Low-complexity high-performance multiscale error diffusion technique for digital halftoning

Yik-Hing Fung
Ka-Chun Lui
Yuk-Hee Chan
The Hong Kong Polytechnic University
Department of Electronic and Information Engineering
Hung Hom 510641
Hong Kong, China
E-mail: enyhchan@polyu.edu.hk

Abstract. *Multiscale error diffusion (MED) is superior to conventional error diffusion algorithms as it can eliminate directional hysteresis completely. However, due to its frame-oriented processing nature, the computational complexity is comparatively high. Furthermore, though theoretically MED can remove directional hysteresis by eliminating predefined scanning paths and causal filters, no comprehensive quantitative analysis on this issue can be found in the literature. A fast MED algorithm is proposed and a detailed analysis on the performance of various MED algorithms including the proposed one are provided. Analysis and simulation results show that the proposed algorithm can reduce the computational complexity without sacrificing the image quality as compared with conventional MED algorithms. The proposed algorithm also supports parallel processing and hence can further reduce the processing time. © 2007 SPIE and IS&T.* [DOI: 10.1117/1.2435197]

1 Introduction

Digital halftoning is a process that converts a gray-level image into a bilevel image and has been widely used in a number of applications such as printing.[1] Though error diffusion can provide a better result than the other halftoning algorithms such as order dithering at a reasonable cost, it suffers from artifacts such as pattern noise, worm-like artifacts, and directional hysteresis.[2] To reduce these artifacts, different modifications to the standard error diffusion[3] were made. For example, Wong's algorithm[4] adjusts the error diffusion filter adaptively and the Nagae'*et al.* algorithm[5] processes the pixels along a space-filling curve. However, most of these modifications are based on the same error diffusion framework in which pixels are processed in a predefined scanning order.

Some other approaches adopt a multiscale approach to produce a halftone by handling the process at multiple spatial resolutions. For example, Peli's algorithm[6] eliminates directional error diffusion and iteratively modifies selected binarized pixels to reduce the weighted averaged error of local regions. However, Peli's algorithm is not an error diffusion algorithm. Katsavounidis and Kuo's algorithm[7] uses a noncausal filter and a nonpredetermined scanning order to halftone a gray-level image. It is superior to some other conventional halftoning algorithms in a way that it preserves the advantages of error diffusion without suffering directional hysteresis. Chan's[8] and Chan and Cheung's[9] algorithms are modified versions of that presented in Ref. 7. It was found that Katsavounidis and Kuo's algorithm introduced error leakage and pattern noise during error diffusion and hence degraded the quality of its produced binary halftones. Chan modified Katsavounidis and Kuo's algorithm[7] accordingly to solve these problems in Refs. 8 and 9. Since no directional error diffusion and no predefined scanning order is involved in these algorithms, theoretically no directional hysteresis exist in the halftoning outputs of these algorithms.

Essentially, all multiscale error diffusion (MED) algorithms[7–9] are two-step iterative algorithms.[7–9] At each iteration, they first locate a critical pixel in the output image **B** based on an updated version of the input image **X** and assign it a binary value. Then the quantization error of the selected pixel is diffused to the neighboring pixels with a noncausal filter so as to update **X**. The iterations are repeated until the sum of all elements of the updated **X** is bounded in absolute value by 0.5. However, due to their frame-oriented processing nature, their complexity is very high as compared with conventional halftoning algorithms such as standard error diffusion.[3]

Two questions arose immediately from the preceding observation. The first is how to reduce the realization effort of MED and the second is whether the realization effort paid in MED is worthwhile. The first question inspires our search for a low-complexity alternative to the realization of MED. As for the second question, its answer relies on a thorough study on how critically and significantly a MED algorithm can improve the halftone quality as compared with a non-MED algorithm. However, though the qualitative ground for MED algorithms to eliminate directional hysteresis is explained and simulation results are presented to support the ground in Refs. 7–9, no comprehensive and quantitative analysis on the performance of a MED algorithm can be found in the literature. As the performance of a halftone algorithm is always the theoretical interest of the researchers working in the area and a systematic analysis on this issue is practically useful, we would also like to conduct a study on this issue.

Paper 06112R received Jun. 29, 2006; revised manuscript received Nov. 13, 2006; accepted for publication Nov. 13, 2006; published online Jan. 31, 2007.
1017-9909/2007/16(1)/013010/12/$25.00 © 2007 SPIE and IS&T.

This paper presents a low-complexity alternative to realize MED. Similar to other MED algorithms, the proposed algorithm removes the scanning-path and filter constraints to eliminate directional hysteresis. The difference is that it puts its focus on the realization complexity and reduces it by tackling the technical problems in a different way. Unlike other MED algorithms,[7–9] which are basically frame-oriented, the proposed algorithm performs error diffusion in a block-based manner to support parallel processing and reduce the effort for locating critical pixels. This is based on the idea that, during multiscale error diffusion, the quantization error of a pixel is usually consumed during its propagation to a distant pixel. The diffusion result of two distant pixels is likely to be independent and hence processing blocks in parallel makes sense to a certain extent. After the proposed algorithm is presented, a detailed analysis on the performance of various MED and the proposed algorithms is given.

The organization of this paper is as follows. Section 2 proposes a fast MED algorithm. In Sec. 3, a comprehensive analysis on the quality performance and the complexity of various MED algorithms including the proposed one are provided. This quantitatively explains why MED is better than conventional halftoning algorithms and proves that the proposed algorithm can reduce the complexity without sacrificing its halftoning performance in terms of different measures. In Sec. 4, simulation results are given to evaluate the performance of the proposed algorithm. Finally, conclusions are given in Sec. 5.

2 Proposed Algorithm

In conventional[7,8] MED, all pixel values of the output image **B** are initialized to be zero and then, based on the grayscale input image **X**, an appropriate number of pixels of **B** are picked iteratively to assign value 1 until the average pixel intensity of **B** is equivalent to that of **X**. From another point of view, white dots are iteratively put in a black background. A considerable amount of realization effort is paid for locating the positions to put the white dots and this effort is proportional to the number of white dots to be introduced. The proposed algorithm reduces the complexity by reducing the number of dots to be put in and the amount of effort to locate a position for a dot.

To reduce the number of dots to be handled, the proposed algorithm first estimates the average intensity of **X**. Without losing of generality, we assume that the maximum and the minimum pixel values of **X** are, respectively, 1 and 0. If the average pixel value of **X** is less than 0.5, white dots should be introduced to a black background. Otherwise, black dots should be introduced to a white background to reduce the realization effort. Hereafter, we assume that white dots are the minority dots and they are introduced to a black background. If it is the opposite, one can negate all pixel values of **X** before carrying out the proposed algorithm and negate all pixel values of the output at the end. The dot budget is defined to be the number of minority dots to be settled and it is the rounded value of $\min(S_x - I_x, I_x)$ where S_x is the total number of pixels in **X**, and I_x is the sum of all pixel values of **X**. Operator $\min(\cdot)$ picks the minimum value from its inputs.

