MODEL-BASED ANALYSIS AND DESIGN OF COLOR SCREEN SETS FOR CLUSTERED-DOT PERIODIC HALFTONING AND DESIGN OF MONOCHROME SCREENS BASED ON DIRECT BINARY SEARCH FOR APERIODIC DISPERSED-DOT HALFTONING

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Altyngul Jumabayeva

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Prof. Jan P. Allebach, Chair

School of Electrical and Computer Engineering

Prof. George T.C. Chiu

School of Mechanical Engineering

Prof. Edward J. Delp

School of Electrical and Computer Engineering

Dr. Robert Ulichney

HP Labs Hewlett-Packard, Inc.

Approved by:

Prof. Dimitrios Peroulis

Head of the School Graduate Program

This document is dedicated to my parents, Abay and Gulzhan, my brother Olzhas, my husband Daulet, and my son Zhan.

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor, committee members, and my research sponsor, the Hewlett-Packard Indigo Division, who made this work possible. First and foremost, I would like to thank my advisor Prof. Jan P. Allebach for providing me the best opportunity I could wish for, which is working with Professor Allebach and the Hewlett-Packard Indigo Halftoning Team. Thank you very much for the invaluable mentoring at each step of my graduate studies, for your patience and for your greatest support, I'll always treasure the memory of studying and doing research under your supervision. I would also like to give sincere thanks to all other committee members Prof. George T.C. Chiu, Prof. Edward J. Delp, and Dr. Robert Ulichney. They were always approachable and ready to make all the arrangements required for the smooth progress of my studies. My special thanks to Dr. Robert Ulichney, Tal Frank and Yotam Ben-Shoshan for always providing great advice and support with my research. I would like to thank my Mom, my Dad, my brother, my husband, and my friends for constant support during my stay at Purdue. Lastly, I would like to thank Purdue University for providing great research opportunities and high quality education for all students around the world.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

ABBREVIATIONS

CPHC	continuous parameter halftone cell
DPHC	discrete parameter halftone cell
HVS	human visual system
BSB	basic screen block
DBS	direct binary search
CCDS	color-content-dependent screening

ABSTRACT

Jumabayeva, Altyngul PhD, Purdue University, August 2019. Model-based Analysis and Design of Color Screen Sets for Clustered-Dot Periodic Halftoning and Design of Monochrome Screens Based on Direct Binary Search for Aperiodic Dispersed-Dot Halftoning. Major Professor: Jan P. Allebach.

Periodic clustered-dot halftones are widely used in electrophotographic printers due to the relatively poor print stability of this class of printers. It is important to analyze the nature and the causes of perceived fluctuation in order to understand the factors that prevent the high-end digital presses from achieving the same print quality as the commercial offset presses. In order to better approximate the screen sets used for the commercial offset presses, irregular screen sets can be considered. We start by developing a set of candidate screen tile-vectors that best fit the specified screen frequency, screen angle, and printer resolution. We then perform Fourier-based analysis of regular and irregular periodic, clustered-dot halftone textures in order to understand how perceived fluctuation relates to the halftoning technology. After exploring the search for the best single separation geometry, we consider the superposition of multiple periodic clustered-dot halftones, and propose to apply HVS-based model, which assists us in finding the best color assignments to the superimposed halftones. It turned out that the choice of the best color assignments depends on different combinations of colorant absorptance values, hence we propose to apply different color assignments within the image depending on the local color and content of the image. Next, we propose a step-by-step screen design for standard and high resolution periodic irregular clustered-dot halftones. Finally, we presented monochrome DBS-based screen design with overlapping clusters of 2×2 or 3×3 pixels, which can also be used in electrophotographic printers.

1. INTRODUCTION

1.1 Digital halftoning and areas of application

Digital Halftoning is the process of rendering a continuous tone image with a limited number of tone levels [1]. The goal of digital halftoning is to generate an image with correct tone and detail of an original image without introducing any visible artifacts. There are many areas where halftoning is used. For example, in certain applications such as image enhancement, image scanning or image compression, a technique called inverse halftoning is used. Whereas halftoning is a technique for converting grayscale images to halftone images, inverse halftoning is a technique used to reconstruct grayscale images from halftone images. There are various inverse halftoning methods. For example, in [2], Multilayer Perceptron neural network (MLP) trained by a Backpropagation (BP) was proposed to be used in order to achieve high quality reconstructed grayscale images. In [3], the least squares filtering and the edge detection classification were suggested to be used. The proposed method was said to enhance the quality of output grayscale images for inverse halftoning. Kite et al. present a fast inverse halftoning algorithm and a new multiscale gradient estimator for error-diffused halftones [4]. They also compare the implementation cost, peak signal to noise ratio, and visual quality with other inverse halftoning algorithms. There's also been a considerable amount of research on color inverse halftoning [2], [5], [6].

Another important area of halftoning is 3D halftoning [7]. In [7], a volumetric slicing method with halftoning algorithm that simultaneously converts the color of 3D models into printable layers was proposed. Their work primarily focused on color reproduction for light-curable 3D printing technology.

Due to the fact that halftoning is employed in image printing, it also plays an important role in digital watermarking and data hiding [8–12]. For example, it can be used in order to protect ownership rights or in order to prevent the illegal distribution of a printed document without permission of its owner. In [12], Guo et al. propose the Look-Up-Table (LUT) strategy to efficiently embed multiple watermarks into a set of multi-scale watermarks, which are then embedded into a host halftone image. Their method significantly improves the embedding capacity and reliability [12].

In this work, we investigate the use of halftoning in printing systems. Halftoning plays an integral role in almost all printing systems. The choice of halftone pattern type and complexity level of the algorithm is dependent on the characteristics of the marking technology, the application space and price-point of the system, and the customer expectations. The two most widely used digital marking technologies are electrophotography and inkjet. Both these technologies are deployed in very low-cost printers intended for home and small-office use, as well as much more expensive commercial printing systems.

According to Dun&Bradstreet First Research, the global commercial printing industry generated about $785 billion (USD) in annual revenue in 2017 [13]. And according to Printing Impressions, another research organization, 63.1% of the capital investments made by commercial printers during the past 3 years were for digital presses, as the migration from traditional analog offset lithographic presses to digital presses continues [14]. The work presented in this dissertation is particularly relevant to high-end digital presses that are widely used in the commercial printing industry.

Offset, lithographic printing systems use a fixed plate to image each color separation for each impression onto the media, typically paper. Although contemporary systems use digital marking technologies to create the plate, once the plate has been created, it can only be used to print one content per impression. Thus, relatively long print runs are needed for offset, lithographic printing to be economically viable. In contrast, digital marking technologies offer much more flexibility in terms of the characteristics of the print job. Every succeeding page can be different, and very short print runs are possible. The challenge has been for high-end digital presses to achieve the level of image quality that customers are used to seeing with prints made using offset, lithographic presses. Achieving that goal with digital presses based on electrophotographic technology is the primary motivation for our work.

Previously, the plates used with offset, lithographic systems were generated using analog imaging systems with very high resolution. The halftone patterns that have traditionally been used are clustered-dot, periodic patterns. Specifically, a single screen that generates square halftone cells is rotated to a different angle for each colorant [15]. The angles that are traditionally used are 15°, 45°, 75°, and 0°, for cyan (C), magenta (M) , black (K), and yellow (Y), respectively. The goal is to achieve maximal angular separation between C, M, and K. The Y screen is not rotated, because the halftone pattern in the yellow separation is much less visible. More recently, digital plate-writers have replaced these analog imaging systems. However, these plate-writers still have very high resolution, typically in the range of 4000 dots/inch. Thus, the halftone patterns that are printed can effectively be the same as those used with earlier analog plate writing systems.

