https://doi.org/10.2352/ISSN.2470-1173.2018.16.COLOR-427
© 2018, Society for Imaging Science and Technology

Color CLU-DBS Halftoning based on Neugebauer Primary Area Coverage: Improving the Breed*

***Weijuan Xi*[1], *Tal Frank*[2], *Yotam Ben-Shoshan*[2], *Robert Ulichney*[3], *Jan Allebach*[1]**
[1] ***Purdue University, West Lafayette, IN-47906, U.S.A., {xi9, allebach}@purdue.edu***
[2] ***HP Inc, Rehovot, ISRAEL***
[3] ***HP Labs, Stowe, MA***

Abstract

Though a color aperiodic, clustered-dot, halftoning (NPAC-MS-MP-CLU-DBS) algorithm can overcome the visible moire and rosette artifacts in conventional color halftoning methods, it still has some disadvantages, such as the color mismatch caused by the initial stage color management method, and texture artifacts caused by the concentric-ring cluster structure. In this paper, first, a new color gamut mapping method is used during the color management process, that is an image-dependent mapping method, which can make the most use of the printer color gamut, in order to reduce the color mismatch between the continuous-tone original and printed halftone images. Secondly, a new color, clustered-DBS halftoning algorithm with separated-cluster structure is developed. As a color halftoning method based on the clustered-DBS algorithm, not only it can overcome the visible moire and rosette artifacts, but also the separated-cluster structure is more stable, compared with the concentric-ring cluster structure. It can also reduce the texture artifacts significantly.

Introduction

Halftoning is the process of generating a pattern of pixels with a limited number of colors that creates the illusion of a continuous-tone image. However, color halftoning presents many problems that are unique to color, mainly due to the interactions between color planes. All color printers use a limited number of colorants, typically three or four colorants. The methods for digital color halftoning may be categorized into three groups according to the computational complexity required to render the continuous-tone image in halftone form, independent of the computation required to design the halftoning algorithm: screening, error diffusion, and iterative processes. Iterative techniques such as least squares [1] and direct binary search (DBS) [2][3] have also been applied to color halftoning. A major advantage of these approaches is that they can support a relatively complex HVS model.

For the conventional periodic clustered-dot halftoning method [8], which use three screens corresponding to cyan (C), magenta (M) and yellow (CMY) with different angles to halftone a color image, we always get visible moire and rosettes. Aperiodic clustered-dot halftoning methods have the advantage of resisting these artifacts: In 1990, Lau proposed the "Green-Noise Digital Halftoning" method [5][6]; In 2004, Damera Venkata proposed the "AM-FM Screen Design" method [7]; In 2013, Goyal proposed the CLU-DBS (clustered-dot Direct Binary Search) algorithm [10]; In 2017, Xi proposed the "Color Halftoning Based on Multi-Stage, Multi-Pass, Clustered-DBS" algorithm, which is the art-of-the-state. As much as we are aware, the only work that compares the quality of aperiodic clustered-dot halftone images generated by different methods is the 2010 Ph.D. dissertation of Gupta [8] showing that the CLU-DBS algorithm always generated the most smooth and homogeneous clustered-dot halftones.

In this paper, a new breed of color CLU-DBS halftoning algorithm based on Neugebauer Primaries Area Coverage (NPAC) [9] is introduced. Firstly, the background knowledge about MS-MP-CLU-DBS and NPAC-MS-MP-CLU-DBS (Color Halftoning Based on Multi-Stage, Multi-Pass, Clustered-DBS algorithm) will be introduced. After that, the new color management pipeline with an image-dependent color gamut mapping method based on NPAC is introduced briefly. Thirdly, the new color clustered-DBS algorithm is presented and also a comparison between the previous concentric-ring NPAC-CLU-DBS algorithm and the new separated color clustered-DBS algorithm is shown at the end of this paper.

Preliminaries

A. MS-MP-CLU-DBS Algorithm

MS-MP-CLU-DBS [10] is a variant of the monochrome DBS algorithm, which uses a dual-filter-based cost metric to generate clustered-dot textures. We use f and g to denote the grayscale and binary images, respectively, and let e represent the error image which stands for the difference between the halftone and gray scale images. So we have $e[\mathbf{m}] = g[\mathbf{m}] - f[\mathbf{m}]$, where $[\mathbf{m}] = [m,n]^T$ represents the discrete spatial coordinate. The cost metric is computed as

$$\theta = \theta_{homog} - \theta_{clust}, \tag{1}$$

where

$$\theta_{homog} = \sum_{\mathbf{m}} e[\mathbf{m}] c^{u}_{\tilde{p}\tilde{p}}[\mathbf{m}], \tag{2}$$