The gray-scale input image **X** is then partitioned into a number of 4×4 nonoverlapped blocks. For each block, an intensity pyramid is constructed as shown in Fig. 1. In formulation, we have

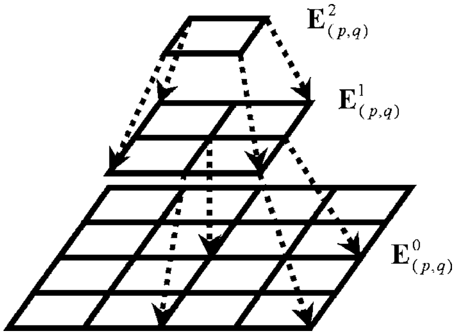


Fig. 1 Intensity pyramid associated with a block of 4×4 pixels.

$$E^k_{(p,q)}(i,j) = \begin{cases} \sum_{m=0,1} \sum_{n=0,1} E^{k-1}_{(p,q)}(2i+m, 2j+n) & \text{if } k = 1,2 \\ X_{(p,q)}(i,j) & \text{if } k = 0 \end{cases}$$

$$\text{for } i,j = 0,1,\ldots,\max(0, 2^{2-k} - 1), \qquad (1)$$

where $X_{(p,q)}(i,j)$ is the intensity value of the (i,j)'th pixel of the (p,q)'th block of **X** and $E^k_{(p,q)}(i,j)$ is the value of the (i,j)'th element of the k'th level of the intensity pyramid associated with the (p,q)'th block of **X**.

Every four adjacent blocks are grouped together to form a macroblock of 8×8 pixels. Except those macroblocks whose total pixel intensity value is less than 0.5, which implies no more white dot should be put to them, all macroblocks are processed in parallel as follows. For each macroblock, the block that carries the maximum total intensity [i.e., the block which has the maximum $E^2_{(p,q)}(0,0)$] in the macroblock is picked and the most wanted pixel in the block is located with the intensity pyramid associated with the selected block by following the maximum intensity guidance. Specifically, when the maximum intensity guidance is adopted, one should always proceed from the current node at level k to its child node of maximum $E^{k-1}_{(p,q)}(i,j)$. At any time, when there is more than one maximum encountered in the search, we select one of them randomly. When level 0 is reached, the selected node specifies the most wanted pixel. For instance, if the node holding $E^0_{(p,q)}(x,y)$ is the node, the (x,y)'th pixel of the block will be the most wanted pixel.

Unless the selected pixel is a boundary pixel of the macroblock, $X_{(p,q)}(x,y)$ should then be quantized to 1 and $B_{(p,q)}(x,y)$, the intensity value of the (x,y)'th pixel of the (p,q)'th block of **B**, is assigned value 1. The reason for discriminating the boundary pixels of a macroblock is discussed later. For the sake of reference, the region in which a pixel can be quantized after being selected is referred to as a qualified region.

Suppose the selected pixel is in the qualified region of a macroblock. After quantizing it to 1, its quantization error $e = X_{(p,q)}(x,y) - 1$ is diffused to the neighbors of the pixel with a noncausal filter to update **X** as follows:

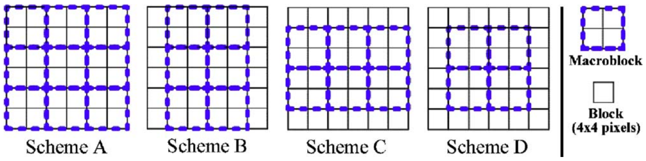


Fig. 2 Four grouping schemes.

$$X_{(p,q)}(x+m,y+n) = \begin{cases} 0 & \text{if } m=n=0 \\ X_{(p,q)}(x+m,y+n)+ew(m,n) & \text{else} \end{cases} \quad \text{for } -1 \le m,n \le 1, \tag{2}$$

where $w(m,n)$ is a coefficient of the diffusion filter defined as

$$w(m,n) = \begin{cases} 1/12 & \text{if } |m|=|n|=1 \\ 1/6 & \text{if } |m| \neq |n| \ \& \ |m|,|n| \le 1. \end{cases} \tag{3}$$

Since only the central part of a macroblock can be quantized and the diffusion filter is of size 3×3, no error can be diffused outside the macroblock and hence all marcoblocks can be processed independently. In other words, all macroblocks can be processed in parallel to reduce the processing tune. This explains why, in the proposed algorithm, boundary pixels of a macroblock are discriminated and not further processed when they are selected. After diffusion, all intensity pyramids of the affected blocks are updated according to Eq. (1).

To provide a chance to handle the boundary pixels of a macroblock and eliminate the potential blocking artifacts caused by the block-based approach, the proposed algorithm changes the way how it groups blocks to form a macroblock in the course of halftoning as follows. After processing all macroblocks as mentioned before in parallel once, all blocks of **X** are regrouped to form new macroblocks. Four grouping schemes are used in turns in the proposed algorithm. As an example, Fig. 2 shows how the four schemes group the blocks in an image of size 6×6 blocks differently. A pixel that is a boundary pixel of a macroblock in a particular round may not be a boundary pixel of a macroblock again in next round. By doing so, all pixels of **X** can be taken care in the course. Note that the regrouping does not affect the intensity pyramids of the blocks and hence does not increase the complexity in this aspect.

The overall effect of using the grouping schemes in turns and excluding the boundary pixels of a macroblock from being processed is equivalent to processing overlapped 6×6-pixel regions, each of which overlaps each of its four-connected neighboring 6×6-pixel regions with an area of 2×6 or 6×2 pixels. Blocking artifacts can hence be eliminated with this approach.

Figure 3 summaries the proposed algorithm in pseudo code. The algorithm iteratively allocates white dots to **B** until all budgeted dots are used up. At each iteration, a considerable number of white dots are allocated at a time. When allocating the dots, it quantizes corresponding pixels of **X** and diffuses the quantization errors. It is possible that, at the last stage of the halftoning process, while there are still budgeted dots on hand, there is no macroblock whose total residual intensity is larger than 0.5. In such a case, to settle the budgeted dots left behind, we select the proper number of macroblocks that carry the most total residual intensity among all macroblocks to locate the pixels for putting the dots.

```
Determine the number of minority dots that should be put
Construct intensity pyramids of all blocks
While the minority dots have not yet totally been located
    For Scheme S = A, B, C, D
        Partition image X with Grouping Scheme S
Parallel
        For each macroblock
            If total residual intensity of the macroblock>0.5
            (Here we assume minority dots are white. The criterion should be adjusted if they are black.)
                Locate the most wanted pixel
                If it is in the qualified region of the macroblock
                    Quantize it
                    Diffuse quantization error
                End
            End
        End
        Update intensity pyramids of all affected blocks
    End
End
```

Fig. 3 Pseudocode of the proposed algorithm.

To handle a boundary pixel or a corner pixel of **X**, diffusion filters such as $[0,0,0;0,*,2;0,2,1]/5$ or $[0,0,0;2,*,2;1,2,1]/8$ is used instead of Eq. (3) to avoid energy leakage, where * marks the position of $w(0,0)$. The macroblocks containing these corner or boundary pixels should also extend their qualified regions to allow quantizing these pixels.

3 Performance Analysis

This section provides an analysis on the performance of various MED algorithms including the proposed one in terms of their output quality and computational complexity. This analysis serves two purposes. First, it quantitatively shows that MED is better than conventional error diffusion and its remedial schemes in various aspects including spatial dot distribution and noise characteristics, which is not provided in any literature[7–9] about MED. Second, it shows that, as compared with other MED algorithms, the performance of the proposed algorithm is comparable to those in Refs. 8 and 9 and much better than that in Ref. 7 while its complexity is significantly reduced.

3.1 *Dot Distribution and Noise Characteristic*

In this part of analysis, dot distribution and noise characteristics of the halftones generated with different halftoning algorithms were studied based on their spatial and spectral statistics.