In contrast, digital, electrophotographic presses are disadvantaged in two ways. First, they have much more limited resolution, typically less than 1000 dots/inch. Second, the electrophotographic marking process is not as stable as is the offset, lithographic marking process. What this means is that there is more variation from dot-cluster to dot-cluster, there is more noise due to scattered colorant, and small dot-clusters may not consistently develop. The overall effect is to create the appearance of graininess in the printed image, which customers find to be objectionable. In addition, large area variation in the period of the halftone patterns, due to mechanical drift in registration or local deformation of the media itself, can cause spatial variation in the interference between the superimposed halftone patterns used to print the different color separations. This phenomenon can result in very objectionable, low-frequency, moire patterns.

1.2 Digital halftoning algorithms and halftone textures

Viewed according to computational complexity, digital halftoning algorithms can be broadly categorized into three groups: 1) point processes - ordered dithering or screening [16]; 2) neighborhood processes - error diffusion [17–22] and dot diffusion [23–25]; 3) iterative processes: Direct Binary Search (DBS) [26, 27]. Among the three groups, the best image quality is considered to be provided by DBS. However, the processing time of DBS may become an issue. Due to the fact that DBS seeks to minimize the total squared perceived error, while algorithms in point and neighborhood processes are mostly heuristic, DBS can generate halftone images of the best quality. The search of an optimal solution for DBS involves two operations: toggle and swap. Among recent works, Liao in [28] proposes a new DBS algorithm called "monotonically non-decreasing swap" (MNDS) DBS which considers toggle and swap separately and the swap operations are considered in the order from the edge to the center of the filter. The second best quality is considered to be provided by error diffusion, especially the tone dependent error diffusion described in [18]. The image quality resulted by the ordered dithering is inferior to the above two techniques, however, it can provide the highest processing speed.

Halftone textures can be classified along two dimensions: periodic or aperiodic, and clustered-dot or dispersed-dot. All four combinations are used in practice. In dispersed-dot halftoning, different gray levels are achieved by changing the relative density of printed dots, whereas in clustered-dot halftoning, different gray levels are achieved by changing the size of printed dots on the page. Due to the fact that the two approaches are analogous to frequency modulation (FM) and amplitude modulation (AM) used in communication systems, sometimes they are referred to as FM and AM halftoning. There has been a lot of research in AM-FM image representation. Havlicek et al. propose to compute the AM-FM image representation for multicomponent, nonstationary images using a statistical component model [29]. Pattichis et al. suggest using an AM-FM representation for fingerprint classification [30]. Acton et al. provide an automated method to repair broken, occluded oriented image textures based on partial differential equations and AM-FM image modeling [31].

FM and AM halftoning can be used in the screening process. Usually, dispersed-dot (FM) screens are more robust to moire artifacts, but appear noisier due to the stochastic nature of the halftone textures. For example, aperiodic, dispersed-dot screens produce what is commonly called blue-noise textures. The concept of blue noise halftoning was first introduced in [32] by Ulichney. The spectra of blue-noise halftones are composed entirely of high-frequency components that are least visible to human viewer [32], [33]. He in [34] presented hierarchical colorant-based direct binary search (HCB DBS) halftoning, which is considered an image quality benchmark for dispersed-dot halftoning algorithms. Similarly, he proposed hierarchical error diffusion (HED) in [21]. Ishizaka in [35] proposes the new spatial measure for dispersed-dot halftoning assuring good point distribution in any density. The work presented in [36] describes a methodology for design of color, aperiodic, dispersed-dot halftoning algorithms. Aperiodic, dispersed-dot halftoning (FM) algorithms are most appropriate for digital printing systems based on inkjet technology. However, they are also used with commercial, offset analog presses, and are of growing interest for digital presses. The methodology described in [36] is based on the concept of Neugebauer Primary Area Coverage (NPAC), and the use of a selection matrix to decide which Neugebauer primary will be printed at each printer-addressable pixel [36–38].

Dispersed-dot screens are primarily used in marking devices which can easily render an isolated dot, such as inkjet printers, whereas clustered-dot screens are used in electrophotographic devices which cannot easily produce an isolated dot. For example, we can use aperiodic, clustered-dot screens with electrophotographic printers. The textures formed by aperiodic clustered-dot screens are usually referred to as green-noise textures [39], [33]. The concept of green noise was first introduced by Lau et al. [39]. They also proposed an algorithm to design screens that generate green-noise textures [40]. Multiscale error diffusion algorithm to produce halftones of desirable green noise characteristics was proposed in [19]. Their algorithm allows one to modify the desirable cluster size freely and provides a linear relationship between the cluster size and the input gray level. Their approach succeeds in eliminating pattern and directional artifacts, and preserving original image details. Similarly, they also optimized the error diffusion filter for blue noise halftoning in [20].

If we were to compare periodic and aperiodic clustered-dot screens, we'd find out a greater level of noisiness in aperiodic clustered-dot screens, especially when they are used in color halftoning. For electrophotographic printers, periodic, clustered-dot screening is preferred to other halftoning approaches due to its stable dot formation, and robustness to printer artifacts, such as dot gain and banding. There is, however, another problem when clustered-dot screens are used to halftone color images. The problem here is that the superposition of the individual color plane separations can result in moire patterns that are extremely objectionable from the viewer's perspective. There are different approaches for solving this problem. For example, Babaei and Hersch propose a novel use of lines to generate color halftone patterns [41], [42]. Since the patterns consist of a thin line for each colorant, they fall in the domain of clustered-dot, periodic patterns that are well-suited to digital presses based on the electrophotographic process. The line patterns are designed in such a way as to explicitly control the interaction of the individual halftone patterns for each colorant, and thus explicitly control the moire that could result from the superposition of these patterns. This paper shares many common elements with the work presented by Jiang et al. in [43] in that it generates color halftone patterns with a single angle and single frequency that incorporate explicit control of colorant overlap. In addition, the HP Indigo Division has recently announced a similar solution that is called "the silky screen" in the trade literature [44], but which has not been described in the scholarly literature.

Chen et al. discuss another approach to the control of moire in the printing of digital color halftone images that is based on lattice theory [45, 46]. This work only considers the choice of screen geometry. It does not address the arrangement of the elements in the threshold matrix that control the shape of the halftone dots at each colorant level. Thus, it does not provide a complete end-to-end procedure for design of threshold matrices for generating digital, color, periodic, clustered-dot halftone patterns, Halftoning solutions based on this screen design approach are currently offered by the HP Indigo Division for their products. They are sold under the trademark “rainbow screen”. The elements of the screen design process that are not described in the paper above have not been published in the open literature.

1.3 Overview of my work

It is important to analyze the nature and the causes of perceived fluctuation in order to understand the factors that prevent the high-end digital presses from achieving the same print quality as the commercial offset presses. Currently, offset plates are generated using very high-resolution plate-writers that can generate faithful renderings of circular halftone dots on any chosen lattice of dot centers. In contrast, digital presses have much lower resolution, and therefore are limited in their capabilities to approximate the screen frequencies and angles that are conventionally used with offset presses. The combination of screen frequency and angle is conveniently represented by the 2×2 *periodicity matrix* $\mathbf{N} = \begin{bmatrix} \mathbf{v_1}^T & \mathbf{v_2}^T \end{bmatrix}$, where $\mathbf{v_1} = \begin{bmatrix} v_{11} & v_{12} \end{bmatrix}$ and $\mathbf{v_2} = \begin{bmatrix} v_{21} & v_{22} \end{bmatrix}$, $v_{11}, v_{12}, v_{21}, v_{22}$ are rational numbers [47], [48].

In Chapter 2, we introduce the algorithm for determining candidate screen tile vectors in order to better approximate the desired screen angle and screen frequency. Candidate screen tile vectors are obtained by admitting non-integer-valued elements in the periodicity matrix. This will result in an *irregular* halftone pattern, in which the shape of the dot clusters varies from dot-to-dot. In contrast, a *regular* halftone pattern results when the periodicity matrix has only integer-valued elements [49]. In this case, each dot-cluster is identical; but the set of achievable combinations of screen frequency and angle is limited. The disadvantage of irregular screens is that the variation in dot shape leads to an increase in perceived fluctuation or noisiness in the printed halftone pattern. To our knowledge, there is relatively little published work on design of irregular halftone screens.

