Removing harmonic distortion in tiled screens

Xiao Chen and Yuk Hee Chan
The Hong Kong Polytechnic University, Hong Kong SAR
E-mail: enyhchan@polyu.edu.hk

Abstract - **When producing halftones with the screening technique, tiling is generally used to construct a screen of appropriate size with screens of small fixed size. Regular tiling and random tiling are two common tiling schemes used nowadays. The former scheme produces periodic screens which introduce harmonic distortion to the screening outputs while the latter may introduce blocking artifacts as the possible connections among adjacent blocks are too many for a training algorithm to train the small screens that are used for tiling. This paper addresses this issue and proposes a new tiling scheme that allows one to train small screens to construct large screens seamlessly. Simulation results show that a screen constructed with the proposed tiling screen bears blue noise characteristics, contains no blocking artifacts and little harmonic distortion.**

I. INTRODUCTION

Digital Halftoning is a process of transforming a continuous tone image to a bi-level image [1]. Among the various digital halftoning techniques, stochastic screening is the approach of the lowest complexity as it is basically a thresholding process in which an image is compared with a predefined threshold plane of identical size to produce a binary output that reflects the comparison result of each pixel.

The predefined threshold plane is generally referred to as a screen. According to the nature of its output, a screen can be classified as either an AM, a FM or a hybrid screen. AM and FM screens tend to produce clustered and dispersed dots in the screening results respectively. In general, clustered dots are more robust to printer artifacts while dispersed dots can provide outputs of better visual quality. A hybrid screen tries to take advantages from both by using dispersed and clustered dots, respectively, to render highlights and midtones.

Huge effort has been devoted to develop screens for different purposes. As examples, genetic algorithm, void-and-cluster, direct binary search (DBS) and neural network have all been exploited by various researchers to develop blue noise screens of blue noise characteristics [2-11]. Studies on screens used for different sampling lattices and different screen periodicities were also conducted [12-15]. Recently, researchers have shifted their interests to develop hybrid screens based on the concept of supercell [16,17].

In ideal situation, for an image of specific size, a dedicated screen with matched size should be developed for screening. However, in practice, we generally design a screen of fixed size (e.g. 64×64) and tilt it to construct screens of other sizes when necessary. For reference purpose, the fixed-size screen used to construct a bigger screen is referred to as a *tile screen* hereafter in this paper.

Tiling can introduce blocking artifacts to a screening output if the spatial inter-correlation among adjacent tile screens in a constructed screen is not taken into account when designing the tile screens. These blocking artifacts appear as some abrupt transition across the boundary of two blocks in a screening output. The current solution is to train a screen constructed with tile screens instead of a single tile screen such that the cross-boundary smoothness can be taken care of when developing the tile screen.

Tiling unavoidably introduces some harmonic distortion to the screening output when the constructed screen is periodic. To solve this problem, one can tile different tile screens randomly to break the periodicity[18]. However, this approach complicates the training process of the tile screens. As tile screens can be connected randomly, there can be a number of possible connections and the training process may not be able to take care of all of them especially when the involved tile screens are too many.

This work addresses these issues and proposes a solution to remove the blocking artifacts and the harmonic distortion that are commonly found in the outputs of conventional screening algorithms. Specifically, we proposed a tiling method to construct aperiodic screens and a DBS-based screen developing algorithm to produce tile screens that can be tiled seamlessly with the proposed tiling method.

This paper is organized as follows. A brief review on a conventional DBS-based screen training algorithm is given in Section II. A tiling method and a corresponding algorithm for producing tiling screens to support the tiling method are proposed in Sections III and IV respectively. Simulation results are provided in Section V for evaluation. A brief conclusion is given at the end.