In spatial statistics, Lau and Arce[2] developed a directional distribution function $D_{r_1,r_2}(\alpha)$ to measure the directional distribution of dots in a dot pattern. In particular, it is defined as the expected number of dots per unit area in an angular segment of the ring bounded by inner radius r_1 and outer radius r_2. The ring is centered at a dot and the segment is indexed by α. In ideal case, we have $D_{r_1,r_2}(\alpha)=1$ for all α, which indicates an isotropic distribution in the pattern. Note that $D_{r_1,r_2}(\alpha)>1$ and $D_{r_1,r_2}(\alpha)<1$, respectively, indicate a favoring and an inhibition of dots in direction α. In our analysis, the annular ring is defined by $r_1=0$ and $r_2=\max(\lambda,3)$, where λ is the principle wavelength of the input gray level, and the ring is divided into 16 equal segments.

Table 1 Summary of the algorithms evaluated for comparison.

Algorithm	Scanning Order	Error Diffusion Filler	Process
SED$_r$ (Ref. 3)	Raster	Nonadaptive, causal	Filter support based
SED$_{sr}$ (Ref. 3)	Serpentine raster	Nonadaptive, causal	Filter support based
AED (Ref. 4)	Raster	Adaptive, causal	Filter support based
HED (Ref. 5)	Space-filling curve	Nonadaptive, causal	Filter support based
PED (Ref. 6)	—	—	Frame based
MED$_k$ (Ref. 7)	Max intensity guidance	Nonadaptive, noncausal	Frame based
MED$_c$98 (Ref. 8)	Max intensity guidance	Adaptive, noncausal	Frame based
MED$_c$04 (Ref. 9)	Extreme intensity guidance	Adaptive, noncausal	Frame based
Proposed	Max intensity guidance	Nonadaptive, noncausal	Block based with four grouping schemes
DD (Ref. 10)	Class matrix guidance	Adaptive, noncausal	Block based
BED (Ref. 11)	Raster	Nonadaptive, causal	Filter support and block based

In spectral statistics, Ulichney[1] developed two spectral statistics to analyze a halftone pattern. The first one is radically averaged power spectrum density (RAPSD). It is defined as the average power in the annular ring with center radius f_p as follows:

$$P(f_p) = \frac{1}{N[R(f_p)]} \sum_{f \in R(f_p)} \hat{P}(f), \tag{4}$$

where $N[R(f_p)]$ is the number of frequency samples in $R(f_p)$, which is an annular ring of width Δ_p partitioned from the spectral domain; and $\hat{P}(f)$ is the magnitude square of the Fourier transform of the output pattern divided by the sample size. The second spectral statistic is anisotropy, which is defined as

$$A(f_p) = \frac{1}{N[R(f_p)] - 1} \sum_{f \in R(f_p)} \frac{[\hat{P}(f) - P(f_p)]^2}{P^2(f_p)}. \tag{5}$$

It provides the SNR of frequency samples of $\hat{P}(f)$ in $R(f_p)$ and is used to measure the strength of directional artifact.

In our analysis, various error diffusion algorithms were applied to a constant gray-level image of size 128×128 and the dot distributions in their outputs were studied in terms of the aforementioned statistics. Multiscale frame-based algorithms including PED (Ref. 6), MED$_k$ (Ref. 7), MED$_c$98 (Ref. 8), and MED$_c$04 (Ref. 9) and the proposed algorithm were included in the comparison. Strictly speaking, PED is not a MED algorithm, but it distributes dots from a multiscale point of view. Due to page constraint, only a few conventional error diffusion algorithms are reported here for comparison. In particular, for non-multiscale-based algorithms, this paper presents the results of standard error diffusion with raster[3] (SED$_r$), serpentine raster[3] (SED$_{sr}$), and space-filling-curve[5] (HED) scanning schemes. The results of an adaptive error diffusion

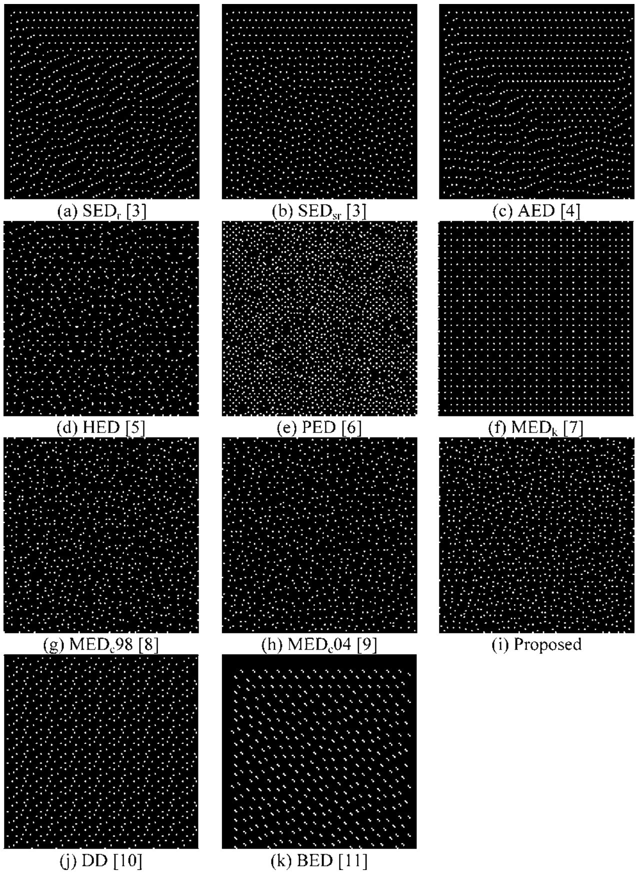
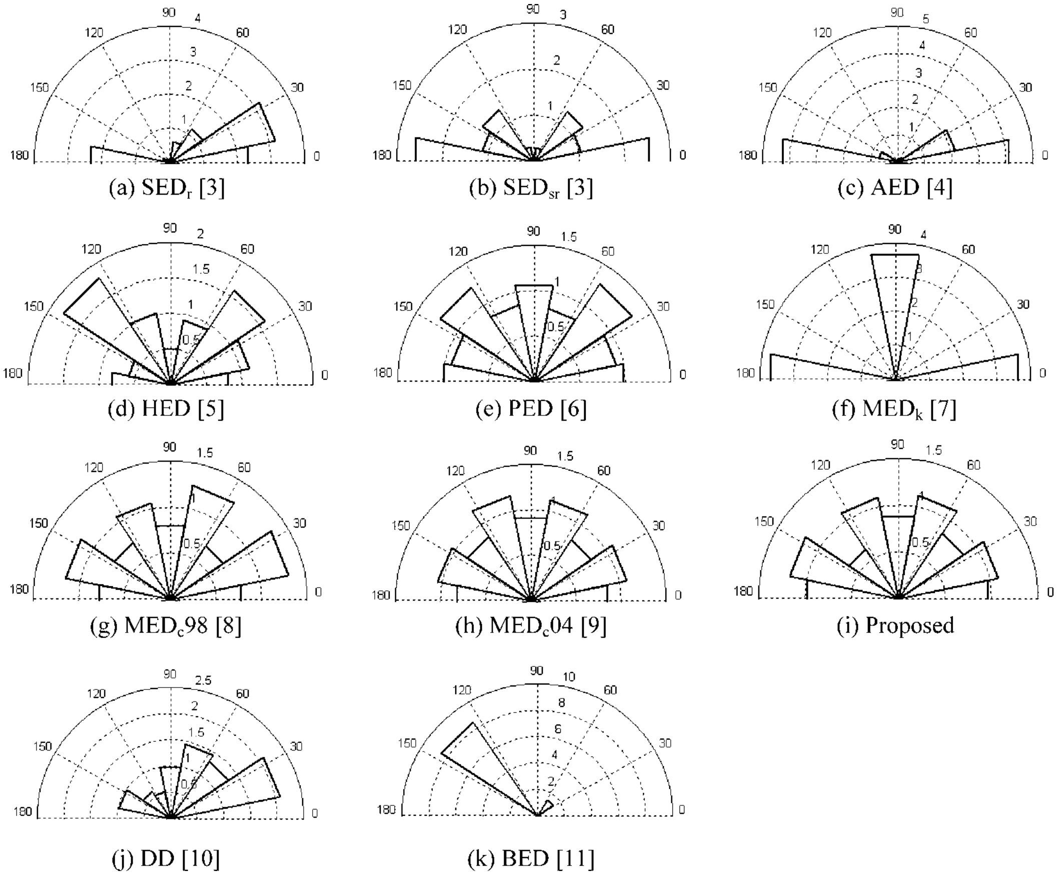