Design of color screen sets for digital presses is a challenging task. First, it is necessary to choose a set of periodicity matrices, one for each colorant, typically cyan, magenta, yellow, and black, that will yield good visual quality when printed individually and also in combination. This requires consideration of moire effects that arise when the halftoned separations are superimposed [45, 46, 50–53] as well as the effect of misregistration between the separations that may occur during printing [48, 50, 54–56]. Once the set of periodicity matrices has been chosen, then it is necessary to develop the sequences according to which dots are added to the dot clusters in the highlight region, and holes are added to the hole clusters in the shadow region [48]. This must be done separately for each color plane or separation. Finally, images are printed using the trial halftone screens; and these are evaluated for the presence of artifacts. This last step is essential, since existing printer models cannot account for all aspects of the printing process that may affect print quality.

We start by developing a set of candidate screen tile-vectors that best fit the specified screen frequency, screen angle, and printer resolution. We then perform Fourier-based analysis of regular and irregular periodic, clustered-dot halftone textures in order to understand how perceived fluctuation relates to the halftoning technology. The main advantage of the proposed model lies in predicting the perceived fluctuation metrics based solely on the periodicity matrix.

In Chapter 3, after exploring the search for the best single separation geometry, we consider the superposition of two periodic clustered-dot halftones, and investigate how to make the best color assignments to the two regular or irregular clustered-dot halftones in order to minimize the perceived error. As we explore the superposition of two irregular color halftones, we need not only take into account the effects of superposition but also the effects of digitization. In order to eliminate dot-on-dot printing in the superposition image, the screen of each colorant is rotated to a different angle. However, this causes other problems such as moire and rosette artifacts, which are the result of the interaction between two lattices. The moire phenomenon refers to a low-frequency structure, which has a very visible pattern and can be observed at the superposition of two halftones, and not in the individual halftones separately [48,51,52]. In contrast, a rosette pattern has a circular or polygonal pattern, which is formed as a result of rotating halftone screens [48,51,52]. The superposition of two color halftones is a very important question to study because the artifacts such as moire, rosette and misregistration can lead to color shifts. Baqai and Allebach presented a systematic method for designing moire- and rosette- free clustered-dot color screens for discrete raster color systems [48]. Whereas Ufuk and Allebach developed a model-based color halftoning method which minimizes the perceived error between the continuous tone original color image and the color halftone image [57]. In our research, we develop a model that agreed with a human observer and allowed fast implementation. The perceived error helps us identify the configuration of colors and screens that will improve the appearance of the superposition image. We also investigate the effects of misregistration on image fluctuation and colorshift.

After applying our HVS-based model to the superposition of three and four clustered-dot color halftones, we came to the conclusion that this color assignment plays a significant role in producing the high quality images. Moreover, the choice of the best color assignments depends on different combinations of colorant absorptance values. Hence, in Chapter 4, we propose to apply different color assignments within the image depending on the local color and content of the image. If the image content locally has a high variance of color and texture, the artifacts due to halftoning will not be as visible as the artifacts in smooth areas of the image. Therefore, our algorithm involves detecting smooth areas of the image by segmenting the image based on the color of the content, and applying the best color assignments in those areas. This approach will primarily improve the quality of rendering the large smooth areas, such as memory colors or flesh tones. The concept of performing image segmentation based on the content and using different halftoning techniques in different regions of the image was investigated before. Park et al developed a method in which they divided a document into smooth and detail objects, and used low-frequency, periodic clustered-dot halftoning in smooth areas to promote stable development, and high-frequency, periodic clustered-dot halftoning in detail areas to provide better rendering of the detail in the image [58]. Ostromoukhov and Nehab checked the local gradient at each pixel, and chose a basic dither matrix based on the magnitude of the gradient [59]. Huang and Bhattacharjya described a process for switching between a periodic, clustered-dot screen in smooth areas, and error diffusion with a screen in detail areas [60]. The screen used for both detail and smooth regions is the same. References [59] and [60] address only halftoning of monochrome images. Reference [58] does consider color. But it is targeted to home/office laser electrophotographic printers, not high-end digital presses. The main novelty of our approach, which is targeted to printing with high-end digital presses, is the application of different color assignments within the image depending on the local color and content of the image, without changing the overall set of screens that are used.

The primary focus of our research is periodic irregular clustered-dot halftones, which provide a larger space of available screen geometries in comparison to the regular clustered-dot halftones, and hence may result in better screen sets with respect to smoothness and robustness. Therefore, we needed a screen design approach that would work for both regular and irregular clustered-dot halftones. In Chapter 5, we started by designing base levels based on the work presented by Baqai and Allebach in [48], and we extended their approach to irregular halftones. Then in order to remove contouring and maze-like artifacts, we followed the work presented by Lee and Allebach in [47], and we extended the hybrid screen design to irregular halftones for standard and high resolution printing. High resolution printing involves considering a cluster of 2×2 pixels at a center of a microcell instead of a single pixel.

In Chapter 6, we present monochrome DBS-based screen design with overlapping clusters of 2×2 or 3×3 pixels, which can also be used in electrophotographic printers. We started by reviewing monochrome DBS-based screen design which produces dispersed-dot textures and is usually used for inkjet printers, which have no trouble in rendering isolated dots. Due to the fact that with electrophotographic printers, isolated dots are not stable, we proposed to use a cluster of 2×2 pixels as a single unit instead of a single pixel. We then introduced overlapping clusters by swapping single dot pixels with single hole pixels to create bigger dot and hole clusters, which resulted in the monochrome DBS-based screen design with overlapping clusters of 2×2 pixels. Lastly, we extended our screen design to the use of clusters of 3×3 pixels instead of 2×2 pixels.

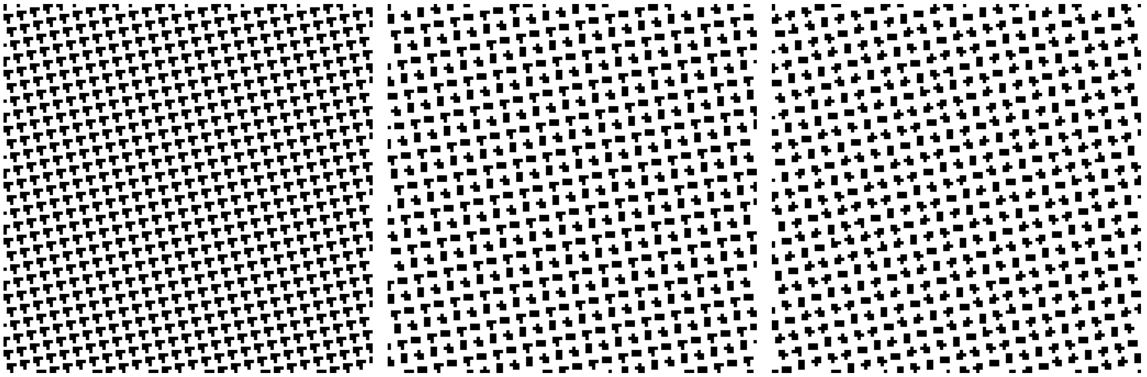
Finally, in Chapter 7, I present my major contributions, list my publications and propose what can be done in the future.