II. REVIEW ON DBS-BASED SCREEN TRAINING ALGORITHM

In conventional realization, we train a periodic screen of size $S_t \times S_t = kS_s \times kS_s$ with constant patches, where k is an integer and S_s is the period along each dimension. A $S_s \times S_s$ segment of the resultant periodic screen is then extracted to form the tile screen. This allows the training algorithm to minimize the perceived error in the boundary regions of connected tile screens and hence helps to remove the blocking artifacts of a screen constructed with the trained tile screen in the future.

In general, to develop a screen, one can produce a set of binary halftones, one for a constant patch of each possible gray level, under a stacking constraint. Assume that there are M possible gray levels in an image and we process constant patches from the one of the lowest gray level and up to the one of the highest. Figure 1 shows how the constant patch of the l^{th} darkest level, I_l, is processed under the stacking constraint with DBS to produce its corresponding binary halftone B_l after I_{l-1} was processed to get B_{l-1}, where $l \in \{1,2\ldots M\text{-}1\}$. Here, we assume that the maximum and the minimum intensity values of an image is 1 and 0 respectively. Hence, we have $I_l(x,y) = l/(M-1)$ for all (x,y), where $Y(x,y)$ denotes the intensity value of the $(x,y)^{th}$ pixel of image *Y*. Accordingly, we have $I_0(x,y) = B_0(x,y) = 0$ and $I_{M-1}(x,y) = B_{M-1}(x,y) = 1$ for all (x,y).

The process iteratively refines a binary halftone, say *B*, to minimize its perceived error from I_l. Specifically, the perceived error is defined as

$$E_l(B) = \| H \otimes (B - I_l) \|^2 \qquad (2)$$

where *H* is a human visual system (HVS) filter, $\otimes$ is the circular convolution operator and $\|Y\|^2$ denote the L^2 norm of image *Y*.

Both I_l and B_l are of size $kS_s \times kS_s$. The perceived error in the boundary region of adjacent blocks is taken into account during the training process. In consequence, the blocking artifacts can be reduced to a certain extent.

After producing halftones B_l for l=0,1,2…M-1, the periodic screen of size $kS_s \times kS_s$ can be obtained with

$$S(x,y) = 1 - \frac{1}{M-1} \sum_{l=0}^{M-1} B_l(x,y) \qquad (3)$$

and a tile screen of size $S_s \times S_s$ is extracted by chopping a period from the periodic screen.

This conventional screen training algorithm can be extended to train any screen constructed with different tile screens. Specifically, one can train a large screen which covers all possible neighboring combinations of the target tile screens. In theory, as the training process minimizes the perceived error in the boundary region of adjacent blocks, the developed tile screens can be connected seamlessly. This approach works well when regular tiling schemes are used, but regular tiling produces periodic screens and there will be harmonic distortion in its screening outputs. When random tiling is used, the training algorithm can break down because the possible number of neighboring combinations increases exponentially with the number of tile screens. The size of the screen to be trained is huge and the increased constraints may counteract with each other to make the training ineffective.

III. Proposed Tiling Method

To make a balance between random tiling and regular tiling, the proposed tiling method allows one to tile a screen with 18 tile screens of equal size. The tile screens are divided into two groups. Group A contains 2 tile screens while Group B contains 16 tile screens. When constructing a screen with the tile screens, the screen is treated as a checkerboard each cell of which is of the same size as a tile screen. Group A tile screens are first randomly assigned to the black checkerboard cells. Group B tile screens are then assigned to the white checkerboard cells based on their 4-connected neighbors as shown in Figure 2, where $T_{A(i)}$ and $T_{B(j)}$ are, respectively, the i^{th} tile screens in Group A and the j^{th} tile screens in Group B.

Partition B_{l-1} to get $B_{l-1,tile} = B_{l-1}(0:S_s-1, 0:S_s-1)$.

Toggle $d = round\left(\frac{l \times S_s \times S_s}{M-1}\right) - round\left(\frac{(l-1) \times S_s \times S_s}{M-1}\right)$ pixels in $B_{l-1,tile}$ whose values are 0 to 1 to maximize the reduction in perceived error.

