig. 1. These blocks \( D_{k,l} \) \((k, l = 1, 2, \ldots)\) were also connected to the neighbor blocks in order to avoid discontinuity around the block boundaries as shown in Fig. 2.

We also divided the reference original grayscale image into same sized blocks so that all comparisons could be done within blocks rather than with the whole image. We calculated the MSE and MSSIM value for the first block of the image and evaluated the block’s objective function [7].

Whenever a current block was located at the boundaries of the image, some part of the block’s reference region fell outside the image area. This part was padded with the information which was copied from the image boundary as shown in Fig. 3.

A pair of black and white pixel was selected at random from the first block. Then their locations were swapped and the block’s objective function value was calculated. If this value was less than the block’s initial/previous value, the swapping was kept and the value was updated for next comparison. If not, the swapping was undone. This swapping process was repeated a number of times for each block of the image. In this way, the pixels within a block could be re-
arranged to better express tone and structure similarity with their original images. After one block was complete, it was updated in the image and the same process was continued for all the other blocks of the image.

3.1.2 Phase Block Shifting

After the pixel swapping process was applied to all blocks of the initial halftone image and the image was updated with the swapped pixels, we again divided the image into equal sized blocks that were phase-shifted from their previous block positions [10]. The pixel swapping process was again continued for all the phase-shifted blocks.

We used total four phase block shifting. Figure 4 illustrates the four phases of a block. After the calculation for each phase was complete for the whole image, the next phase was applied and so on. This phase-shifting enabled better pixel replacements also within adjacent block areas. Hence, this accelerated the process of minimizing the objective function value.

3.1.3 Considering an Appropriate Initial Pixel Arrangement

Different initial pixel arrangements have a different distribution of black and white pixels in a halftone image. This initial pixel arrangement plays a vital role in the process of re-generating halftone images since we used pixel swapping algorithm. The black and white pixel pairs present in a binary image were swapped and re-arranged to better resemble tone and structure.

We considered two examples of initial pixel arrangement as shown in Fig. 5. The Stucki’s generated image was very clear and already had a high MSSIM value of 0.647. But we could see that, many light shaded areas of the image were represented with white pixels only, lacking in proper representation of tone and structure. In such error diffusion halftone methods, the pixels were allocated very strictly and locked into a regular stable pattern. Therefore, if we applied pixel swapping operation in this type of initial pixel arrangements, the image could be improved up to a certain level only. But for the case of Random Dither images, the pixels were more loosely distributed while maintaining the local gray level which was similar to the original image. This characteristic was very favorable for the pixel swapping process and with increasing number of trial swaps, the pixels could be well arranged to represent the detailed structures.

3.1.4 Algorithm for Method I

Here, we summarize the algorithm for Method I using the following steps:

Step 1: Divide initial binary image and original grayscale image into equal non-overlapping blocks.
Step 2: Calculate first block’s objective function value.
Step 3: Randomly pick a pair of black and white pixel from first block and swap their locations. Calculate objective function after swap. Keep swap if objective function value decreases. Otherwise undo swap.
Step 4: Continue trial swaps for certain number of evaluations for the block. Update current block information and move to next block.
Step 5: Repeat same pixel swapping process from Step 2 to 5 for all blocks.
Step 6: Re-divide the resultant binary image into blocks that are phase shifted from previous block positions. Repeat process for all phase shifted blocks.
Step 7: Apply a total of four phase shifts.

3.2 Method II

3.2.1 Problems in Method I

As the pixel swapping algorithm is dependent on the swapping of the black and white pixels that are already present in a binary image, there is no scope for changing the gray level for further improvement when necessary. Also, the pixel swapping algorithm randomly selects a black and white pixel pair from an existing binary image. This limits the probability of every pixel being selected and broadens the probability of selecting the same pixel pair more than once.
3.2.2 Pixel Flipping within Blocks

In order to overcome these drawbacks, in proposed Method II, we replaced the pixel-swapping algorithm with bit-climbing algorithm \[11\] to generate structurally similar halftone images.

In proposed Method II, first we divided an arbitrary binary image into equal sized non-overlapping blocks that were correlated with neighbor blocks. This reduced the evaluation area and simplified the process. Then we applied the bit-climbing algorithm to flip each pixel in a raster scan order within these blocks as shown in Fig. 6. Whenever flipping decreased the value of the block’s objective function, it was kept. Otherwise the flipping was undone. This allowed to make proper change in gray level where necessary. After all the pixels of the block were flipped and evaluated, the generated block was updated in the image. This process was continued for all the blocks of the image. We maintained the same pixel flipping sequence because the SSIM value of every pixel is also dependant on its surrounding pixels. After one cycle was complete \((T = 1)\), the same process was repeated for a number of cycles until the decrement of the objective function value saturated \((T \approx 7)\). In this way, the pixels were directly changed to an appropriate value and the generated images could resemble better tone and structure similarity with their original images.