2. SINGLE SEPARATION ANALYSIS FOR CLUSTERED DOT HALFTONES

2.1 Introduction

High-end digital printing that is intended to compete with traditional offset lithographic printing is one of the most exciting new growth areas in printing. Customers in this market segment base their judgment of print quality, and thus the acceptability of digital print output as a substitute for offset prints, on the quality to which they are accustomed with offset. Currently, offset plates are generated using very high-resolution plate-writers that can generate faithful renderings of circular halftone dots on any chosen lattice of dot centers. In contrast, digital presses have much lower resolution, and therefore are limited in their capabilities to approximate the screen frequencies and angles that are conventionally used with offset presses. The combination of screen frequency and angle is conveniently represented by the 2×2 *periodicity matrix* $\mathbf{N} = \begin{bmatrix} \mathbf{v_1}^T & \mathbf{v_2}^T \end{bmatrix}$, where $\mathbf{v_1} = \begin{bmatrix} v_{11} & v_{12} \end{bmatrix}$ and $\mathbf{v_2} = \begin{bmatrix} v_{21} & v_{22} \end{bmatrix}$, $v_{11}, v_{12}, v_{21}, v_{22}$ are rational numbers [47], [48].

In this chapter, we first, introduce the algorithm for determining candidate screen tile vectors in order to better approximate the desired screen angle and screen frequency. Candidate screen tile vectors are obtained by admitting non-integer-valued elements in the periodicity matrix. This will result in an *irregular* halftone pattern, in which the shape of the dot clusters varies from dot-to-dot. In contrast, a *regular* halftone pattern results when the periodicity matrix has only integer-valued elements [49]. In this case, each dot-cluster is identical; but the set of achievable combinations of screen frequency and angle is limited. The disadvantage of irregular screens is that the variation in dot shape leads to an increase in perceived fluctuation or noisiness in the printed halftone pattern. Fig. 2.1 illustrates the relative quality of regular and irregular halftone textures. To our knowledge, there is relatively little published work on design of irregular halftone screens.



(a) (b) (c)

$$\mathbf{N}_1 = \begin{bmatrix} 4 & -1 \\ 1 & 4 \end{bmatrix} \qquad \mathbf{N}_2 = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix} \qquad \mathbf{N}_3 = \begin{bmatrix} \frac{13}{3} & \frac{-7}{6} \\ \frac{7}{6} & \frac{13}{3} \end{bmatrix}$$

Fig. 2.1.: The three halftones (a), (b), and (c) correspond to the same target tile vector with a screen frequency of 180 lpi, screen angle of 15°, and printer resolution of 812.8 dpi. The halftone in (a) is a regular halftone, whereas the halftones in (b) and (c) are irregular halftones. Note that with the use of irregular halftones ((b) and (c)), the shapes of dot clusters are not identical, which may lead to an increase in perceived fluctuation in the printed halftone pattern.

Design of color screen sets for digital presses is a challenging task. First, it is necessary to choose a set of periodicity matrices, one for each colorant, typically cyan, magenta, yellow, and black, that will yield good visual quality when printed individually and also in combination. This requires consideration of moire effects that arise when the halftoned separations are superimposed [50], [51], [52], [45], [46], [53] as well as the effect of misregistration between the separations that may occur during printing [48], [50], [54], [55], [56]. Once the set of periodicity matrices has been chosen, then it is necessary to develop the sequences according to which dots are added to the dot clusters in the highlight region, and holes are added to the hole clusters in the shadow region [48]. This must be done separately for each color plane or separation. Finally, images are printed using the trial halftone screens; and these are evaluated for the presence of artifacts. This last step is essential, since existing printer models cannot account for all aspects of the printing process that may affect print quality.

It is in the context of this larger problem that we have conducted the work reported in this chapter. Specifically, our goal is to develop a tool that can be used to eliminate periodicity matrices that will result in poor quality halftones, based only on an analysis of the periodicity matrix itself, without considering the specific threshold matrix that will control the growth of dot clusters in highlights and hole clusters in shadows, as well as the midtone halftone textures. Consequently, we consider only a single separation; and we assume an ideal circular dot shape for a single absorptance level = 0.25, where 0 corresponds to no colorant (white) and 1 corresponds to maximum colorant (black). In [67], we described the Fourier analysis that underlies this work. In this chapter, we develop a rigorous framework for determining the minimum set of frequency components that must be included in the evaluation of halftone fluctuation. We also propose a set of four ratios to quantify this fluctuation. These ratios are fully based on an analytic expression for the Continuous-Space Fourier Transform (CSFT) of the halftone pattern.

2.2 Preliminaries

In this chapter, we use boldface to indicate vector or matrix quantities. We use lower case to indicate vector quantities, and upper case to indicate matrix quantities. We use $(\mathbf{x}) = (x, y)^T$ to represent the continuous coordinates in the units of inches. We use $(\mathbf{u}) = (u, v)^T$ to represent frequency coordinates, where u and v are the horizontal and vertical spatial frequency variables in the units of cycles per inch.

2.3 Development of a Set of Candidate Screen Tile Vectors

2.3.1 Algorithm for obtaining candidate screen tile vectors

The design of any periodic screen starts by introducing the periodicity matrix $\mathbf{N} = \begin{bmatrix}\mathbf{v_1}^T & \mathbf{v_2}^T\end{bmatrix}$, where $\mathbf{v_1} = \begin{bmatrix}v_{11} & v_{12}\end{bmatrix}$ and $\mathbf{v_2} = \begin{bmatrix}v_{21} & v_{22}\end{bmatrix}$, $v_{11}, v_{12}, v_{21}, v_{22}$ are rational numbers [47]. The whole spatial domain can then be tiled according to $k_1\mathbf{v_1} + k_2\mathbf{v_2}$, $(k_1, k_2) \in \mathbb{Z}^2$. In our work we only consider the orthogonal and square geometries, hence $\mathbf{v_1} \cdot \mathbf{v_2} = 0$, and $\|\mathbf{v_1}\| = \|\mathbf{v_2}\|$, where $\|\mathbf{v_1}\|$ is the microcell size. For example, if $\mathbf{N} = \begin{bmatrix}4 & -1\\ 1 & 4\end{bmatrix}$, then the geometry is square and orthogonal with the microcell size of $\|\mathbf{v_1}\| = \|\mathbf{v_2}\| = \sqrt{5}$ [47], [45]. The two tile vectors generate a parallelogram, which is also known as the *continuous-parameter halftone cell* (CPHC) [45]. The CPHC for the geometry with periodicity matrix $\mathbf{N} = \begin{bmatrix}4 & -1\\ 1 & 4\end{bmatrix}$ is illustrated in Fig. 2.2.

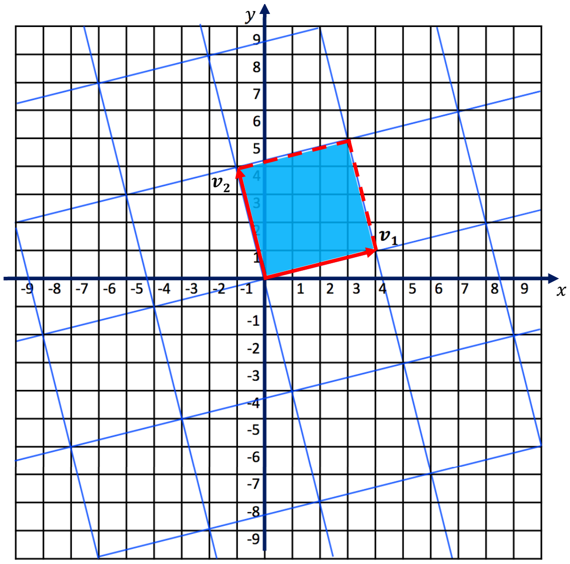


Fig. 2.2.: Example of the continuous parameter halftone cell (CPHC) for the geometry with periodicity matrix $\mathbf{N} = \begin{bmatrix}4 & -1\\ 1 & 4\end{bmatrix}$.