$$\theta_{clust} = 2\sum_{\mathbf{m}} e[\mathbf{m}] \Delta c_{\tilde{p}\tilde{e}_0}[\mathbf{m}], \tag{3}$$

$$c^{u}_{\tilde{p}\tilde{e}}[\mathbf{m}] = \sum_{n} e[\mathbf{n}] c^{u}_{\tilde{p}\tilde{p}}[\mathbf{m}-\mathbf{n}], \tag{4}$$

$$\Delta c_{\tilde{p}\tilde{e}_0}[\mathbf{m}] = \sum_{\mathbf{n}} e_0[\mathbf{n}] \Delta c_{\tilde{p}\tilde{p}}[\mathbf{m}-\mathbf{n}]. \tag{5}$$

Here $e_0[\mathbf{n}]$ is the initial halftone error at pixel $\mathbf{n}$, and $\Delta c_{\tilde{p}\tilde{e}_0}[\mathbf{m}]$ is initialized by $e_0[\mathbf{m}]$. In this cost metric, θ_{homog} encourages the

*Research supported by HP Inc., Palo Alto, CA.

formation of homogeneous texture and θ_{clust} encourages the formation of dot-clusters. The term θ_{homog} is computed by filtering the error image $e[\mathbf{m}]$ with the updated filter $c^{u}_{\tilde{p}\tilde{p}}[\mathbf{m}]$ (Eq. (4)), and then forming the inner product of this filter output with $e[\mathbf{m}]$ (Eq. (2)). The processing of the term θ_{clust} is similar (Eqs. (3) and (5)), except that the filter $\Delta c_{\tilde{p}\tilde{p}}[\mathbf{m}]$ is computed from the difference between the initialization and update filters, and the filtering is performed on the initial error $e_0[\mathbf{m}]$.

For the MS-MP-CLU-DBS algorithm, two kinds of refinement operations: Multi-Stage process and Multi-Pass process, are added, since experience shows that the structure of the initial halftone strongly affects the cluster distribution in the halftone image.

B. NPAC-MS-MP-CLU-DBS Algorithm

Different from the MS-MP-CLU-DBS algorithm, NPAC-MS-MP-CLU-DBS, as a color halftoning algorithm [11][12], can halftone color images and represent the halftone result with a selected set of Neugebauer Primaries (NPs). Here we consider only the case of the 8 NPs, corresponding to the three colorants Cyan (C), Magenta (M), and Yellow (Y). In order to extend the monochrome MS-MP-CLU-DBS algorithm to color, we process the NPs one-by-one in a default sequence order, which is shown in Fig. 1.

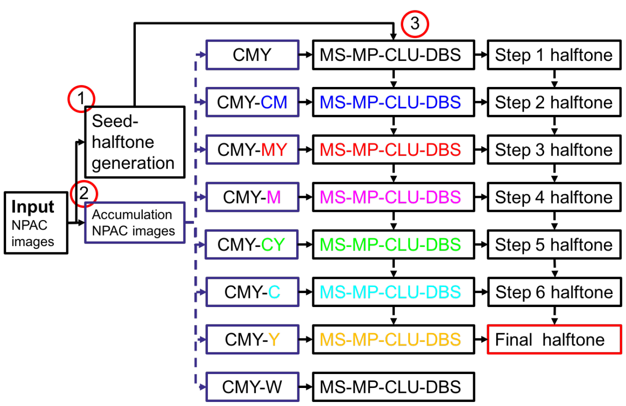


Figure 1. *Block diagram of NPAC-MS-MP-CLU-DBS.*

Similar to the MS-MP-CLU-DBS algorithm, NPAC-MS-MP-CLU-DBS still needs to generate the seed-halftone in the first place. Each dot in the seed-halftone should be the center of a dot-cluster for a sequence of NPs arranged in concentric rings. Secondly, we set the default NP order as: CMY, CM, MY, M, CY, C, Y, W, which follows the luminance values from the lowest to the highest. Then the 8 accumulation NPAC images are generated based on the percentage of each NP. Through these steps, the color halftoning problem can be transfered to 8 monochrome halftoning steps. For each pixel in the i-th accumulation image, the gray value is calculated as

$$g_{i_{[\mathbf{m}]}} = \sum_{j=1}^{i} p_{NP_{j_{[\mathbf{m}]}}}, \tag{6}$$

where $g_{i_{[\mathbf{m}]}}$ is the gray value of pixel in the i^{th} NP with spatial coordinate $[m,n]$, $p_{NP_{j_{[\mathbf{m}]}}}$ is the percentage of the j^{th} NP at spatial coordinate $[m,n]$. Figure 2 shows an example illustrating the process of generating a halftone image using the previously proposed concentric ring NPAC-MS-MP-CLU-DBS algorithm.