For the order of bit flipping process, we adopted raster scan order because of simple implementation in this paper. However, we can employ other scan orders such as random, Peano \[12\] and Zigzag \[13\] orders, which may affect the image quality and number of cycles until saturation. This point should be investigated as one of the future works.

4. Results and Discussions

4.1 Experimental Setup

In our experiments, we used original images of size 256 × 256 pixels with \(N = 256\) gray levels and the generated images were bi-level halftone images. The block size used was 16 × 16 with equal weighting factors, \(w_g = w_c = 0.5\) for the objective function unless mentioned otherwise. We mainly described the results using “Snail” and “Ribbon” as test images that were used in \[7\].

4.2 Effect of Phase Block Shifting

First, we investigated the effect of phase block shifting (PBS) in proposed Method I using the pixel swapping algorithm. By using this algorithm within blocks, we reduced the search and evaluation area, resulting in the reduction of calculation time. But this also limited the number of suitable pixels available for swapping. So in order to overcome this issue, we applied phase block shifting. This allowed pixels from the surrounding areas of a block also to be available for swapping. Figure 7 shows the effects of applying PBS on objective function values with respect to swap trial numbers for “Snail” and “Ribbon” images respectively. We could see that PBS remarkably accelerated to decrease the value of objective function in an image. Experiments were conducted for a total of 400, 800, 1200, 2000 and 4000 trial swaps per block (this value was normalized for phase shifts i.e. total swap trials / number of phases).

The 4-phase block shift achieved better results than 1-phase or 2-phase block shifts. This was because, whenever there weren’t sufficient pixels available for swapping within a block, with increasing the phases, more pixels became available to find a better swap. So, we could efficiently accelerate (i.e. with fewer evaluations) the generation of halftone images with higher quality by applying PBS.

3.2.3 Algorithm for Method II

We summerize the algorithm for Method II using the following steps:

Step 1: Divide arbitrary binary image and original grayscale image into equal non-overlapping blocks.

Step 2: Calculate first block’s objective function value.

Step 3: Flip each pixel of the first block in a raster scan order and calculate block’s objective function after every flip. Undo flip whenever it doesn’t decrease objective function value.

Step 4: Update current block information and move to next block.

Step 5: Continue same process for all blocks. Thus, runtime \(T = 1\) is complete.

Step 6: Repeat process until saturation \((T \approx 7)\).
4.3 Effect of Initial Pixel Arrangement on Pixel Swapping

Next, we investigated the effect of initial pixel arrangement for proposed Method I. Since in the case of pixel swapping, only the black and white pixels present within an initial binary image were used, the role of an appropriate initial pixel arrangement came into consideration. We considered various initial pixel arrangements and Fig. 8 showed these effects for “Snail” and “Ribbon” images, respectively.

Figure 9 shows an enlarged view for better understanding. As the number of trial swaps increased, the value of objective function reduced for all types of initial pixel arrangements. The highest reduction (improvement) was achieved for Complete Random initial pixel arrangement, but the final value was still worse.

This was because the gray level was not preserved and hence, the low luminance and contrast component values held back the MSSIM and objective function value from improvement. Among others, while Jarvis and Stucki’s initial pixel arrangement showed a low objective function value, the improvement was small and less than Random and Ordered Dither due to their strict pixel distribution. On the other hand, the Random Dither pixel arrangement was worse initially, but achieved the lowest objective function, which was slightly lower than Ordered Dither as we increased the number of trial swaps. The reason for this was that, in Random Dither, the pixels were more loosely distributed than Ordered Dither, while maintaining the local gray level. So, with increasing number of trial swaps, the pixels could be re-arranged preserving SSIM as well as MSE. Hence, the efficiency of image generation along with quality was improved by using random dither initial pixel arrangements.

4.4 Effect of Initial Pixel Arrangement on Pixel Flipping

Next, we investigated the effect of various initial pixel arrangements on proposed Method II which used pixel flipping algorithm. Figure 10 shows the effect of various initial pixel arrangements on objective function values for “Snail” and “Ribbon” images respectively. Here, the objective function value was compared at various run-times rather than trial-swap numbers as in Fig. 9. Trial swap numbers indicated the total number of evaluations for a block whereas run time indicated the number of evaluations per pixel. So, for any block, \( T = 7 \) was equivalent to only 1792 swap trials.