Fig. 4 Halftoning results of a 128×128 input of constant gray-level 13/255 for the (a) SED$_r$, (b) SED$_{sr}$, (c) AED, (d) HED, (e) PED, (f) MED$_k$, (g) MED$_c$98, (h) MED$_c$04, (i) proposed, (j) DD and (k) BED algorithms.



(a) SED$_r$ [3]

(b) SED$_{sr}$ [3]

(c) AED [4]

(d) HED [5]

(e) PED [6]

(f) MED$_k$ [7]

(g) MED$_c$98 [8]

(h) MED$_c$04 [9]

(i) Proposed

(j) DD [10]

(k) BED [11]

Fig. 5 Corresponding directional distribution functions of Fig. 4.

algorithm[4] (AED) are also presented for comparison. Note that they are all typical examples of eliminating directional hysteresis by adjusting the error diffusion filter[4] and the scanning path[5] to diversify the error distribution direction. Two block-based error diffusion algorithms including Mese and Vaidyanathan[10] dot diffusion (DD) algorithm and Damera-Venkdata *et al.*'s[11] block error diffusion (BED) algorithm[11] are also included in our simulations for comparison. The dot shape used in simulating[11] is a 2×2 cluster $[1,0;0,1]$. Table 1 summarizes the presented algorithms.

Figure 4 shows the halftoning results of a 128×128 image of constant gray level $g=13/255$. Directional ripples appear in Figs. 4(a)–4(c), which implies directional hysteresis. There are pattern artifacts in Fig. 4(f). Figures 4(g)–4(i) are visually better than Figs. 4(a)–4(c) as they do not contain any directional ripples and pattern artifacts. As for Fig. 4(e), one can see that dots are denser than the other outputs. In fact, PED tends to introduce more minority dots than necessary, which results in a brighter or darker output.

Severe blocking effect and pattern artifacts can be found in Fig. 4(j). This is expected as blocks are processed independently and a predefined class matrix is used in DD to determine the processing order of the pixels in a block. BED processes blocks and diffuses error in a raster scanning order as in standard error diffusion and hence artifacts caused by directional hysteresis exist in Fig. 4(k). As a block-based algorithm, the proposed MED algorithm successfully eliminates the blocking effect by using four grouping schemes in turns. Besides, no predefined scanning order and no fixed causal filter is used in the proposed algorithm. Consequently, no blocking artifact and directional ripple is found in Fig. 4(i).

Figure 5 shows the directional distribution functions $D_{r_1,r_2}(\alpha)$ of the patterns shown in Fig. 4. Note that only the upper halves of the plots are shown. The lower half of the plot of $D_{r_1,r_2}(\alpha)$ can be obtained with $D_{r_1,r_2}(\alpha+\pi)=D_{r_1,r_2}(\alpha)$. The plots shown in Figs. 5(a)–5(d), 5(j), and 5(k) are not symmetric in all directions. This reflects the fact that in Figs. 4(a)–4(d), 4(j), and 4(k) dots are not uniformly distributed in all directions and they suffer from directional hysteresis. For SED$_r$ and BED, this is expected as they exploit causal diffusion filter and fixed scanning path. Though SED$_{sr}$, AED, and HED are purposely proposed to reduce the directional hysteresis, they cannot eliminate it completely.

Theoretically, MED$_k$, MED$_c$98, MED$_c$04, and the proposed algorithm can eliminate directional hysteresis as no causal filter and no predetermined scanning path is used in these algorithms. One can see that the plots in Figs. 5(e)–5(i) are more or less symmetric in all four directions (east, north, west, and south). This supports the theory. However, when the same issue is addressed at a finer direction resolution, MED$_c$98, MED$_c$04, and the proposed algorithm are better than MED$_k$ in a way that their plots are symmetric in eight directions, while MED$_k$'s plot is not. In other words, MED$_c$98, MED$_c$04, and the proposed algorithm can eliminate directional hysteresis in more directions. PED's performance is comparable to MED$_c$'s.

To have a better picture of the directional hysteresis introduced by a halftoning algorithm, a measure called directional index function is defined as

$$\sigma^2(g) = \frac{1}{16}\sum_{\alpha=1}^{16}[1 - D_{0,\max(\lambda,3)}(\alpha)]^2, \quad \forall\, g \tag{6}$$

where $D_{0,\max(\lambda,3)}(\alpha)$ is the directional distribution function values of the algorithm's halftoning output of a constant input, g is the gray level of the input, and λ is the principle wavelength of the input. This measure does not carry any information about the direction of the directional hysteresis in the output. It simply reflects how severe the direction hysteresis is in the output. The greater the value, the more severe the directional hysteresis is. In ideal case, its value should be zero.

Figure 6 shows the directional index functions of the presented algorithms at different gray levels. A logarithmic scale is used for the abscissa in the plot. In Fig. 6, we can see that PED, MED$_c$98, MED$_c$04, and the proposed algorithms provide a better performance as compared with the others in terms of this measure. Another interesting observation is that, though MED$_k$ is more symmetric than SED$_r$, SED$_{sr}$, and AED in four directions, when the direction resolution is increased from 4 to 16, the directional index function values of MED$_k$ are even larger than those of the others in quite a number of gray-level inputs.