Tile $B_{l-1,tile}$ to construct an initial B_l.

Do
  For $x = 0: S_s\text{-}1$
  For $y = 0: S_s\text{-}1$
    $B_l^+ = B_l$
    For $m \in [x-w, x+w]$
    For $n \in [y-w, y+w]$
      $m' = ((m + S_s) \bmod S_s)$
      $n' = ((n + S_s) \bmod S_s)$
      If $B_l(x,y) \neq B_l(m',n')$ and $B_{l-1}(x,y) = B_{l-1}(m',n') = 0$
        $B_l^{'} = B_l$
        For $i, j \in \{0,1..k\text{-}1\}$
          Swap $B_l^{'}(x+iS_s, y+jS_s)$ and $B_l^{'}(m'+iS_s, n'+jS_s)$
        End
        If $E_l(B_l^{'}) < E_l(B_l^{+})$, $B_l^+ = B_l^{'}$, **End**
      End
    End End
    $B_l = B_l^+$
  End End
While B_l was updated in the last iteration

Fig.1 Pseudo code for producing B_l with B_{l-1} in conventional approach

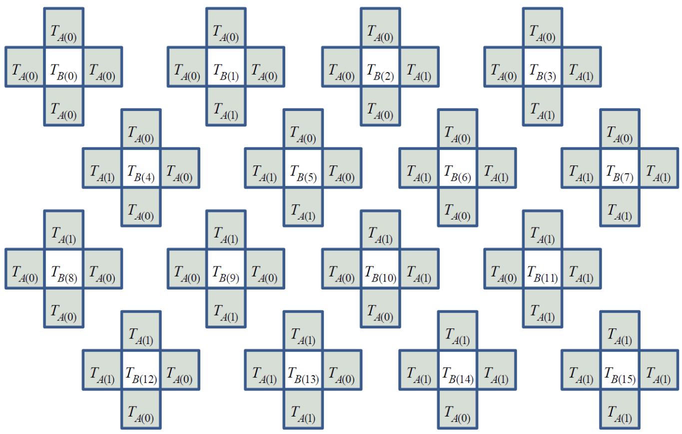

Fig.2 The 16 possible neighboring combinations of a Group B tile in a checkerboard pattern

In formulation, the tile screen assigned to a specific white checkerboard cell is selected as

$$T_{B(t)} = T_{B(8a+4b+2c+d)} \tag{4}$$

when the Group A tile screens assigned to the upper, the left, the right and the lower neighboring checkerboard cells of the white checkerboard cell are $T_{A(a)}$, $T_{A(b)}$, $T_{A(c)}$ and $T_{A(d)}$ respectively. Note that a, b c and d can only be either 0 or 1. Hence, t is bounded in [0,15].

Since there are only two tile screens in $\Phi_A=\{T_{A(i)}|i=0,1\}$, there are 16 possible direct neighboring combinations for a Group B tile as shown in Figure 2. In such an arrangement, the 16 Group B tile screens can be tailor-made to handle the 16 neighboring combinations individually such that tile screens from different groups can put together seamlessly.

To support the proposed hybrid tiling method, a tile screen developing algorithm is presented in next section to produce the required tile screens.

IV. Screen Training Algorithm

In the proposed tiling method, Group A tile screens $T_{A(0)}$ and $T_{A(1)}$ are assigned randomly to black checkerboard cells first without any neighboring constraint and they cannot be neighbors to each other. Hence, Group A tile screens can be developed individually with any screen developing algorithms without taking care of the inter-tile correlation.