We could see that, although at the initial run-times the rate of improvement for objective function values were different for various initial pixel arrangements, but after some run-times (\( T \approx 6 \)) they eventually merged and saturated. This was because the pixels of an initial binary image were...
directly changed to an appropriate color to reduce the objective function value. After a few run times, all the pixels became equally arranged despite of their initial pixel arrangements. So, our proposed Method II was independent of the initial pixel arrangements. The same quality of halftone images could be obtained from an arbitrary initial pixel arrangement. This was an advantage over the halftoning method proposed by Pang et al. [7] and also our proposed Method I.

4.5 Pixel Swapping versus Pixel Flipping

Here, we compared pixel flipping over pixel swapping at the same number of evaluations and showed the results for different block sizes such as: $4 \times 4$, $8 \times 8$, $16 \times 16$ and $32 \times 32$. Figure 11 shows objective function values for the representative test images “Snail” and “Ribbon” respectively. The evaluation started with random dither as initial pixel arrangement. The term run-time, $T$ was defined as the number of evaluations per pixel within a block. At the first and second run-time ($T = 1$ and $2$), the proposed Method I (pixel swapping) showed better values than proposed Method II (pixel flipping). But the scenario changed from the third run-time ($T \geq 3$) and pixel flipping consistently outperformed pixel swapping. Pixel flipping achieved better values of objective function for all the compared block sizes. However, considering the objective function value and evaluation time, block size $16 \times 16$ had the best results. Hence, our proposed Method II could contribute not only to increase the efficiency but also to further improve the value of objective function.

4.6 Generated Images

Here, in Fig. 12 we visually compared the “Ribbon” image which was generated from various initial pixel arrangements using Method I at 4000 trial swaps per block. So, (4000/4-phases) black-white pixel pairs were swapped in each phase to try to find a better value of objective function. The image generated from Complete Random initial pixel arrangements showed that the local gray level was not preserved but the structure was fairly reconstructed. The low values of luminance and contrast components suppressed the improvement in the MSSIM value. Moreover, since a large area of complete black or white could not be generated due to block size limitation, the mean SSIM reduced. For the case of Stucki pixel arrangements, the lighter shaded regions were represented with only white pixels. As a result, the structure in that region deteriorated. Due to strict pixel arrangement, within that region, the computational blocks...
did not have sufficient black pixels available for swapping. Hence, the structure in those regions could not be regenerated properly. The images generated from Ordered Dither and Random Dither initial pixel arrangements showed the best results having more structure and tone similarity in all regions of the image. The gray level preserved loose pixel distribution of their initial pixel arrangements was favorable for the pixel swapping algorithm.

Next, in Fig. 13 and Fig. 14 we show images generated by Method I and Method II at run-time $T = 3$ and $T = 10$ respectively for visual comparison. In case of both methods, the initial image taken was Random dither. At the early stage, i.e. $T = 3$, Method II generated image had sharp detailed structure but less preserved gray level than compared to Method I generated image. However, at the later stage, i.e. $T = 10$, Method II generated image preserved both fine detailed structure and gray level with their original images than compared to Method I generated image. In other words, by using Method II we could obtain high-quality halftone image with fewer evaluations i.e. at a faster rate.

4.7 Effects of Weights in Objective Function

Table 1 shows the effect of varying the parameter weights of objective function for Method II “Ribbon” image at run-time $T = 7$. We emphasized on MSE or MSSIM by varying the weights and further reduced the value of Objective function. Since the the parameter weights were always equal to unity, there was a trade-off between MSE and MSSIM. As the weight on MSE was increased (the
Table 1  Effect of weights in objective function.