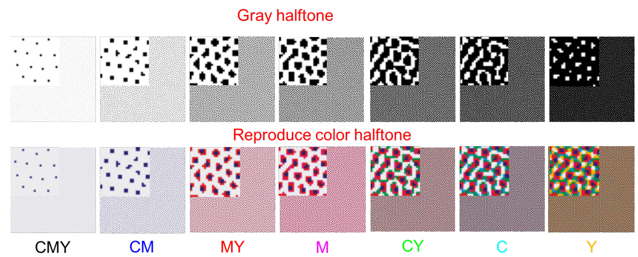


Figure 2. *Example: halftone result for the concentric-ring cluster NPAC-MS-MP-CLU-DBS algorithm*

Separated-Cluster Halftoning Algorithm

Although the NPAC-MS-MP-CLU-DBS can overcome the visible moire and rosette artifacts, the concentric-ring structure is still not stable enough and may increase the texture artifacts, as we can see in Fig. 2. For the new clustered-dot halftoning method proposed in this paper, we develop a separate-cluster structure for the color cluster halftoning, instead of the concentric-ring structure.

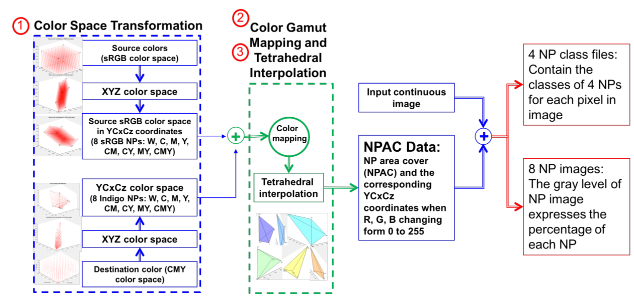


Figure 3. *Pipeline for color management.*

The new color aperiodic separated-clustered halftoning algorithm, which based on the clustered direct binary search (CLU-DBS), contains two main parts. The first part is the color management. Through color space transformation, color gamut mapping and tetrahedral interpolation, the percentage of each Neugebauer Primary (NP) can be calculated to get the NP images for every color continuous-tone image. The pipeline of the color management is shown in Fig. 3. The second part is the halftoning process based on the NPAC data that we can get from color management. The detailed information about these two parts is introduced in the next two subsections.

A. Color Management

Both the previous concentric-ring cluster NPAC-CLU-DBS algorithm and the new separated-cluster NPAC-CLU-DBS algorithm contain a color management process [13][14], which can calculate the NPAC data for each color. Compared with the color management in the concentric ring cluster algorithm, which is an image-independent color management method, the separated-cluster algorithm uses an image-dependent color management method, which can use the printer color gamut more efficiently and reduce the color mismatch. The new color gamut mapping method contains three steps: color space alignment, image-dependent compression, and image-independent compression. The block diagram of this process is shown in Fig. 4. Either color management scheme – that proposed previously for the concentric ring cluster algorithm and that proposed for the new separated cluster algorithm could be used with either halftoning algorithm. Presumably, the new color management scheme would yield improved image quality for the concentric ring cluster algorithm compared to that proposed previously.

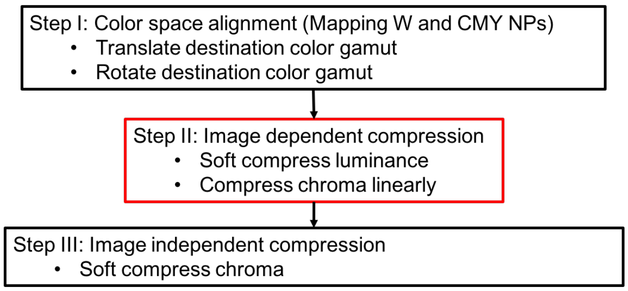


Figure 4. *Block diagram of color gamut mapping.*

Step I. Color Space Alignment

The source gamut is the sRGB color gamut; and the destination gamut is the Indigo printer color gamut, respectively. The prerequisite color mapping step is to transfer both source and destination color gamuts into the $YyCxCz$ color space[15]. The coordinates of the NPs in the source and destination color gamuts are shown in Table 1 and Table 2, respectively. As we can see, the destination white (W) and black (CMY) NPs are not perfectly neutral colors, while they are supposed to have the same coordinates with the source W and CMY NPs. Thus, the very first step of gamut mapping is to conduct color space alignment. Firstly, we translate the destination color gamut so that the destination CMY NP coordinate is $(0,0,0)$ in $YyCxCz$ color space, which is same as the source black (CMY) NP; Secondly, we rotate the destination color gamut about the CMY NP coordinate. After this step, the W and CMY NPs in both the source and destination color gamuts are collinear. The method for calculating the rotation matrix is illustrated as follows: We regard the W-CMY line in both the source and destination color gamuts as two 3D-vectors, so that we have