After $T_{A(0)}$ and $T_{A(1)}$ are developed, Group B tile screens are developed independently. Suppose we are now developing tile screen $T_{B(t)}$ and the size of a tile screen is $S_s \times S_s$. To develop $T_{B(t)}$, we need to produce halftones for constant patches of different gray levels. Let I_l^* be the $S_s \times S_s$ constant patch whose gray level is the l^{th} darkest gray level. Starting from I_0^*, we gradually produce $B_{l,B(t)}$, binary halftones of I_l^*, from $l = 0$ to $l = M$-1 based on $B_{l,A(1)}$, $B_{l,A(0)}$ and $B_{l-1,B(t)}$ as shown in Figure 3. Note that $B_{l,A(1)}$ and $B_{l,A(0)}$ are also binary halftones of I_l^*. They are obtained by screening I_l^* with tile screens $T_{A(1)}$ and $T_{A(0)}$ respectively. Another remark we would like to make is that we have $B_{0,B(t)}(x,y)=0$ and $B_{M-1,B(t)}(x,y)=1$ for all (x,y) and t since they are the halftones for the darkest and the brightest constant patches respectively.

In Figure 3, $B_{l,B(t)}^{\circ}$ is an initial version of $B_{l,B(t)}$. It is obtained by randomly picking d= $round((S_s \times S_s)l/(M-1))$ - $round((S_s \times S_s)(l-1)/(M-1))$ pixels whose pixel values are 0 in $B_{l-1,B(t)}$ and toggling their values to 1. When refining

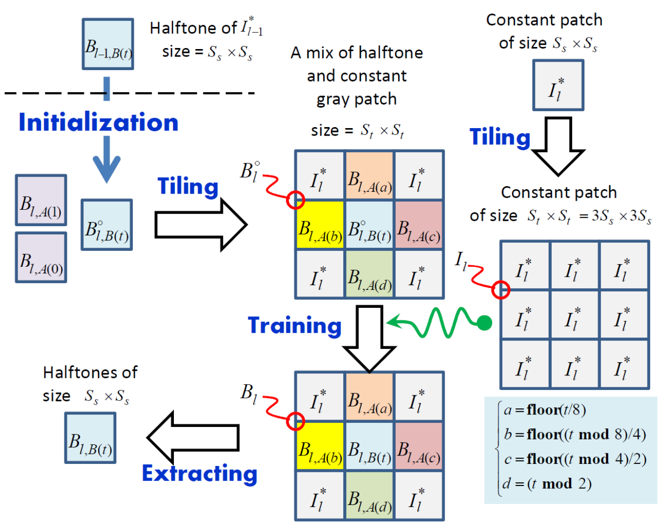

Fig.3 Procedures for producing $B_{l,B(t)}$ from $B_{l,A(0)}$, $B_{l,A(1)}$ and $B_{l-1,B(t)}$ with the approach

Tile $B_{l,A(1)}$, $B_{l,A(0)}$, $B_{l,B(t)}^{\circ}$ and I_l^* to construct an initialized B_l°.

$B_l' = B_l^{\circ}$

Do

For $x = 0 : S_s$-1

For $y = 0 : S_s$-1

$B_l^+ = B_l'$

For $m = \max(0, x-w) : \min(S_s-1, x+w)$

For $n = \max(0, y-w) : \min(S_s-1, y+w)$

If $\{ B_l'(S_s+x, S_s+y) \neq B_l'(S_s+m, S_s+n)$ and $B_{l-1,B(t)}(x,y) = B_{l-1,B(t)}(m,n) = 0 \}$

$B_l'' = B_l'$

Swap $B_l''(S_s+x, S_s+y)$ and $B_l''(S_s+m, S_s+n)$

If $\|H \otimes (B_l'' - I_l)\|^2 < \|H \otimes (B_l^+ - I_l)\|^2$, $B_l^+ = B_l''$, **End**

End

End End

$B_l' = B_l^+$

End End

While B_l' was updated in the last iteration

$B_{l,B(t)} = B_l'(S_s : 2S_s-1, S_s : 2S_s-1)$

Fig. 4 Pseudo code for refining B_l in the proposed scheme

$B_{l,B(t)}^{\circ}$ to produce $B_{l,B(t)}$ with DBS, we try to minimize the perceived error between B_l° and I_l, a constant patch of size $3S_s \times 3S_s$. B_l° is an image obtained by tiling $B_{l,A(1)}$, $B_{l,A(0)}$,