<table>
<thead>
<tr>
<th>$w_g$</th>
<th>$w_c$</th>
<th>MSE</th>
<th>MSSIM</th>
<th>Objective function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>1.0</td>
<td>0.009668009</td>
<td>0.807617</td>
<td>0.192383</td>
</tr>
<tr>
<td>0.1</td>
<td>0.9</td>
<td>0.00901167</td>
<td>0.80956</td>
<td>0.172297167</td>
</tr>
<tr>
<td>0.2</td>
<td>0.8</td>
<td>0.00875693</td>
<td>0.807528</td>
<td>0.155729986</td>
</tr>
<tr>
<td>0.3</td>
<td>0.7</td>
<td>0.00794139</td>
<td>0.808192</td>
<td>0.136648017</td>
</tr>
<tr>
<td>0.4</td>
<td>0.6</td>
<td>0.00704698</td>
<td>0.806738</td>
<td>0.118775992</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>0.00625564</td>
<td>0.807656</td>
<td>0.09929982</td>
</tr>
<tr>
<td>0.6</td>
<td>0.4</td>
<td>0.0053429</td>
<td>0.805526</td>
<td>0.0899534</td>
</tr>
<tr>
<td>0.7</td>
<td>0.3</td>
<td>0.0042721</td>
<td>0.804619</td>
<td>0.061587347</td>
</tr>
<tr>
<td>0.8</td>
<td>0.2</td>
<td>0.00313602</td>
<td>0.802329</td>
<td>0.042043016</td>
</tr>
<tr>
<td>0.9</td>
<td>0.1</td>
<td>0.00198458</td>
<td>0.793285</td>
<td>0.022547622</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0</td>
<td>0.00137809</td>
<td>0.554906</td>
<td>0.00137809</td>
</tr>
</tbody>
</table>

weight on MSSIM reduced), the value of MSE was further reduced improving the gray level preciseness. But the structure similarity was reduced. Figure 15 shows the visual comparison of the generated images at various weights: $(w_g, w_c) = (0.0, 1.0), (0.5, 0.5), (1.0, 0.0)$ respectively. We see that, when all the weight was on SSIM, the generated image was very sharp and clearly defined all the structures but the gray level was not preserved. On the other hand, when all the weight was on MSE, the generated image was blur and failed to represent all the structural details of the image. However, when the weights were equal for MSE and SSIM, the generated image had a balance in preserving both gray level and structure similarity.

5. Conclusions

To improve the efficiency of generating structure similarity based halftone images, we proposed two halftoning methods. Firstly, we reduced the evaluation area by applying pixel swapping algorithm within inter-correlated blocks and showed the effect of using an appropriate initial pixel arrangement to improve efficiency in proposed Method I. Secondly, we replaced the pixel swapping algorithm with pixel-flipping (bit-climbing algorithm) in proposed Method II. We could show that pixel flipping further improved the efficiency and was independent from the effect of initial pixel arrangement. It also contributed to achieving a better objective function value in the generated images.

As future works, the effect of changing the objective function parameter weights adaptively based on the local image features should be brought into consideration. Also, the effects of employing other scan orders except the raster scan order for bit flipping within blocks should be investigated. Furthermore, the extension to color image halftoning should be investigated.

References


Aroba Khan received her B.S. degree in Electrical and Electronic Engineering from Khulna University of Engineering and Technology (KUET), Bangladesh in 2007. After working as a lecturer at Eastern University, Bangladesh for two years, she completed her M.S. degree in Electrical and Electronic Engineering from Shinshu University, Nagano, Japan in 2011. Her research interests includes image processing and evolutionary algorithm.
Hernan Aguirre received his Engineer degree in computer systems from Escuela Polit'ecnica Nacional, Quito, Ecuador in 1992. From 1997 to 2003 he was a research scholar sponsored by the Japanese Ministry of Education, Culture, Sports, Science and Technology. He received the M.S. and Ph.D. degrees from Shinshu University, Japan, in 2000 and 2003, respectively. Currently, he is an associate professor in the Department of Electrical and Electronic Engineering, Faculty of Engineering, Shinshu University. His research interests include multi-objective evolutionary algorithm, multidisciplinary design optimization, computational intelligence and their applications. He received iFAN2010 Best paper Award from SICE and GECCO2011 Best Paper Award from ACM-SIGEVO. He is a member of IEEE, ACM-SIGEVO, IEICE and IPSJ.

Kiyoshi Tanaka received his B.S. and M.S. degrees in Electrical Engineering and Operations Research from National Defense Academy, Yokosuka, Japan, in 1984 and 1989, respectively. In 1992, he received Dr. Eng. degree from Keio University, Tokyo, Japan. In 1995, he joined the Department of Electrical and Electronic Engineering, Faculty of Engineering, Shinshu University, Nagano, Japan, where he is currently a full professor. His research interests include image and video processing, human visual perception, information hiding, evolutionary computation, multi-objective optimization, smart grid, and their applications. He received IEVC2010 Best Paper Award from IIEEJ, iFAN2010 Best paper Award from SICE, GECCO2011 Best Paper Award from ACM-SIGEVO and ISPACS2011 Best Paper Award from IEEE. He is a member of IEEE, IPSJ and IIEEJ. He is the editor in chief of Journal of the Institute of Image Electronics Engineers Japan.