$$v_1 = YyCxCz_{W_{dest}} - YyCxCz_{CMY_{dest}}, \tag{7}$$

$$v_2 = YyCxCz_{W_{source}} - YyCxCz_{CMY_{source}}, \tag{8}$$

where $YyCxCz_{W_{dest}}$ and $YyCxCz_{CMY_{dest}}$ are the $YyCxCz$ coordinates of destination W and CMY NPs; and $YyCxCz_{W_{source}}$ and $YyCxCz_{CMY_{source}}$ are the $YyCxCz$ coordinates of the source W and CMY NPs. Then, the rotation vector and angle from v_1 to v_2 can be calculated using a cross product and a dot product

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \frac{v_1 \times v_2}{\|v_1 \times v_2\|}, \tag{9}$$

$$\theta = cos^{-1}\left(\frac{v_1 \cdot v_2}{\|v_1\| \cdot \|v_2\|}\right), \tag{10}$$

x is the rotation vector, and θ is the rotation angle. Based on these values, the rotation matrix R can be calculated using exponential map

$$R = e^{A\theta} = I + sin(\theta) \cdot A + (1 - cos(\theta)) \cdot A^2, \tag{11}$$

where I is the identity matrix and A is the skew-symmetric matrix corresponding to x, which can be expressed as

$$A = [x]_\times = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}. \tag{12}$$

After the translation and rotation operations, all the coordinates in destination color gamut are changed as follows

$$YyCxCz_{dest_{update}} = R \cdot (YyCxCz_{dest} - YyCxCz_{CMY_{dest}}). \tag{13}$$

The updated NP coordinates of the destination color gamut are shown in Table 3.

Step II. Image-Dependent Compression

The following is the basic idea of image-dependent compression: Instead of the sRGB color gamut, we regard the smallest cylinder which contains all the colors in the color image as our source color gamut. In order to implement the compression, we scale the luminance values of sRGB colors in the first place. Let Yy^M represents the max luminance value of all the luminance values in the source color image, and Y_W represent $Yy_{W_{dest_{update}}}$, which in short is the updated destination W. We don't compress the luminance of source image colors if $Yy^M \leq Y$. We should compress the lumiance of source image colors as follows, only if $Yy^M < Y$

$$Yy_0 = \begin{cases} (1-\lambda) \cdot \frac{Y_W}{Yy^M} Yy_i + \lambda Yy_i & if\ 0 \leq Yy_i < Y_W \\ (1-\lambda) \cdot \frac{Y_W}{Yy^M} Yy_i + \lambda Y_W & if\ Y_W \leq Yy_i \leq Yy^M \end{cases}, \tag{14}$$

where Yy_i represents the luminance value of a color in the source color image.

Next, we keep the hue of each color unchanged by dividing the whole color gamut into 360 hue sectors uniformly by different hue angles within in each hue sector, we scale the chroma linearly so that the chroma of all colors in the source color image within that hue sector are inside the smallest convex polygon in the $C_x - C_z$ plane that is formed by the chroma values of the 8 updated destination NPs. Figure 5 is an example to show the transformation of coordinates before and after the image-dependent compression. Each red star in Fig. 5 represents a color in the continuous-tone color image. The 8 updated destination NPs are also shown in the $Cx - Cz$ plane. Since, the chroma values of W and CMY NPs are 0 and the chromaticity of the CM NP is too small to compose a convex polygon, we can only focus on the other 5 NPs: C, M, Y, CY, and MY, and scale all the colors into the convex polygon composed by these 5 NPs (shown in Fig. 5) based on the formula

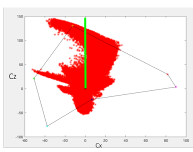
$$c_0 = \begin{cases} \frac{c_0^M}{c_i^M} c_i & if\ c_0^M < c_i^M \\ c_i & otherwise \end{cases}. \tag{15}$$