$B^{\circ}_{l,B(t)}$ and I^*_l as shown in Figure 3. The four direct neighbors of $B^{\circ}_{l,B(t)}$ in B°_l, namely, $B^{\circ}_{l,B(a)}$, $B^{\circ}_{l,B(b)}$, $B^{\circ}_{l,B(c)}$ and $B^{\circ}_{l,B(d)}$, are determined by

$$\begin{cases} a = \text{floor}(t/8) \\ b = \text{floor}((t \bmod 8)/4) \\ c = \text{floor}((t \bmod 4)/2) \\ d = (t \bmod 2) \end{cases} \quad (5)$$

The refinement is done under a stacking constraint until no further refinement can be made. Figure 4 shows the details of how to get $B_{l,B(t)}$ based on $B_{l,A(1)}$, $B_{l,A(0)}$ and $B_{l-1,B(t)}$ in pseudo code. The inter-tile correlation is taken care of by minimizing the perceived error in the boundary regions of the tiles.

Once $B_{l,B(t)}$ for all $l \in \{0,1\ldots M\text{-}1\}$ are obtained, tile screen $T_{B(t)}$ can be determined as

$$T_{B(t)}(x,y) = 1 - \frac{1}{M-1}\sum_{l=0}^{M-1} B_{l,B(t)}(x,y) \quad (6)$$

After all tile screens in Group B are produced, tile screens in both Group A and Group B are stored and they are ready for constructing a tile screen of any desirable size with the proposed hybrid tiling method in the future.

V. SIMULATION RESULTS

Simulations were carried out to evaluate the performance of the proposed screen tiling scheme. For comparison, the performance of random tiling and regular tiling were also evaluated. When random tiling scheme is used, 18 tile screens are trained with the approach used in [18]. When regular tiling scheme is used, only one tile screen is trained with the conventional DBS-based algorithm reviewed in Section II. The tile screens used to support the proposed tiling scheme are trained with the approach discussed in Section IV. All tile screens are of size 32×32.

Figure 5 shows some 128×128 screens that were constructed with different tiling schemes and their corresponding spectra. From their spectra, one can see that there is strong harmonic distortion in the screen constructed with regular tiling while there is little in the others. In fact, Figs. 5(b) and (c) show that the screens constructed with random tiling and our proposed tiling scheme bear blue noise characteristics. The low frequency noise is weak while the high frequency noise is strong. It is advantageous as our eyes behave as a lowpass filter that can effectively remove the high frequency noise in their corresponding screening outputs. Note that the screening outputs of a blue noise screen bear blue noise characteristics as well.

One interesting observation is that there are blocking artifacts in the screen constructed with random tiling as shown in Fig. 5(b). When training the tile screens with the algorithm proposed in [18], tile screens are trained sequentially and the algorithm only tries to smooth the connection between the tile screen being trained and the tile screen that is most recently trained. There is no guarantee that two randomly selected trained tile screens can be smoothly connected to each other. This explains why there can be