Due to the assumption that the screens are square and orthogonal, the tile vectors $\mathbf{v_1}$ and $\mathbf{v_2}$ take the following form:

$$\begin{aligned} \mathbf{v_1} &= \begin{bmatrix} v_{11} & v_{12} \end{bmatrix}, \\ \mathbf{v_2} &= \begin{bmatrix} -v_{12} & v_{11} \end{bmatrix}. \end{aligned} \tag{2.1}$$

Hence, we have the periodicity matrix $\mathbf{N} = \begin{bmatrix} v_{11} & -v_{12} \\ v_{12} & v_{11} \end{bmatrix}$, where v_{11} and v_{12} represent the distance between the two microcell centers in the horizontal and the vertical directions in units of printer addressable pixels.

The parameters of the halftone screen are the printer resolution ρ in units of dots per inch (dpi), the screen frequency F_s in units of cycles per inch (cpi), and the screen angle α in degrees. The tile vectors $\mathbf{v_1}$ and $\mathbf{v_2}$ can be expressed in terms of the parameters as follows:

$$\begin{aligned} \mathbf{v_1} &= \begin{bmatrix} v_{11} & v_{12} \end{bmatrix} = \frac{\rho}{F_s} \begin{bmatrix} \cos\alpha & \sin\alpha \end{bmatrix} = \|\mathbf{v_1}\| \begin{bmatrix} \cos\alpha & \sin\alpha \end{bmatrix} \\ \mathbf{v_2} &= \begin{bmatrix} -v_{12} & v_{11} \end{bmatrix} = \frac{\rho}{F_s} \begin{bmatrix} -\sin\alpha & \cos\alpha \end{bmatrix} = \|\mathbf{v_1}\| \begin{bmatrix} -\sin\alpha & \cos\alpha \end{bmatrix}. \end{aligned} \tag{2.2}$$

The smallest element in the imaging system is one printer-addressable pixel, therefore it is easy to work with non-negative integer values for v_{11} and v_{12}, i.e. $v_{11}, v_{12} \in \mathbb{Z}^*$. The screens with periodicity matrices whose entries are all integers are called *regular*. The range of regular screens is limited within the common industry screen frequency range of $150 < F_s < 300$ lpi [1]. This frequency range is based on the capabilities of the typical digital marking engine processes and the visibility of the resulting halftone textures at normal viewing distances. In order to design screens which are more smooth and whose parameters are closer to the target screen frequency and screen angle, it is necessary to enlarge the space of available screen geometries with the use of *irregular* geometries. This is achieved by allowing v_{11}, v_{12}

to be rational numbers, i.e. $v_{11}, v_{12} \in \mathbb{Q}$. Hence, the periodicity matrix tile vectors take the form:

$$\begin{aligned} \mathbf{v_1} &= \begin{bmatrix} v_{11} & v_{12} \end{bmatrix} = \begin{bmatrix} \frac{p_1}{q_1} & \frac{p_2}{q_2} \end{bmatrix}, \\ \mathbf{v_2} &= \begin{bmatrix} -v_{12} & v_{11} \end{bmatrix} = \begin{bmatrix} -\frac{p_2}{q_2} & \frac{p_1}{q_1} \end{bmatrix}, \end{aligned} \tag{2.3}$$

where $p_1, q_1, p_2, q_2 \in \mathbb{Z}^*$, $\mathbb{Z}^*$ is the set of the non-negative integers.

In order to obtain the non-negative integer values p_1, q_1, p_2, q_2, which will produce the tile vectors closest to the target screen parameters, we propose to perform a search over the space of realizable periodicity matrices [65]. We start with an irrational target tile vector $\mathbf{v_1^{target}}$ obtained using Eq. 2.2, and search over the candidate rational tile vectors of the form $\mathbf{v_1} = \begin{bmatrix} \frac{p_1}{q_1} & \frac{p_2}{q_2} \end{bmatrix}$, which minimize the Euclidean distance between the target tile vectors and candidate tile vectors. The space of candidate screen tile vectors includes all possible combinations of $\mathbf{\frac{p_1}{q_1}}$ and $\mathbf{\frac{p_2}{q_2}}$ within the desired limit values p_{limit}, and q_{limit}. Thus, for each fixed value of q_{limit}, we search over all $1 \leq p_1 \leq p_{limit}$, and $1 \leq p_2 \leq p_{limit}$, $1 \leq q_1 \leq q_{limit}$, and $1 \leq q_2 \leq q_{limit}$ to find the best fit to the target set. The block diagram for the procedure is shown in Fig 2.3 [65]. If for any given (q_1, q_2) combination the best fit may be for some $q_1' \leq q_1$, and/or $q_2' \leq q_2$, such combination of (q_1, q_2) will not be taken into account.

After obtaining the candidate screen tile vectors, their screen parameters can be evaluated. The screen parameters that we are most interested in are summarized in Table 2.1. In most screening implementations, the screen is usually stored in a form of 2-D rectangular array. The rectangular region that can represent a periodic, clustered-dot halftone screen is not unique. Holladay [62], [63] was one of the earliest researchers to work on the problem of determining a rectangular region to represent a clustered-dot periodic screen. Baqai et al. [48], Lin et al. [61] and Lee et al. [47] followed his approach. However, they only considered regular screens. Lin, Baqai and Lee referred to their version of the rectangular region as the Basic Screen Block (BSB). Tang et al. [49] extended it to the case of irregular screens. It was concluded that to obtain the Basic Screen Block size for irregular screens, we first obtain a supercell matrix

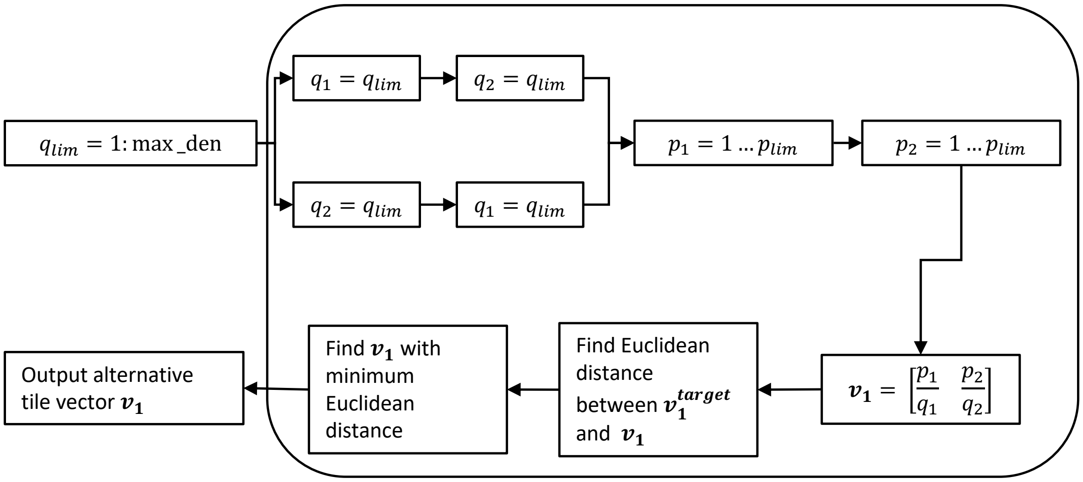

$q_{lim} = 1\text{: max\_den}$
$q_1 = q_{lim}$
$q_2 = q_{lim}$
$q_2 = q_{lim}$
$q_1 = q_{lim}$
$p_1 = 1 \ldots p_{lim}$
$p_2 = 1 \ldots p_{lim}$
$\boldsymbol{v_1} = \begin{bmatrix} \frac{p_1}{q_1} & \frac{p_2}{q_2} \end{bmatrix}$
Find Euclidean distance between $\boldsymbol{v_1^{target}}$ and $\boldsymbol{v_1}$
Find $\boldsymbol{v_1}$ with minimum Euclidean distance
Output alternative tile vector $\boldsymbol{v_1}$

Fig. 2.3.: Search algorithm for generating candidate screen tile-vectors.