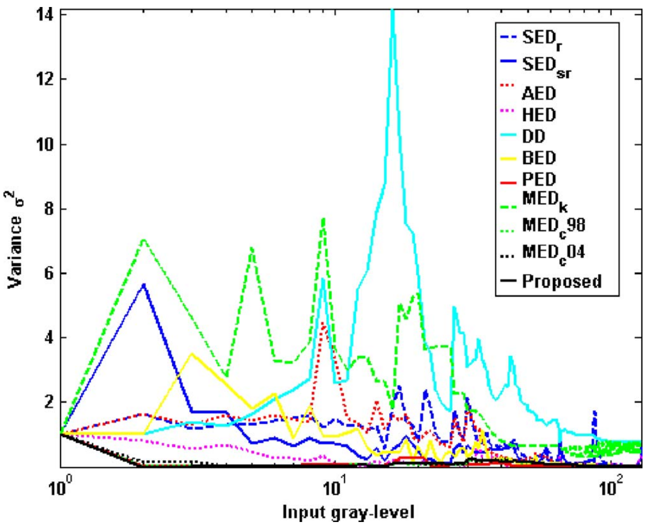
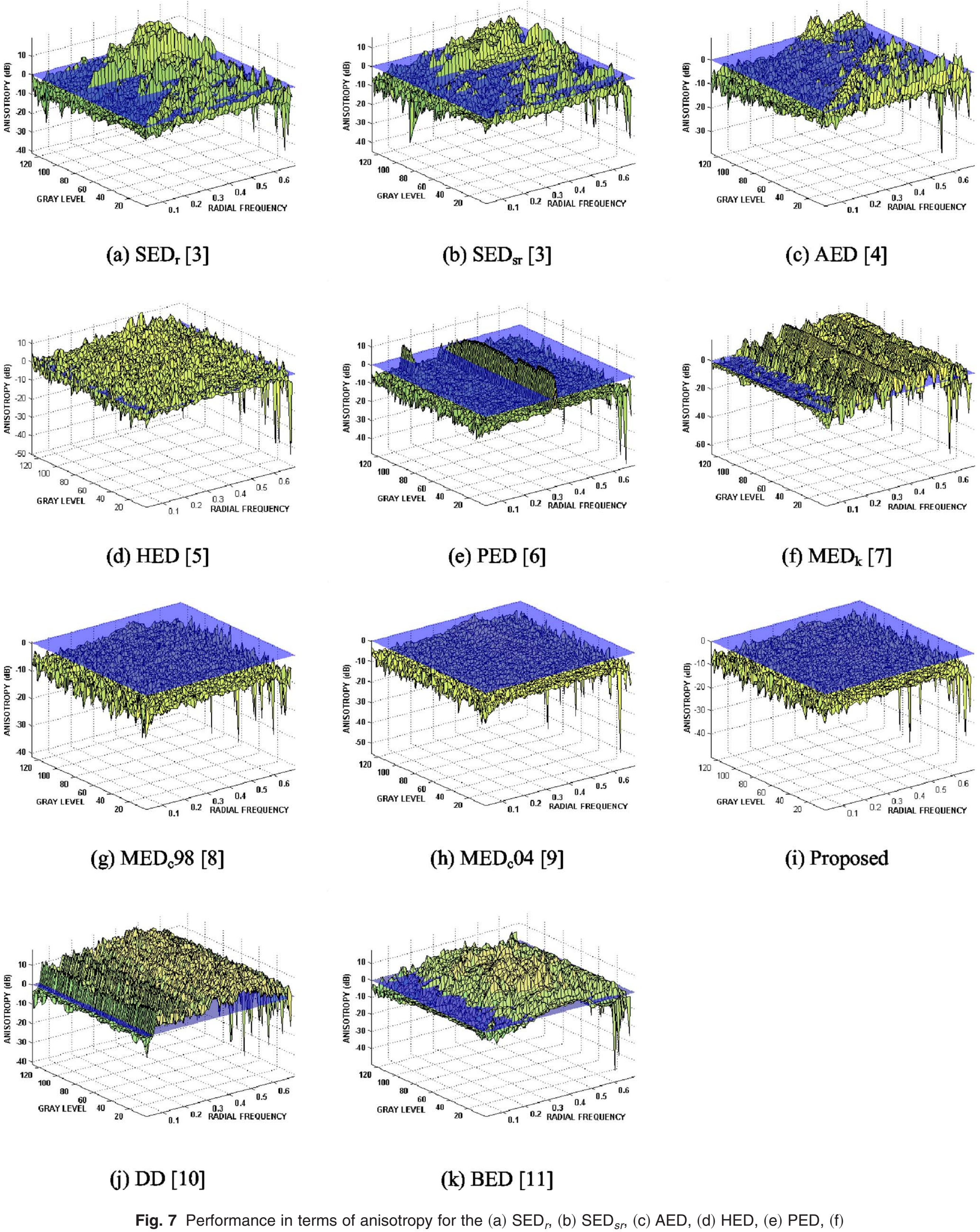


Fig. 6 Performance in terms of directional distribution of dots.

Figure 7 shows the performance of various algorithms in terms of the anisoiropy of dots in their halftoning results of different constant gray-level inputs. As mentioned in Ref. 1, when $A(f_p) > 0$ dB happens directional components are considered to be strong or noticeable to human eyes. To provide a reference to study the performance of the algorithms, a surface defined by $A(f_p) = 0$ dB is added in each of the plots. The plots show that MED$_c$98, MED$_c$04, and the

Table 2 Average number of operations per pixel of various MED algorithms.

Image Size	Image	Average Number of Operations per Pixel								
		MED$_k$			MED$_c$98			Proposed		
		ADD	CMP	MUL	ADD	CMP	MUL	ADD	CMP	MUL
256×256	"Baboon"	14.67	12.12	1.01	13.66	32.32	1.01	9.45	6.72	0.99
	"Barb"	13.32	11.01	0.92	12.39	29.37	0.92	8.79	6.67	0.92
	"Boat"	15.45	12.78	1.06	14.37	34.07	1.06	8.94	6.80	0.94
	"Lena"	14.01	11.59	0.97	13.04	30.91	0.97	9.24	7.04	0.97
	"Peppers"	11.77	9.75	0.81	10.96	26.00	0.81	7.76	6.34	0.82
	Average	**13.84**	**11.45**	**0.95**	**12.88**	**30.53**	**0.95**	**8.83**	**6.71**	**0.93**
512×512	"Baboon"	11.87	11.01	0.82	14.26	36.47	1.01	9.48	6.68	0.99
	"Barb"	14.77	13.67	1.01	12.95	33.15	0.92	8.86	6.64	0.93
	"Boat"	13.41	12.43	0.92	15.02	38.44	1.07	8.97	6.67	0.94
	"Lena"	15.55	14.41	1.07	13.64	34.88	0.97	9.31	6.96	0.97
	"Peppers"	14.12	13.08	0.97	11.47	29.35	0.82	7.84	6.28	0.82
	Average	**13.95**	**12.92**	**0.96**	**13.47**	**34.46**	**0.96**	**8.89**	**6.65**	**0.93**



(a) SED$_r$ [3]

(b) SED$_{sr}$ [3]

(c) AED [4]

(d) HED [5]

(e) PED [6]

(f) MED$_k$ [7]

(g) MED$_c$98 [8]