Table 1: $Y_yC_xC_z$ **coordinates of source NPs**

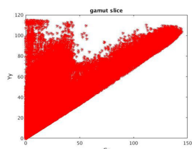
NPs	W	C	M	Y	CM	CY	MY	CMY
Y_y	116	90.190	32.842	108.968	7.032	83.158	25.811	0
C_x	0	-114.876	158.762	-43.886	43.886	-158.762	114.876	0
C_z	0	-41.124	-119.841	160.965	-160.965	119.841	41.124	0

Table 2: $Y_yC_xC_z$ **coordinates of destination NPs**

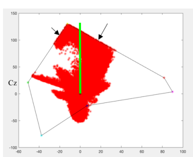
NPs	W	C	M	Y	CM	CY	MY	CMY
Y_y	98.480	26.524	20.353	84.922	2.851	19.784	19.458	2.176
C_x	0.0	-36.830	90.901	-12.296	7.809	-49.832	82.719	1.229
C_z	0	-77.928	3.294	130.052	-22.565	21.001	29.505	-0.183



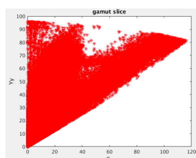
(a) top view of original image colors



(b) profile map of original image colors



(c) top view of step II mapping result



(d) profile map of step II mapping result

Figure 5. *Image dependent compression. Each red star represents a color in the color image:* ***(a)*** *is the top view of the original image colors;* ***(b)*** *is the profile map of the original image sliced at the green line;* ***(c)*** *is the top view of the image colors after the step II compression;* ***(d)*** *is the profile map of image colors after step II compression sliced at the green line.*

where, c_i and c_0 represent the chroma values of each color before and after the image-dependent compression, respectively; c_0^M represents the maximum chroma value of destination color gamut in i-th component (one of the 360 components); and c_i^M represents the maximum chroma value of the source colors in the i-th component.

Image-dependent compression can utilize the destination color gamut more effectively and reduce the color mismatch. It has a remarkable effect for the color image, that the colors only cover a small part of the sRGB source color gamut.

Step III. Image-Independent Compression

Since, we have already compressed the image colors into the smallest convex cylinder in the $Cx-Cz$ plane composed of the destination NPs, in this step we only need to soft compress the chroma for each color. This step looks like Step II but it is a totally different process since it is an image-independent compression. The compression scale only depends on the updated source color gamut obtained from Step II and the destination color gamut obtained from Step I, and not on the $YyCxCz$ coordinates or the color gamut of the image colors. Similar to what we did for Step II, we also need to divide the $Cx-Cz$ plane into 360 hue sectors, and compress the colors as follow

$$c_0 = \begin{cases} (1-\lambda)\frac{G_0^M}{G_i^M}c_i + \lambda c_i & if\ 0 \le c_i < G_0^M \\ (1-\lambda)\frac{G_0^M}{G_i^M}c_i + \lambda G_0^M & if\ G_0^M \le c_i \le G_i^M \end{cases} \tag{16}$$

where c_i and c_0 represent the color chroma values before and after the image-independent compression; G_0^M represents the maximum chroma value of updated destination color gamut in i-th hue sector; and G_i^M represents the maximum chroma value of updated source color gamut, which is formed by the destination NPs in i-th hue sector. The parameter $\lambda = \frac{1}{3}$ in our experiment. After this step, the $YCxCz$ coordinates of all the image colors are inside the updated destination color gamut, which means we can now use the 8 destination NPs to reproduce any colors in the color image.

Reproduction Image Comparison

In this section, we propose the method to do the image reproduction. The process is shown in Fig. 6. Based on the image reproduction method, we can generate the reproduction images for both color management methods – that proposed previously for the concentric ring cluster algorithm and that proposed for the new separated cluster algorithm in this paper. Figure 7 contains an illustration to compare the two color management methods. Figure 7(a) is a continuous-tone color image; Fig. 7(b) is the reproduction image based on the color management method proposed in this paper; and Fig. 7(c) is the reproduction image based on the color management method proposed previously for the concentric ring cluster algorithm. As one can see, the reproduction image in Fig. 7(b) has a better color match to the continuous-tone original image. As a result of that, the printed image quality can be improved significantly.