$\mathbf{S} = \begin{bmatrix} s_{11} & -s_{12} \\ s_{12} & s_{11} \end{bmatrix}$ using the diagonal repetition number matrix $\mathbf{M}$. The elements of the repetition matrix $\mathbf{M}$ on the diagonal are equal to the least common multiple of the denominator of the elements of the periodicity matrix. For example, if the periodicity matrix is $\mathbf{V} = \begin{bmatrix} \frac{9}{5} & -\frac{18}{5} \\ \frac{18}{5} & \frac{9}{5} \end{bmatrix}$, then the repetition matrix is $\mathbf{M} = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$ and the supercell periodicity matrix is $\mathbf{S} = \mathbf{VM} = \begin{bmatrix} \frac{9}{5} & -\frac{18}{5} \\ \frac{18}{5} & \frac{9}{5} \end{bmatrix} \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix} = \begin{bmatrix} 9 & -18 \\ 18 & 9 \end{bmatrix}$. To determine the Basic Screen Block size B, we use

$$B = \left| \frac{\det \mathbf{S}}{\gcd(s_{11}, s_{12})} \right|, \tag{2.4}$$

Table 2.1.: Basic screen parameters

Symbol	**Definition**
$\mathbf{v_1}$, $\mathbf{v_2}$	the tile vectors
α	the screen angle in degrees
F_s	the screen frequency in cycles per inch (commonly called lines per inch)
ρ	the printer resolution in printer-addressable pixels per inch (commonly called dots per inch (dpi)) ($\in \mathbb{R}$)
$\mathbf{M}$	the repetition matrix (diagonal)
$\mathbf{S}$	the supercell periodicity matrix
H_{block}, W_{block}	the basis screen block height and width

and obtain the Basic Screen Block of size $B = 45$. Since we only work with orthogonal and square geometries, the BSB height and width are the same.

2.3.2 Experimental results

In this section, examples of obtaining the set of candidate screen tile vectors for the target screen parameters of 180 lpi, 812.8 dpi, and 15°, 45°, and 75°, are demonstrated. For the first set of screen parameters (15°, 180 lpi, 812.8 dpi), we start by obtaining the target tile vectors for the screen angle of $\mathbf{v_1^{target}}$ and $\mathbf{v_2^{target}}$ using Eq. 2.2:

$$\mathbf{v_1^{target}} = \begin{bmatrix} v_{11} & v_{12} \end{bmatrix} = \frac{\rho}{F_s} \begin{bmatrix} \cos\alpha & \sin\alpha \end{bmatrix} = \frac{812.8}{180} \begin{bmatrix} \cos 15^\circ & \sin 15^\circ \end{bmatrix} = \begin{bmatrix} 4.36 & 1.17 \end{bmatrix}, \tag{2.5}$$

$$\mathbf{v_2^{target}} = \begin{bmatrix} -v_{12} & v_{11} \end{bmatrix} = \begin{bmatrix} -1.17 & 4.36 \end{bmatrix}.$$

The set of candidate screen tile vectors when the maximum denominator is set to 9 and the maximum value for a numerator is set to 50 is shown in Fig. 2.4 a). Thus, for each fixed value of q_{limit}, where $1 \leq q_{limit} \leq 9$, we search over all $1 \leq p_1 \leq 50$, and $1 \leq p_2 \leq 50$, $1 \leq q_1 \leq q_{limit}$, and $1 \leq q_2 \leq q_{limit}$ to find the best fit to the target set. Therefore, each row in the table corresponds to setting the limit of the denominator to q_{limit}, whereas columns 1 through 6 contain the candidate tile vectors of the form $\mathbf{v_1} = \begin{bmatrix} \frac{p_1}{q_1} & \frac{p_2}{q_2} \end{bmatrix}$, which were obtained by minimizing the distance between $\mathbf{v_1}$ and $\mathbf{v_1^{target}}$.

Similarly, the sets of candidate screen tile vectors for the target screen parameters of (45°, 180 lpi, 812.8 dpi) and (75°, 180 lpi, 812.8 dpi) are demonstrated in Figs. 2.4 b) and c). With the target screen parameters (45°, 180 lpi, 812.8 dpi), the target tile vector is $\mathbf{v_1^{target}} = \begin{bmatrix} 3.19 & 3.19 \end{bmatrix}$. Whereas, with the target screen parameters (75°, 180 lpi, 812.8 dpi), the target tile vector is $\mathbf{v_1^{target}} = \begin{bmatrix} 1.17 & 4.36 \end{bmatrix}$.

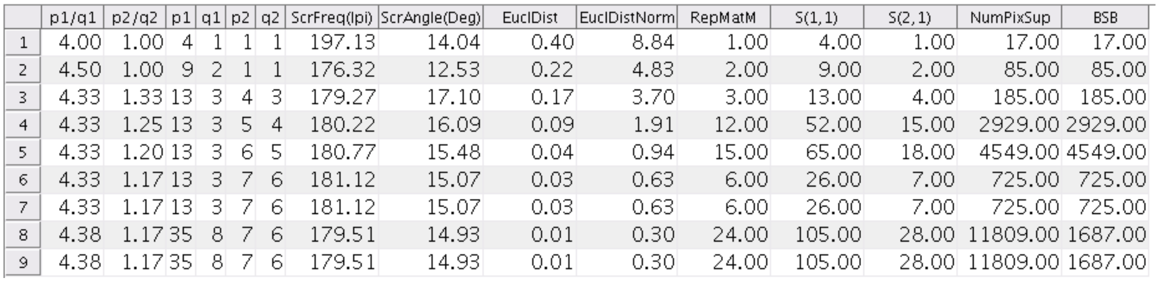
From the results in Fig. 2.4 we can conclude that as we increase the limit for maximum denominator, i.e. as we go from row 1 to row 9, the Euclidean distance between the target tile vector and the candidate tile vector gets smaller. Hence, the screen angle and the screen frequency of the candidate tile vectors become closer to the target screen angle and the target screen frequency. However, their Basic Screen Block size may get bigger, which may result in some undesirable artifacts. In order to measure image fluctuation solely based on the periodicity matrix, we looked into the theoretical model for halftone image fluctuation in the Fourier domain.

2.4 Theoretical Model for Halftone Image Fluctuation

2.4.1 Fourier spectra of continuous-space analog and digital halftones

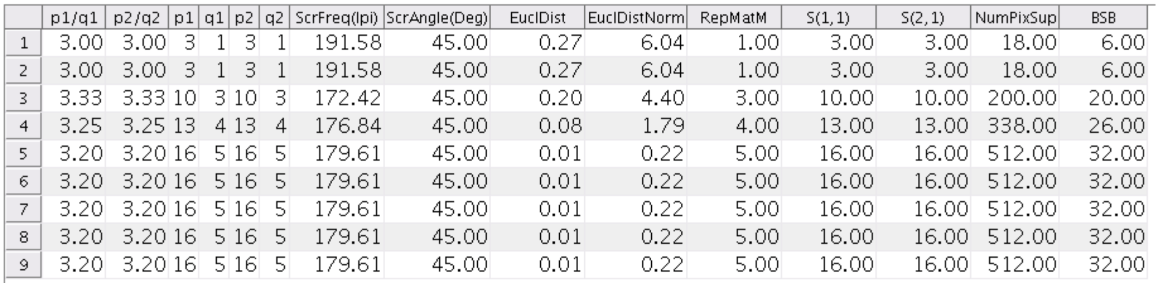
For our analysis, we use the idealized halftone patterns consisting of circular dots for the analog halftones. Hence, our continuous-space halftone image is defined as

$$g(\mathbf{x}) = \sum_{\mathbf{k} \in Z^2} \operatorname{circ}\left(\mathbf{M}\left(\mathbf{x} - R\mathbf{N}\mathbf{k}\right)\right), \tag{2.6}$$