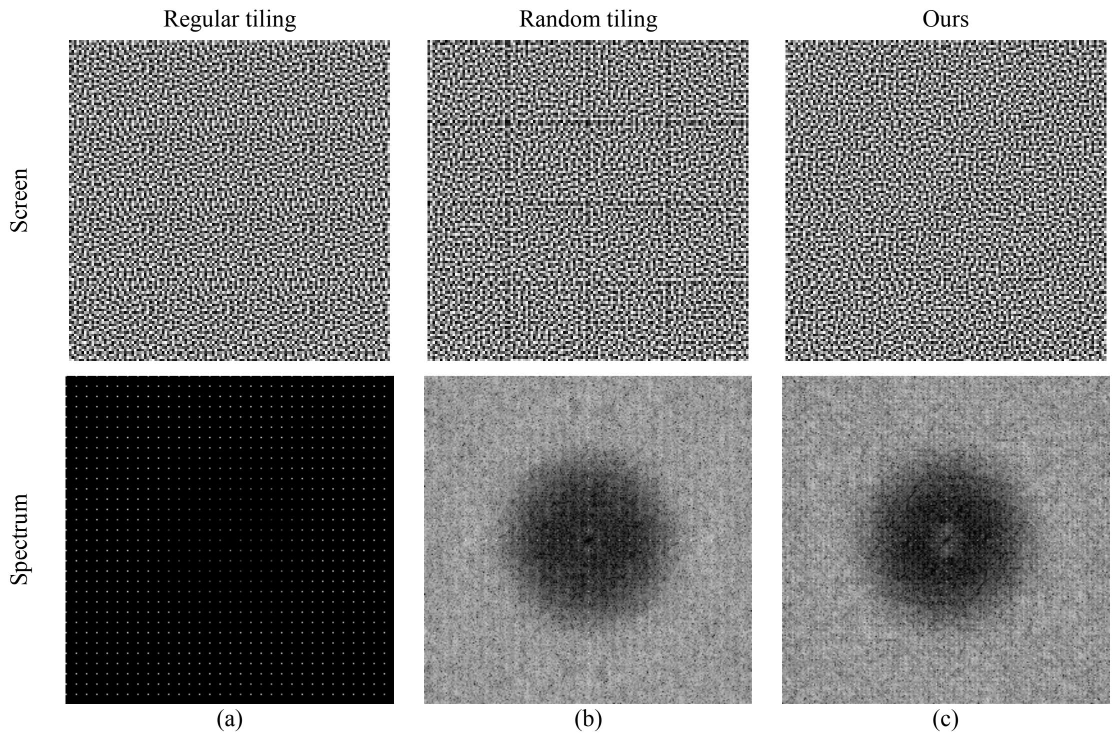

Fig. 5. Screens constructed with different tiling methods: (a) periodic tiling, (b) random tiling and (c) the proposed tiling schemes.