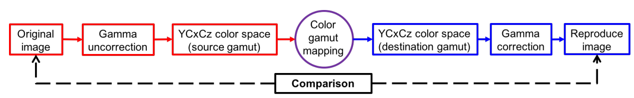


Figure 6. *Block diagram of generating reproduction image.*

Table 3: Updated $Y_yC_xC_z$ coordinates of destination NPs

NPs	CMY	CM	MY	M	CY	C	Y	W
Y_y	0	0.5483	16.2968	17.0369	18.2991	24.6837	83.1596	96.3120
C_x	0	6.5868	81.7039	89.8964	-50.8319	-37.7467	-12.4661	0
C_z	0	-22.3827	29.6565	3.4432	21.1498	-77.7910	130.0776	0

(a) original continuous image

(b) reproduction image based on the new color management method introduced in this paper

(c) reproduction image based on the color management method in [11]

Figure 7. Comparison between the color management methods in this paper and in paper [11].

Separated-Cluster NPAC-CLU-DBS Algorithm

Though the previous concentric-ring NPAC-MS-MP-CLU-DBS algorithm can overcome the visible moire and rosette artifacts effectively, its concentric-ring cluster structure may increase the texture artifacts and be less stable for printing. In this paper, we propose a separated-cluster NPAC-MS-MP-CLU-DBS algorithm, which has a separated-cluster texture, to overcome the disadvantage of the concentric-ring texture. The separated-cluster halftoning algorithm contains two main parts: 1). Generate and color the seed-halftone; 2). Generate the separated-cluster color halftone image based on the NPAC data.

Step I: Generate and Color the Seed-Halftone

Similar to the concentric-ring NPAC-MS-MP-CLU-DBS algorithm, we also need to generate a seed-halftone pattern at the first place. However, instead of generating the whole seed-halftone at the same time, we generate 4 groups of uniformly distributed seed-halftone dots. This idea comes from the tetrahedral interpolation step of color management. According to the tetrahedral interpolation, each sRGB color can be reproduced by at most 4 NPs, so that we only need to generate 4 groups of seed-halftone dots corresponding to 4 NPs instead of 8 NPs in all. In order to generate 4 groups uniformly interleaved halftone dots, swap-only standard DBS is used. The 8 NPs are labeled from 1 to 8, which is shown in Fig. 8; Let ρ represent the desired halftone frequency, $R(dpi)$ represent the printer resolution, then the seed-halftone absorptance of each group can be calculated as $\delta = (\rho/R)^2/4$.

NP order

i	1	2	3	4	5	6	7	8
NP	CMY	CM	MY	M	CY	C	Y	W

Figure 8. NP order.

Figure 9 shows an example result of the 4 groups of interleaved seed-halftone pattern. The detailed steps of using swap-only standard DBS algorithm to generate 4 groups of seed-halftone dots are discussed as follows. Firstly, we generate a uniform gray halftone pattern with gray level 4δ by the standard DBS algorithm. Then all the dots are equally divided into two groups; and the swap operation is processed between these two groups according to swap-only standard DBS until the algorithm converges to the smallest error between either group and the uniform pattern of gray level of 2δ to achieve two groups of interleaved seed-halftones. We repeat this process by dividing each group into two subgroups and similarly doing the swap-only step. After that we can get 4 groups of interleaved seed-halftone patterns, as shown in Fig. 9.

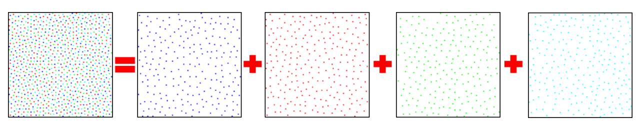


Figure 9. Seed halftone generation.

Then, we should color the dots in the seed-halftone patterns. Since each pixel can be reproduced by 4 NPs, we set the default color of a dot to be the i-th NP label of the current pixel, where i-th means the current dot belongs to the i^{th} seed-halftone group. As shown for example in Fig. 10, we assume one uniform pattern whose color can be reproduced by 25% CMY, 25% C, 25% M and 25% Y, so that for each pixel, the 4 labels corresponding to the colors are 1, 4, 6, and 7 according to Fig. 8. Then, for a dot in the first seed halftone group, it corresponds to the NP label 1 so that it will be colored by CMY; Similarly, for a dot in the second seed halftone group, it corresponds to the NP label 4 so that it will be colored by M, and so on.

Step II: Generate the Separated-Cluster Color Halftone Image

Based on the seed-halftone pattern, we perform the MP-CLU-DBS algorithm for each NP in sequence. Unlike the concentric-ring NPAC-MS-MP-CLU-DBS algorithm, which should generate the accumulation NPAC images, the

separated-cluster NPAC-MS-MP-CLU-DBS algorithm performs the halftoning process only according to each NP percentage. As shown in Fig. 11, in each stage and pass, we halftone each NPAC image by the CLU-DBS algorithm, then iterate to do the Multi-Stage and Multi-Pass processes. Specifically, the algorithm follows the criteria below:
1. We process each NP separately. We only consider the halftone dots corresponding to the current NP in each iteration and process the 8 NPs in sequence iteratively.
2. One pixel can only be colored by one NP, repeatedly depositing different NPs at the same pixel is not allowed.