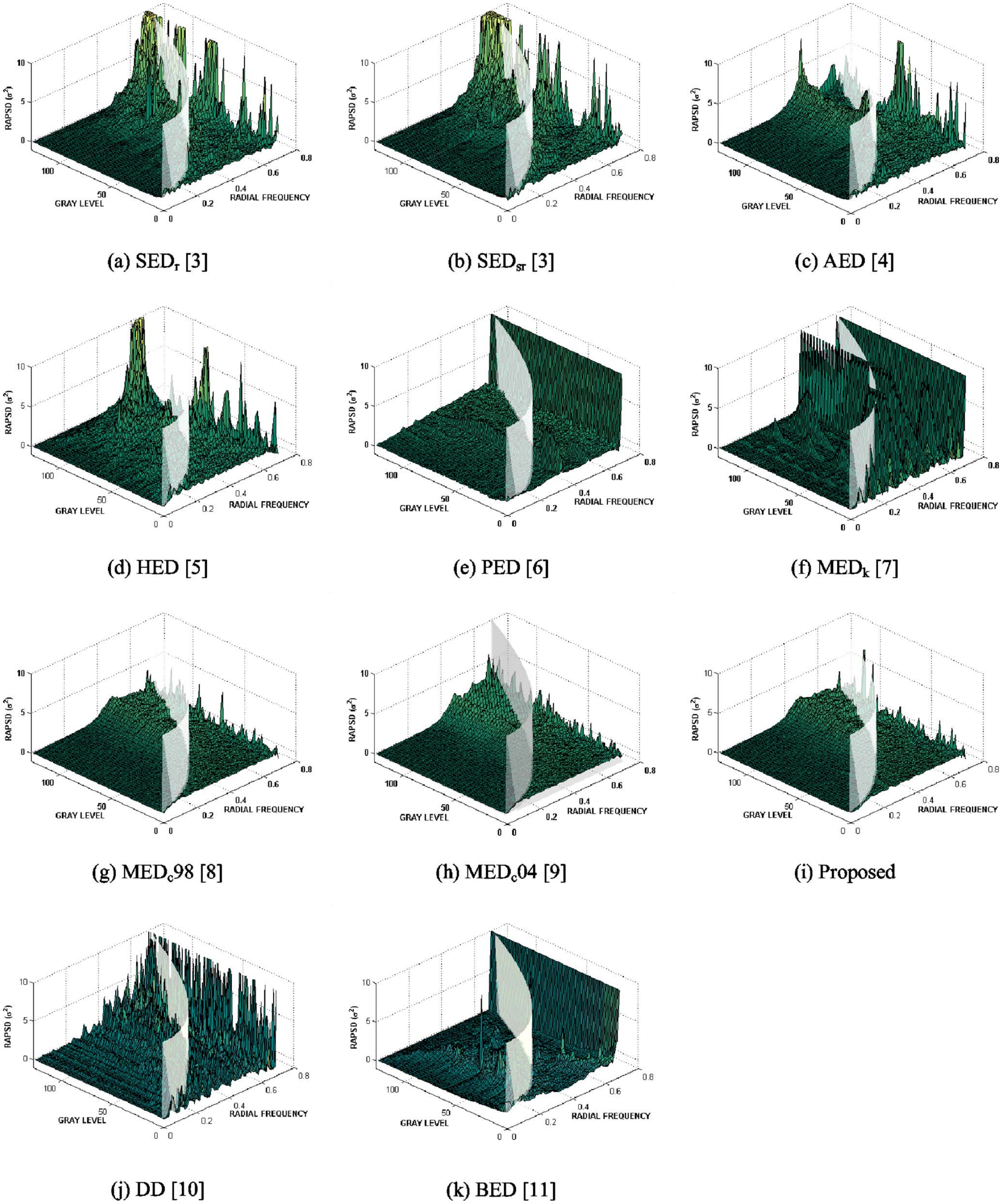
(h) MED$_c$04 [9]

(i) Proposed

(j) DD [10]

(k) BED [11]

Fig. 7 Performance in terms of anisotropy for the (a) SED$_r$, (b) SED$_{sr}$, (c) AED, (d) HED, (e) PED, (f) MED$_k$, (g) MED$_c$98, (h) MED$_c$04, (i) proposed, (j) DD, and (k) BED algorithms.



(a) SED$_r$ [3]

(b) SED$_{sr}$ [3]

(c) AED [4]

(d) HED [5]

(e) PED [6]

(f) MED$_k$ [7]

(g) MED$_c$98 [8]

(h) MED$_c$04 [9]

(i) Proposed

(j) DD [10]

(k) BED [11]

Fig. 8 Performance in terms of RAPSD for the (a) SED$_r$, (b) SED$_{sr}$, (c) AED, (d) HED, (e) PED, (f) MED$_k$, (g) MED$_c$98, (h) MED$_c$04, (i) proposed, (j) DD, and (k) BED algorithms.

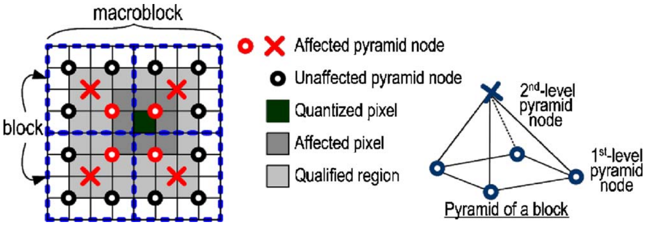


Fig. 9 Elements to be updated in a pyramid in the worst case.

proposed algorithms are better than the other algorithms. Its corresponding anisotropy is well below 0 dB in all combinations of gray levels and radial frequencies.

Blue noise halftoning is characterized by a distribution of binary pixels where the minority dots are spread as homogeneously as possible.[1] It is visually pleasant as it does not clash with the structure of an image. Pixels distributed in this way create an aperiodic and isotropic pattern and it does not contain any low-frequency spectral components.

Figure 8 shows the performance of various algorithms in terms of RAPSD. For easier comparison, the range of RAPSD shown in all these plots is bounded to be less than 10. If a RAPSD value is larger than 10, it is clipped and the clipped value is displayed in the plots.

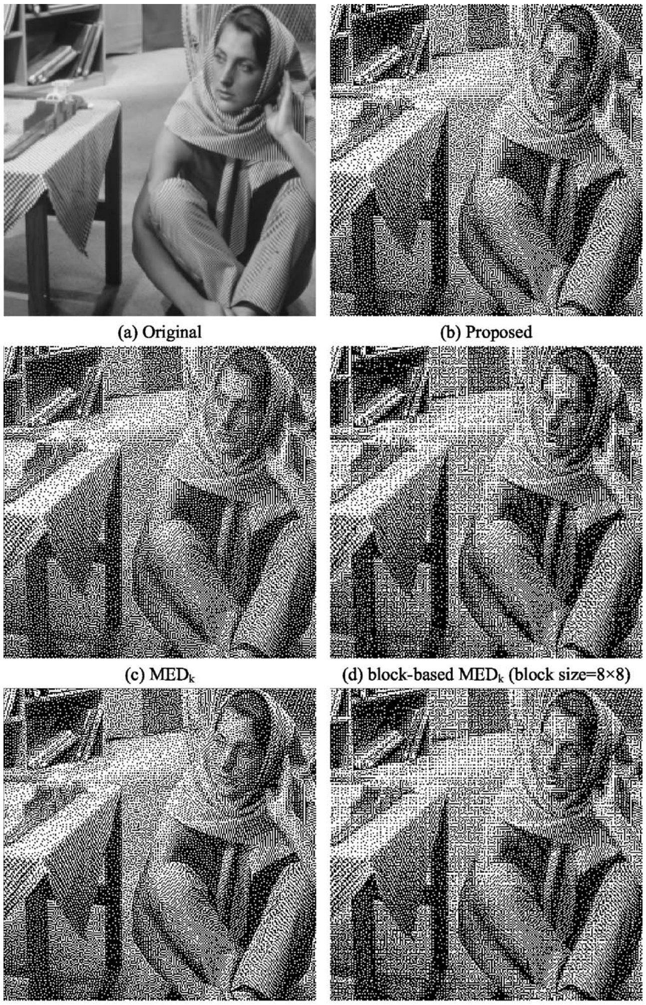
A good blue noise generator should produce a result that carries little or no low-frequency spectral components. The result should also provide a flat high-frequency spectral region and a spectral peak at blue noise principal frequency f_b. To provide a clear picture of the performance of the algorithm, a white surface that marks the principal frequency f_b for a particular gray level is added in each of the plots as a reference for comparison. Figures 8(g)–8(i) show that the outputs of MED$_c$98, MED$_c$04, and the proposed algorithms have all these features. The harmonics that appear in the plot shown in Fig. 8(f) explain why there are so many pattern artifacts in the outputs of MED$_k$.

3.2 *Computational Complexity Analysis*

In this part of analysis, the computational complexity of MED algorithms is provided based on an assumption that the input image is of size $N \times N$, where N is a multiple of 4.