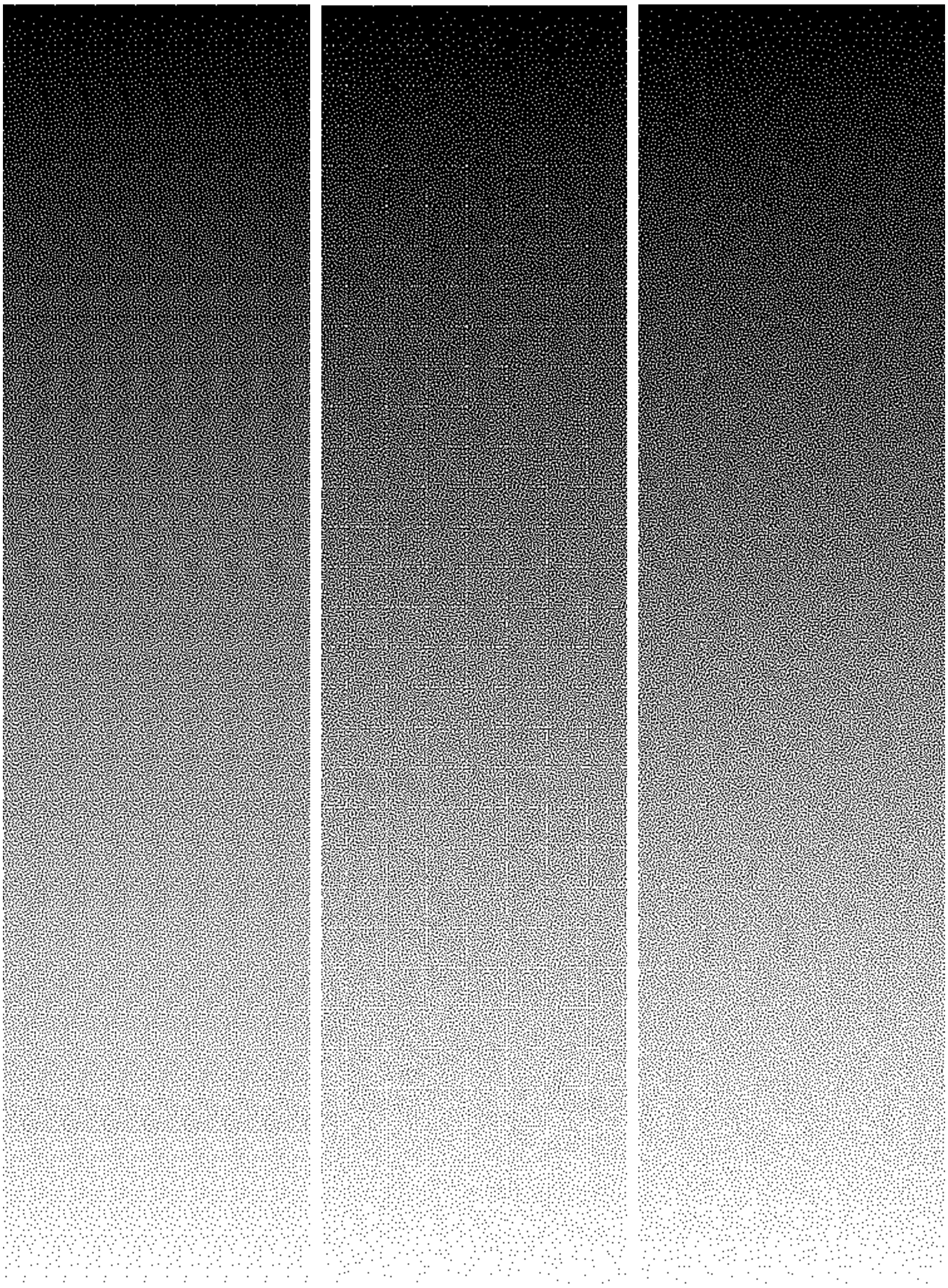

Fig. 6 Screening outputs of a ramp using screens constructed with different tiling methods

blocking artifacts in Fig. 5(b). In contrast, the screen constructed with the proposed scheme contains little harmonic distortion and no blocking artifacts.

Fig. 6 shows the screening outputs of a ramp input with the screens constructed with different tiling schemes. One can see that Fig. 6(a) is periodic and Fig. 6(b) contains blocking artifacts. In contrast, there is no blocking artifacts and little harmonic distortion in Fig. 6(c).

VI. Conclusions

In this paper, a tiling method is proposed to construct aperiodic screens of any sizes with trained tile screens of fixed size. To support this tiling method, a DBS-based screen developing algorithm is proposed to produce tile screens that can be tiled seamlessly with the proposed tiling method. Simulation results show that the screening outputs of the screens constructed with the proposed tiling method bear blue noise characteristics, contain no blocking artifacts and carry little harmonic distortion. The performance of the proposed tiling scheme is better than regular tiling and random tiling schemes.

The tile screens are developed based on DBS in this work. Recent studies have found that multiscale error diffusion (MED) is able to produce halftones of any desired noise characteristics defined by various noise models flexibly [19-22]. It would be a positive direction to explore if tile screens developed with MED instead of DBS can provide a better screening performance.

Acknowledgment

The work described in this paper is substantially supported by a grant from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project No: PolyU5120/13E).

References

[1] D. L. Lau, and G. R. Arce, *Modern digital halftoning*. CRC Press, 2001

[2] R.Ulichney, "The void-and-cluster method for dither array generation," in *Proc. SPIE*, Vol.1913, pp.332-343.

[3] T.J. Alander, T. Mantere and T. Pyylampi, "Threshold matrix generation for digital halftoning by genetic algorithm optimization", *IEEE Trans. Commun*, 1998, Vol. 34, pp. 269-277.