Following this algorithm, we obtain the halftone results shown in Fig. 12. Compare with the previous concentric-ring NPAC-MS-MP-CLU-DBS algorithm and the PARAWACS-MS-MP-CLU-DBS algorithm [17], the halftone result of the separated-cluster halftoning algorithm has a better cluster texture, each NP in the halftone pattern has its own separated cluster and is not mixed with other NPs.

Conclusion

In this paper, we develop a new color management method that can reduce color mismatch effectively. In addition, the new separated-cluster NPAC-MS-MP-CLU-DBS algorithm can overcome the shortage of concentric-ring NPAC-MS-MP-CLU-DBS algorithm to yield a more stable halftone image. As a result of the new color management and separated-cluster halftoning algorithm, we can obtain higher image quality with inherently unstable marking processes, such as laser electrophotography.

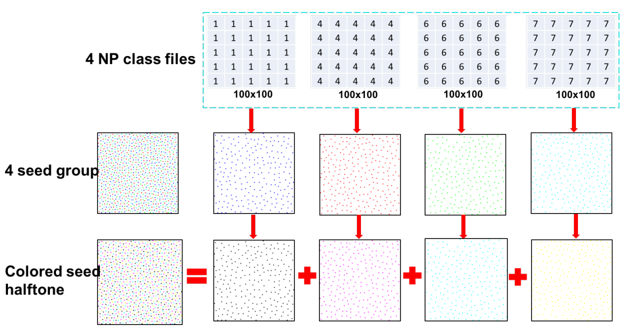


Figure 10. *Coloring the seed-halftone pattern according to the 4 NP files and NP order in Fig. 8. Two rows of halftone patterns are shown in this figure. The first row of patterns is randomly colored, which is only used to distinguish the dots in different groups. The second row is colored seed-halftone patterns according to the 4 NP files and NP order in Fig. 8.*

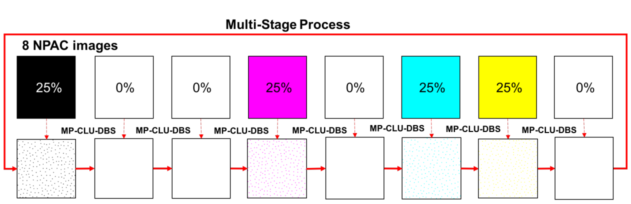


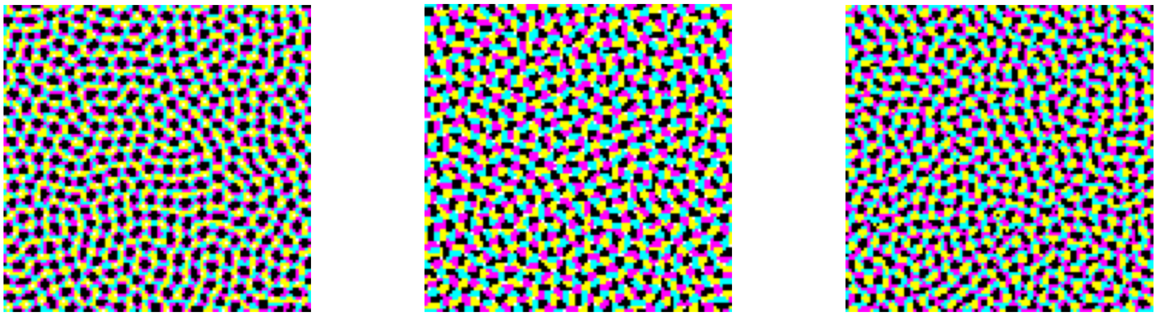
Figure 11. *Process to generate the separated-cluster halftone image.*