For the proposed algorithm, at the initial stage, N^2 additions are required to construct $N^2/16$ intensity pyramids and determine the intensity level of minority dots. The realization complexity for the steps left behind is then proportional to the number of minority dots ($\leq N^2/2$) to be settled in the output.

To settle a dot, all involved operations are confined in a macroblock. First of all, nine comparisons are required to locate the most wanted pixel. If it is in the qualified region of a macroblock, the searching effort will not be wasted. Since there are 36 pixels in the qualified region of an 8×8 macroblock, a reasonable estimate of the hit ratio is 36/64 though the real hit ratio is higher than this in our simulation. Accordingly, on average, the effort for locating a qualified pixel is $9(64/36)=16$ comparisons. Two multiplications and nine additions are required to diffuse the quantization error. Finally, at most 12 additions are required to update the affected blocks in the macroblock. This extreme case happens when all four blocks are affected, as shown in Fig. 9. By considering comparison as addition, the upper bound of the complexity is $2N^2/2$ multiplications and $N^2+(9+16+12)N^2/2$ additions, which implies at most one multiplication and 19.5 additions per pixel.


(a) Original

(b) Proposed

(c) MED$_k$

(d) block-based MED$_k$ (block size=8×8)

(e) MED$_c$98

(f) block-based MED$_c$98 (block size=8×8)

Fig. 10 Halftones produced with various MED algorithms.

The initialization stage of MED$_k$ takes N^2-1 additions. For each introduced dot, it takes $3\log_2 N$ comparisons to locate the most wanted pixel, nine additions, and two multiplications to diffuse the quantization error and at most $4\log_2 N$ additions to update the intensity pyramid. Unlike our proposed algorithm, MED$_k$ introduces white dots instead of minority dots and hence the number of introduced dots is bounded by N^2 instead of $N^2/2$. The upper bound of the complexity is $2N^2$ multiplications and $N^2-1+(7\log_2 N+9)N^2$ additions, which implies at most two multiplications and $7\log_2 N+10$ additions per pixel. Its complexity bound per pixel is $O(\log_2 N)$, while that of the proposed algorithm is a constant.

Since block overlapping and block shifting are, respectively, used in MED$_c$98 and MED$_c$04 to remove blocking effect, the structure of the intensity pyramids involved is more complicated as compared with that used in MED$_k$. Accordingly, their realization complexity is even higher. In particular, the complexity bound of MED$_k$04 is roughly three-fold of that of MED$_k$.

Table 3 Visual quality of halftones produced with different algorithms in terms of (a) MSE_v and (b) MSE_v of boundary pixels and interior pixels.

Image Size	Image	(a) $\text{MSE}_v \times 10^{-3}$			(b) MSE_v (boundary pixels and interior pixels)$\times 10^{-3}$					
		MED_k	MED_c98	Proposed	MED_k		MED_c98		Proposed	
					Boundary Pixels	Interior Pixels	Boundary Pixels	Interior Pixels	Boundary Pixels	Interior Pixels
256×256	“Baboon”	0.0688	0.0198	0.0251	0.0689	0.0687	0.0199	0.0197	0.0254	0.0249
	“Barb”	0.1306	0.0196	0.0286	0.1313	0.1301	0.0196	0.0196	0.0288	0.0284
	“Boat”	0.2328	0.0212	0.0253	0.2333	0.2324	0.0214	0.0211	0.0258	0.0249
	“Lena”	0.1279	0.0209	0.0272	0.1276	0.1281	0.0209	0.0208	0.0272	0.0273
	“Peppers”	0.2723	0.0189	0.0260	0.2717	0.2728	0.0188	0.0189	0.0262	0.0259
	Average	**0.1665**	**0.0201**	**0.0264**	**0.1666**	**0.1664**	**0.0201**	**0.0200**	**0.0267**	**0.0263**
512×512	“Baboon”	0.0927	0.0195	0.0262	0.0929	0.0926	0.0196	0.0194	0.0263	0.0261
	“Barb”	0.1987	0.0193	0.0285	0.1991	0.1984	0.0195	0.0192	0.0290	0.0282
	“Boat”	0.2804	0.0202	0.0276	0.2805	0.2803	0.0203	0.0202	0.0280	0.0273
	“Lena”	0.1688	0.0203	0.0281	0.1689	0.1687	0.0206	0.0201	0.0284	0.0278
	“Peppers”	0.3938	0.0193	0.0283	0.3936	0.3940	0.0195	0.0191	0.0286	0.0281
	Average	**0.2269**	**0.0197**	**0.0277**	**0.2270**	**0.2268**	**0.0199**	**0.0196**	**0.0281**	**0.0275**

Table 4 Quality measurement of halftones produced with different algorithms in terms of (a) WSNR, (b) LDM, and (c) UQI.

Image Size	Image	(a) WSNR			(b) LDM			(c) UQI		
		MED_k	MED_c98	Proposed	MED_k	MED_c98	Proposed	MED_k	MED_c98	Proposed
256×256	"Baboon"	24.7720	24.6042	24.5954	0.9139	0.9158	0.9160	0.1641	0.1826	0.1770
	"Barb"	23.8069	24.4193	24.3880	0.9361	0.9387	0.9378	0.1705	0.1907	0.1870
	"Boat"	24.2005	24.8692	25.1572	0.9218	0.9229	0.9232	0.1388	0.1576	0.1528
	"Lena"	24.3110	24.7865	24.7064	0.9267	0.9288	0.9281	0.1253	0.1403	0.1374
	"Peppers"	22.0826	23.7100	23.6978	0.9549	0.9518	0.9511	0.1483	0.1745	0.1724
	Average	**23.8346**	**24.4778**	**24.5090**	**0.9307**	**0.9316**	**0.9312**	**0.1494**	**0.1691**	**0.1653**
512×512	"Baboon"	24.7605	24.4745	24.6803	0.9583	0.9593	0.9593	0.1791	0.2002	0.1926
	"Barb"	23.6959	24.4917	24.4763	0.9591	0.9607	0.9602	0.1445	0.1634	0.1583
	"Boat"	24.2644	24.9879	25.3348	0.9539	0.9541	0.9543	0.0976	0.1114	0.1093
	"Lena"	24.3285	24.9699	24.9251	0.9532	0.9543	0.9539	0.0794	0.0901	0.0878
	"Peppers"	22.1165	24.0698	23.9476	0.9705	0.9666	0.9660	0.0845	0.1043	0.1022
	Average	**23.8332**	**24.5988**	**24.6728**	**0.9590**	**0.9590**	**0.9587**	**0.1170**	**0.1339**	**0.1300**

4 Simulation Results

The analysis presented in the previous section proves the theoretical advantage of MED in different measures. In practice, simulation results on real images also reveal this fact when comparing MED with conventional error diffusion algorithms.[7–9] Accordingly, this section does not put its focus on the comparison between MED and conventional algorithms again. Instead, the focuses are on (1) the complexity of different MED algorithms in their practical realization, (2) whether there is a drop in the visual quality of the output of the proposed algorithm as compared with those of other MED algorithms when real images are processed, (3) whether the proposed algorithm introduces blocking artifacts in its output, and (4) what happen if one partitions an input image into a number of 8×8 blocks and then directly applies MED_k or MED_c98 to each block so as to support parallel processing in a straightforward manner. To address these issues, simulation was carried out to evaluate the performance of different MED algorithms and their variants on a set of *de facto* standard 8-bit gray-scale images including "Baboon," "Barb," "Boat," "Lena," and "Peppers."