[4] H. Ancin, A.K. Bhattacharjya and J. Shu, "Improving void-and-cluster for better halftone uniformity", in *Proc. of the SPIE*, Vol. 3300, pp. 321-329 (1998).

[5] H. Szu, Zhang, Y. Sun, and C. C. Li, (1993, October). "Neural network adaptive digital image screen halftoning (DISH) based on wavelet transform preprocessing," in *Proc. IEEE IJCNN'93*, Nagoya, Oct 1993, Vol. 2, pp. 1215-1218.

[6] M. Analoui, and J. Allebach, (1992, March). New results on reconstruction of continuous-tone from halftone. In *Proc. IEEE ICASSP'92*, Vol. 3, pp. 313-316, 1992

[7] J. Allebach and Q. Lin, "FM screen design using DBS algorithm," in *Proc. IEEE ICIP'96*, vol. 1, pp. 549-552, 1996

[8] T. Mitsa and K. J. Parker, "Digital halftoning technique using a blue-noise mask," *Journal of the Optical Society of America A*, 9(11), 1992, pp.1920-1929

[9] T. Mitsa and K. J. Parker, "Digital halftoning using a blue noise mask," in *Proc. IEEE ICASSP'91*, Vol.2, pp.2809–2812, 1991

[10] T. Mitsa and K. J. Parker, "Digital halftoning using a blue-noise mask," In *Proc. Electronic Imaging' 91*, San Jose, CA . International Society for Optics and Photonics, Jun, 1991, pp. 47-56

[11] M. Yao and K. J. Parker, "Modified approach to the construction of a blue noise mask," *Journal of Electronic Imaging*, 3(1), 1994, pp. 92-97.

[12] T. S. Rao, G. R. Arce and J. P. Allebach, "Analysis of ordered dither for arbitrary sampling lattices and screen periodicities," *IEEE Trans. ASSP*, 38(11), 1990, pp.1981-2000.

[13] T. S. Rao and G. R. Arce, "Halftone patterns for arbitrary screen periodicities," *JOSA A*, 5(9), 1988, pp. 1502-1511.

[14] F. S. Yang and A. Zakhor, (1993, November). "Block based halftoning at 45 degree screen angle," in *IEEE Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers,* 1993, pp. 940-945..

[15] V. Ostromoukhov and R.D. Hersch, "Halftoning by rotating non-Bayer dispersed dither arrays," in *Proc. SPIE*, Vol. 2411, Feb 1995, pp.180-197

[16] G. Y. Lin and J. P. Allebach, "Generating stochastic dispersed and periodic clustered textures using a composite hybrid screen," *IEEE Trans, Image Processing*, 15(12), 2006, pp. 3746-3758.

[17] C. Lee and J. P. Allebach, "The hybrid screen – improving the breed," *IEEE Trans. Image Processing*, Feb 2010, pp. 435-450

[18] D.Kacker and J.P. Allebach, "Aperiodic Micro Screen Design using DBS and Training," in *Proc. SPIE*, Vol. 3300, pp.386-397, 1998

[19] Y.H. Chan and S.M.Cheung, "Feature-preserving multiscale error diffusion for digital halftoning," *Journal of Electronic Imaging*, vol.13, no.3, pp.639-645, 2004

[20] Y. H. Fung and Y. H. Chan, " Tone-dependent noise model for high-quality halftones," *Journal of Electronic Imaging*, 22(2), 023004, 2013. doi:10.1117/1.JEI.22.2.023004

[21] Y.H. Fung and Y.H. Chan,"Optimizing the error diffusion filter for blue noise halftoning with multiscale error diffusion," *IEEE Trans. Image Process*., vol. 22, no.1, Jan 2013, pp.413-417

[22] Y.H. Fung and Y.H. Chan, "Green Noise Digital Halftoning with Multiscale Error Diffusion," *IEEE Trans. Image Process*., vol.19, no.7, Jul 2010, pp.1808-1823.