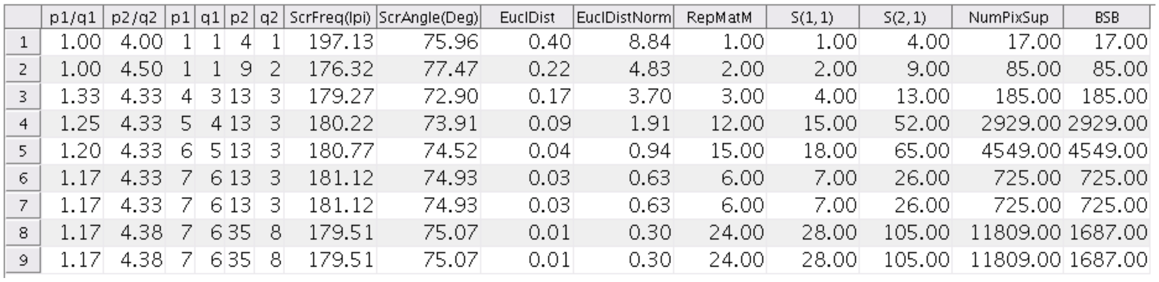
	p1/q1	p2/q2	p1	q1	p2	q2	ScrFreq(lpi)	ScrAngle(Deg)	EuclDist	EuclDistNorm	RepMatM	S(1,1)	S(2,1)	NumPixSup	BSB
1	4.00	1.00	4	1	1	1	197.13	14.04	0.40	8.84	1.00	4.00	1.00	17.00	17.00
2	4.50	1.00	9	2	1	1	176.32	12.53	0.22	4.83	2.00	9.00	2.00	85.00	85.00
3	4.33	1.33	13	3	4	3	179.27	17.10	0.17	3.70	3.00	13.00	4.00	185.00	185.00
4	4.33	1.25	13	3	5	4	180.22	16.09	0.09	1.91	12.00	52.00	15.00	2929.00	2929.00
5	4.33	1.20	13	3	6	5	180.77	15.48	0.04	0.94	15.00	65.00	18.00	4549.00	4549.00
6	4.33	1.17	13	3	7	6	181.12	15.07	0.03	0.63	6.00	26.00	7.00	725.00	725.00
7	4.33	1.17	13	3	7	6	181.12	15.07	0.03	0.63	6.00	26.00	7.00	725.00	725.00
8	4.38	1.17	35	8	7	6	179.51	14.93	0.01	0.30	24.00	105.00	28.00	11809.00	1687.00
9	4.38	1.17	35	8	7	6	179.51	14.93	0.01	0.30	24.00	105.00	28.00	11809.00	1687.00

(a) The set of the candidate screen tile-vectors for 180 lpi, 15° and 812.8 dpi. The target tile vector is $\mathbf{v_1^{target}} = \begin{bmatrix}4.36 & 1.17\end{bmatrix}$.



	p1/q1	p2/q2	p1	q1	p2	q2	ScrFreq(lpi)	ScrAngle(Deg)	EuclDist	EuclDistNorm	RepMatM	S(1,1)	S(2,1)	NumPixSup	BSB
1	3.00	3.00	3	1	3	1	191.58	45.00	0.27	6.04	1.00	3.00	3.00	18.00	6.00
2	3.00	3.00	3	1	3	1	191.58	45.00	0.27	6.04	1.00	3.00	3.00	18.00	6.00
3	3.33	3.33	10	3	10	3	172.42	45.00	0.20	4.40	3.00	10.00	10.00	200.00	20.00
4	3.25	3.25	13	4	13	4	176.84	45.00	0.08	1.79	4.00	13.00	13.00	338.00	26.00
5	3.20	3.20	16	5	16	5	179.61	45.00	0.01	0.22	5.00	16.00	16.00	512.00	32.00
6	3.20	3.20	16	5	16	5	179.61	45.00	0.01	0.22	5.00	16.00	16.00	512.00	32.00
7	3.20	3.20	16	5	16	5	179.61	45.00	0.01	0.22	5.00	16.00	16.00	512.00	32.00
8	3.20	3.20	16	5	16	5	179.61	45.00	0.01	0.22	5.00	16.00	16.00	512.00	32.00
9	3.20	3.20	16	5	16	5	179.61	45.00	0.01	0.22	5.00	16.00	16.00	512.00	32.00

(b) The set of the candidate screen tile-vectors for 180 lpi, 45° and 812.8 dpi. The target tile vector is $\mathbf{v_1^{target}} = \begin{bmatrix}3.19 & 3.19\end{bmatrix}$.



	p1/q1	p2/q2	p1	q1	p2	q2	ScrFreq(lpi)	ScrAngle(Deg)	EuclDist	EuclDistNorm	RepMatM	S(1,1)	S(2,1)	NumPixSup	BSB
1	1.00	4.00	1	1	4	1	197.13	75.96	0.40	8.84	1.00	1.00	4.00	17.00	17.00
2	1.00	4.50	1	1	9	2	176.32	77.47	0.22	4.83	2.00	2.00	9.00	85.00	85.00
3	1.33	4.33	4	3	13	3	179.27	72.90	0.17	3.70	3.00	4.00	13.00	185.00	185.00
4	1.25	4.33	5	4	13	3	180.22	73.91	0.09	1.91	12.00	15.00	52.00	2929.00	2929.00
5	1.20	4.33	6	5	13	3	180.77	74.52	0.04	0.94	15.00	18.00	65.00	4549.00	4549.00
6	1.17	4.33	7	6	13	3	181.12	74.93	0.03	0.63	6.00	7.00	26.00	725.00	725.00
7	1.17	4.33	7	6	13	3	181.12	74.93	0.03	0.63	6.00	7.00	26.00	725.00	725.00
8	1.17	4.38	7	6	35	8	179.51	75.07	0.01	0.30	24.00	28.00	105.00	11809.00	1687.00
9	1.17	4.38	7	6	35	8	179.51	75.07	0.01	0.30	24.00	28.00	105.00	11809.00	1687.00

(c) The set of the candidate screen tile-vectors for 180 lpi, 75° and 812.8 dpi. The target tile vector is $\mathbf{v_1^{target}} = \begin{bmatrix}1.17 & 4.36\end{bmatrix}$.

Fig. 2.4.: The set of the candidate screen tile-vectors for 180 lpi, 812.8 dpi, and three screen angles 15°, 45°, and 75°. Each row in the tables corresponds to setting the limit of the denominator to q_{limit}, where $1 \leq q_{limit} \leq 9$, and $p_{limit} = 50$. Columns 1 through 6 contain the candidate tile vectors of the form $\mathbf{v_1} = \begin{bmatrix}\frac{p_1}{q_1} & \frac{p_2}{q_2}\end{bmatrix}$, which were obtained by minimizing the Euclidean distance between $\mathbf{v_1}$ and $\mathbf{v_1^{target}}$ (column 9). Column 8 contains the values of Euclidean distance normalized by the by the length of the target tile vector. Columns 7-8, 10-14 contain the values for the parameters presented in Table 2.1.

where the parameter R is the horizontal and vertical distance between printer-addressable pixels in units of inches. The matrix $\mathbf{M}$ is a diagonal matrix defined as diag$\left\{\frac{1}{d}, \frac{1}{d}\right\}$, where the parameter d is a diameter of each circular dot. The CSFT of $g\left(\mathbf{x}\right)$ is given by

$$G\left(\mathbf{u}\right) = \frac{\mathrm{jinc}\left(\mathbf{M}^{-T}\mathbf{u}\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\frac{1}{R}\mathbf{N^{-T}k}\right), \tag{2.7}$$

where the CSFT of the circ function is the jinc function defined as $\mathrm{jinc}\left(\mathbf{u}\right) = \frac{J_1\left(\pi\sqrt{u^2+v^2}\right)}{2\sqrt{u^2+v^2}}$, where J_1 is the Bessel function of the first kind with order 1.