References

[1] T. Pappas, "Model-Based Halftoning of Color Images," IEEE Transactions on Image Processing, 6(7), pp.1014-1024, 1995.
[2] T. J. Flohr, B. W. Kolpatzik, R. Balasubramanian, D. A. Carrara, C. A. Bouman and J. P. Allebach, "Model Based Color Image Quantization," in Proc. SPIE, Human Vision, Visual Processing, and Digital Display IV, San Jose, CA, vol. 1913, pp. 270-281, Feb. 1993.
[3] A. U. Agar and J. P. Allebach, "Model-Based Color Halftoning Using Direct Binary Search," Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts V, Proc. SPIE 3963, pp. 521-535, 2000.
[4] F. Baqai and J. P. Allebach, "Computer-Aided Design of Clustered-Dot Color Screens Based on a Human Visual System Model," Estados Unidos: The Institute of Electrical and Electronics Engineers, Inc-IEEE, 2002.
[5] D. Lau, G. Arce and N. Gallagher, "Digital Halftoning by Means of Green-Noise Masks," Journal of the Optical Society of America A, vol. 16, no. 7, pp. 1575, 1999.
[6] B.S. Lee, "Characterization of Opponent Channels, Automated Detection of PQ Defects and Green-Noise Mask Design by Dual-Metric," PhD Thesis, 2006.
[7] N. Damera-Venkata and Q. Lin, "AM-FM Screen Design Using Donut Filters," in Proc. SPIE-IS&T. Color Imaging IX: Processing, Hardcopy, and Applications, vol. 5293, pp. 469480, Jan. 2004.
[8] M. Gupta, P. Goyal, C. Staelin, M. Fischer, O. Shacham and J. P. Allebach, "Improved Clustered-Dot Direct Binary Search Halftoning," IEEE Transactions on Image Processing, pp. 473-487, 2011
[9] J. Morovič, P. Morovič, and J. Arnabat, "HANS: Controlling Ink-Jet Print Attributes via Neugebauer Primary Area Coverages," IEEE Trans. Image Process. 21, 688696, 2012.
[10] P. Goyal, M. Gupta, C. Staelin, M. Fischer, O. Shacham, J. P. Allebach, "Clustered-Dot Halftoning with Direct Binary Search," IEEE Trans. on Image Process, Vol. 22, pp. 473487, February 2013.
[11] W. Xi, T. Frank, Y. Ben-Shoshan, R. Ulichney, J. P. Allebach, "Color Halftoning Based on Multi-Stage, Multi-Pass, Clustered-DBS," IS&T Electronic Imaging, Color Imaging XXII: Displaying, Processing, Hardcopy, and Applications, pp. 103-113, 2017.
[12] W. Jiang, W. Xi, U. Sarkar, R. Ulichney, J. P. Allebach, "Color Halftoning Based on Neugebauer Primary Area Coverage," IS&T Electronic Imaging, Color Imaging XXII: Displaying, Processing, Hardcopy, and Applications, pp. 91-102, 2017.
[13] R. S. Gentile, J. P. Allebach, E. Walowit, "A Comparison of Techniques for Color Gamut Mismatch Compensation," Proceedings of the SPIE, vol. 1077, pp. 34254, 1989.
[14] M. Wolski, J. P. Allebach, and C. A. Bouman, "Gamut Mapping: Squeezing the Most Out of Your Color System," IS&T/SID 2nd Color Imaging Conference, Scottsdale, pp. 89-92, 1994.
[15] B. W. Kolpatzik and C. A. Bouman, "Optimized Error Diffusion for Image Display," J. Electron. Imag., vol. 1, pp. 277292, July 1992.
[16] J.-H. Lee and J. P. Allebach, "Colorant Based Direct Binary Search Halftoning," Journal of Electronic Imaging, vol. 11, pp. 517527, Oct. 2002.
[17] P. Morovic, J. Morovic, J. Gondek, M. Gaubatz, R. Ulichney, "PARAWACS: Color Halftoning with a Single Selector Matrix," Final Program and Proceeding - IS and T/SID Color Imaging Conference, pp. 41-46, 2016.

Author Biography

Weijuan Xi received her BS in Electrical Engineering from North-



(a) concentric-ring structure (b) separate-cluster structure (c) PARAWACS

Figure 12. *Halftone Methods Comparison: We assume a uniform pattern can be reproduced by 25% CMY, 25% C, 25% M and 25% Y.* ***(a)*** *is the halftone result of the concentric-ring NPAC-MS-MP-CLU-DBS algorithm, which has a concentric-ring structure;* ***(b)*** *is the halftone result of the separated-cluster NPAC-MS-MP-CLU-DBS algorithm, which has a separated-cluster structure;* ***(c)*** *is the halftone result of the PARAWACS-MS-MP-CLU-DBS algorithm [17] which also has the concentric-ring structure.*

western Polytechnical University, China (2013). She is currently pursuing a Ph.D. degree, and working on image processing in the School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana, USA. Her work is focused on image quality, computer vision, and deep learning.