Table 2 shows the average number of additions (ADD), comparisons (CMP), and multiplications (MUL) required per pixel to produce the halftones with different MED algorithms in the simulation. It shows that the proposed algorithm can remarkably reduce the number of operations as compared with conventional MED algorithms. On average, when the input image is of size 512×512, the complexity of the proposed algorithm is only 59% of MED_k and 34% of MED_c98 in terms of total number of operations per pixel. Note that while the other MED algorithms do not support parallel processing, the proposed algorithm does and it can further reduce the processing time significantly.

Halftone visibility metrics[2] can be used to measure the distortion observed by a human viewer between an original gray-scale image **X** and its binary halftone **B**. In particular, it is defined as

$$\mathrm{MSE}_v = \frac{1}{N \times N} \| \mathrm{hvs}(\mathrm{X}, \mathrm{vd}, \mathrm{dpi}) - \mathrm{hvs}(\mathrm{B}, \mathrm{vd}, \mathrm{dpi}) \|^2, \tag{7}$$

where hvs is the human visual system (HVS) filter function defined in Ref. 2, vd is the viewing distance in inches, and dpi is the printer resolution. In our simulations, the viewing distance was fixed at 20 in. and printer resolution of 600 dpi was considered. Table 3a shows the MSE_v results of the evaluated MED algorithms. One can see that MED_c98 provides the best result and the performance of the proposed algorithm is very close to it. The proposed algorithm reduces the complexity at no cost of the image quality.

To explore whether the proposed algorithm introduces more artifacts to the boundary region of a block, block boundary pixels and block interior pixels of a HVS-filtered halftone were separated and their contribution to MSE_v was evaluated individually. As shown in Table 3b, the difference between their contributions is very small.

Table 4 shows the performance of the MED algorithms in terms of weighted SNR[12] (WSNR), linear distortion measure[13] (LDM), and universal objective image quality index[14] (UQI). WSNR uses the contrast sensitivity function[15] (CSF) of the HVS to measure the distortion of halftone image while LDM is used to measure the linear distortion. UQI is an index to qualify an image. In terms of all these measurements, the performance of the proposed algorithm is more or less the same as that of MED$_c$98.

Figure 10 shows the halftone outputs of various MED algorithms for subjective evaluation. As mentioned earlier, one can divide an input image into a number of 8×8 blocks and then applies either MED$_k$ or MED$_c$98 to each block independently. This approach turns a frame-based MED algorithm into a block-based algorithm and makes parallel processing possible immediately. However, this straightforward approach does not work. As shown in Figs. 10(d) and 10(f), serious blocking artifacts are visible in their outputs. In contrast, as shown in Fig. 10(b), there is no visible blocking artifact in the result of the proposed algorithm and its visual quality is subjectively very close to that of MED$_c$98.

5 Conclusions

A fast MED algorithm for digital halftoning was proposed and a detailed analysis on various MED algorithms was presented. Analysis results show that, like other MED algorithms, the proposed algorithms can provide a better performance as compared with conventional error diffusion algorithms in terms of the directional distribution of dots, anisotropy, and blue noise characteristic, while its computational complexity is significantly reduced as compared with conventional MED algorithms. As the proposed algorithm supports parallel processing, processing time can further be reduced to enable real-time processing. Simulation results also demonstrated that, in practical applications, the proposed algorithm could reduce the computational complexity without sacrificing the image quality of its output.

Acknowledgments

This work was supported by a grant from the research Grants Council of the Hong Kong Special Administrative Region (PolyU Grant A-PA3U) and a grant from The Hong Kong Polytechnic University (POLYU Grant A-PA3U).

References

1. R. A. Ulichney, *Digital Halftoning*, MIT Press, Cambridge, MA (1987).
2. D. L. Lau and G. R. Arce, *Modern Digital Halftoning*, Marcel Dekker, New York (2001).
3. R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial gray scale," in *Proc. SID Int. Symp. Digest of Technical Papers*, *Society for Information Displays*, 36–37 (1975).
4. P. W. Wong, "Adaptive error diffusion and its application in multiresolution rendering," *IEEE Trans. Image Process.* **5**(7), 1184–1196 (1996).
5. T. Nagae, T. Agui, and H. Nagahashi, "A generalization of the Peano scan and its application to halftoning," *ITEJ Tech. Rep.* **16**, 25–30 (1992).
6. E. Peli, "Multiresolution, error-convergence halftone algorithm," *J. Opt. Soc. Am. A* **8**(4), 625–633 (1991).
7. I. Katsavounidis and C. C. J. Kuo, "A multiscale error diffusion technique for digital halftoning," *IEEE Trans. Image Process.*, **6**(3), 483–490 (1997).
8. Y. H. Chan, "A modified multiscale error diffusion technique for digital halftoning," *IEEE Signal Process. Lett.* **5**(11), 277–280 (1998).
9. Y. H. Chan and S. M. Cheung, "Feature-preserving multiscale error diffusion for digital halftoning," *J. Electron. Imaging* **13**(3) 639–645 (2004).
10. M. Mese and P. P. Vaidyanathan, "Optimed halftoning using dot diffusion and methods for inverse halftoning," *IEEE Trans. Image Process.* **9**(4), 691–708 (2000).
11. N. Damera-Venkata, J. Yen, V. Monga, and B. L. Evans, "Hardcopy image barcodes via block error diffusion," *IEEE Trans. Image Process.* **14**(12), 1977–1989 (2005).
12. T. D. Kite, B. L. Evans, and A. C. Bovik, "Modeling and quality assessment of halftoning by error diffusion," *IEEE Trans. Image Process.* **9**(5), 909–922 (2000).
13. M. Valliapaan, B. L. Evans, D. A. D. Tompkins, and F. Kossentini, "Lossy compression of stochastic halftones with JBIG2," in *Proc. IEEE Int. Conf. on Image Processing*, Vol. 1, pp. 214–218 (1999).
14. Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.* **9**(3), 81–84 (2001).
15. T. Mitsa and K. Varkur, "Evaluation of contrast sensitivity functions for the formulation of quality measures incorporated in halftoning algorithms," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Vol. 5, pp. 301–304 (1993).

Yik-Hing Fung received his BEng degree with honors and his PhD degree in electronic and information engineering in 2000 and 2006, respectively, from The Hong Kong Polytechnic University, Hong Kong, China. Between 2000 and 2001, he was an engineer in The Hong Kong Polytechnic University for a project entitled Web Based Portable Monitoring and Surveillance System. His research interests include image restoration, digital image halftoning, and image compression.

Ka-Chun Lui received his BEng degree with honors in 2001 from The Hong Kong Polytechnic University, Hong Kong, China. He is currently pursuing his MPhil degree at the same university. His research interests include image restoration and image compression.

Yuk-Hee Chan received his BSc degree with honors in electronics from the Chinese University of Hong Kong in 1987, and his PhD degree in signal processing from The Hong Kong Polytechnic University in 1992. Between 1987 and 1989, he worked as a research and development engineer at Elec & Eltek Group, Hong Kong. He joined The Hong Kong Polytechnic University in 1992 and is now an associate professor in the Department of Electronic & Information Engineering. He has published over 110 research papers in various international journals and conferences. His research interests include image and video compression, image restoration, and fast computational algorithms in digital signal processing. Dr. Chan is a member of IEEE and IEE. He was the Chairman of the IEEE Hong Kong Joint Chapter of Circuits and Systems and Communications in 2003-04.