For the digital halftones, we use sampled versions of the analog halftones. Therefore, in order to obtain the digital rendering of the continuous-space halftone $g\left(\mathbf{x}\right)$, we sample $g\left(\mathbf{x}\right)$ on the printer lattice, and obtain $g_s\left(\mathbf{x}\right) = \mathrm{comb}_{RR}\left[g\left(\mathbf{x}\right)\right]$. Then, we convolve $g_s\left(\mathbf{x}\right)$ with the printer dot profile function $\mathrm{rect}\left(\frac{\mathbf{x}}{R}\right)$, which has CSFT $R^2\mathrm{sinc}\left(R\mathbf{u}\right)$. The CSFT of the resulting printed digital halftone image $g_{\mathrm{printer}}\left(\mathbf{x}\right)$ is given by

$$\begin{aligned} G_{\mathrm{printer}}\left(\mathbf{u}\right) = \mathrm{sinc}\left(R\mathbf{u}\right)\sum_{\mathbf{m}\in Z^2}\frac{\mathrm{jinc}\left(\mathbf{M}^{-T}\left(\mathbf{u}-\mathbf{Vm}\right)\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|} \\ \times\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\mathbf{Vm}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right), \end{aligned} \tag{2.8}$$

where $\mathbf{V} = \mathrm{diag}\left\{\frac{1}{R}, \frac{1}{R}\right\}$.

Lastly, in order to obtain the perceived analog and digital rendering continuous-space halftone images, we convolve them with the point-spread function of the human visual system (HVS). In the Fourier domain, we thus have

$$\begin{aligned}\tilde{G}\left(\mathbf{u}\right) &= H_{\mathrm{HVS}}\left(\mathbf{u}\right) G\left(\mathbf{u}\right), \\ &= H_{\mathrm{HVS}}\left(\mathbf{u}\right) \frac{\operatorname{jinc}\left(\mathbf{M}^{-T}\mathbf{u}\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|} \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u} - \frac{1}{R}\mathbf{N}^{-\mathbf{T}}\mathbf{k}\right), \\ \tilde{G}_{\mathrm{printer}}\left(\mathbf{u}\right) &= H_{\mathrm{HVS}}\left(\mathbf{u}\right) G_{\mathrm{printer}}\left(\mathbf{u}\right), \\ &= H_{\mathrm{HVS}}\left(\mathbf{u}\right) \operatorname{sinc}\left(R\mathbf{u}\right) \sum_{\mathbf{m}\in Z^2} \frac{\operatorname{jinc}\left(\mathbf{M}^{-T}\left(\mathbf{u} - \mathbf{V}\mathbf{m}\right)\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|} \\ &\times \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m} - \frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right),\end{aligned} \tag{2.9}$$

Equation (2.8) contains two summations over infinite limits. In order to evaluate these summations and to determine which spectral components contribute to the perceived halftone image, we need to determine limits for both **m** and **k**. That is the topic of the following two subsections.

Determining Limits of Summation for m

We start by re-writing (2.8) as

$$\tilde{G}_{\mathrm{printer}}\left(\mathbf{u}\right) = \left(\frac{1}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\right)\left(\frac{\pi}{4}\right)\tilde{G}^{\mathrm{norm}}_{\mathrm{printer}}\left(\mathbf{u}\right), \tag{2.10}$$

where

$$\tilde{G}^{\mathrm{norm}}_{\mathrm{printer}}\left(\mathbf{u}\right) = \bar{H}_{\mathrm{PPD}}\left(\mathbf{u}\right) \sum_{\mathbf{m}\in Z^2} \bar{H}_{\mathrm{AHI}}\left(\mathbf{u} - \mathbf{V}\mathbf{m}\right), \tag{2.11}$$

$$\begin{aligned}\bar{H}_{\mathrm{PPD}}\left(\mathbf{u}\right) &= \bar{H}_{\mathrm{HVS}}\left(\mathbf{u}\right)\bar{H}_{\mathrm{PD}}\left(\mathbf{u}\right) \\ &= \bar{H}_{\mathrm{HVS}}\left(\mathbf{u}\right)\operatorname{sinc}\left(R\mathbf{u}\right),\end{aligned} \tag{2.12}$$

and

$$\bar{H}_{\mathrm{AHI}}\left(\mathbf{u}\right) = \bar{H}_{\mathrm{AHD}}\left(\mathbf{u}\right) \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u} - \frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right), \tag{2.13}$$

where

$$\bar{H}_{\text{AHD}}\left(\mathbf{u}\right) = \frac{4}{\pi}\text{jinc}\left(\mathbf{M}^{-T}\mathbf{u}\right). \tag{2.14}$$

Equations (2.10)-(2.14) contain several new acronyms, which are defined in Table 2.2. We use the overbar to denote the fact that the corresponding quantity is upper-bounded by its value at the origin ($\mathbf{u} = \mathbf{0}$), which is unity. Thus, $\left|\bar{H}_{\text{PPD}}(\mathbf{u})\right| \leq \left|\bar{H}_{\text{PPD}}(\mathbf{0})\right| = 1$ and $\left|\bar{H}_{\text{AHD}}(\mathbf{u})\right| \leq \left|\bar{H}_{\text{AHD}}(\mathbf{0})\right| = 1$.

Table 2.2.: Acronyms and their definitions.

Acronym	**Definition of Term**
PD	Printer Dot
PPD	Perceived Printer Dot
AHI	Analog Halftone Image
AHD	Analog Halftone Dot
BPPD	Bandwidth of Perceived Printer Dot
BAHI	Bandwidth of Analog Halftone Image
PROD	Product

We want to determine the limits of summation for $\mathbf{m}$ and $\mathbf{k}$ that are as small as possible, yet provide satisfactory accuracy. Larger limits lead to more accurate results, and smaller limits reduce computation. We wish to determine the contribution of these terms to the region of the frequency domain that is visible to the human viewer. Equation (2.11) shows that the normalized spectrum $\tilde{G}^{\text{norm}}_{\text{printer}}(\mathbf{u})$ of the printed halftone image consists of replications of the analog halftone image spectrum $\bar{H}_{\text{AHI}}(\mathbf{u})$ displaced to frequencies $\mathbf{u} = \mathbf{Vm}$ and weighted by the spectrum $\bar{H}_{\text{PPD}}(\mathbf{u})$ of the perceived printer dot. Figure 2.5 depicts the normalized spectrum of the digital halftone.

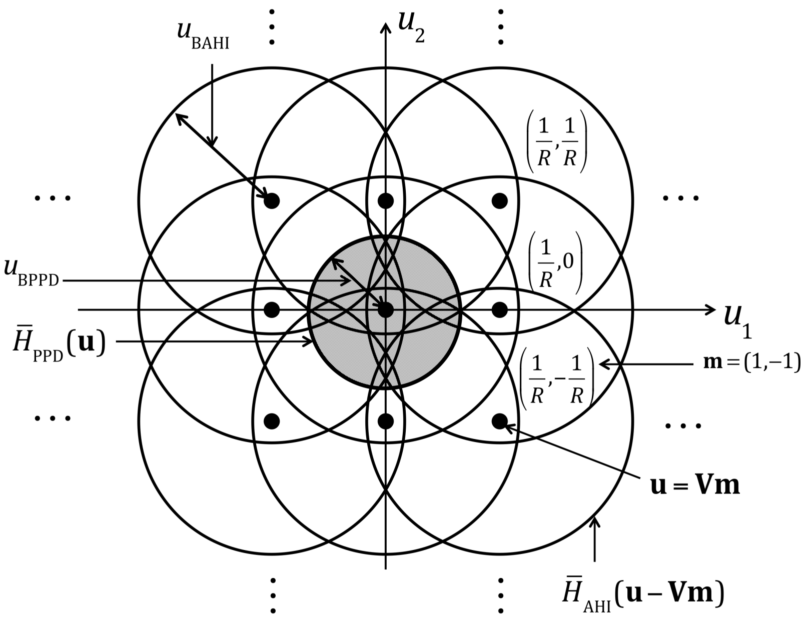

u_{BAHI}
u_2
$\left(\frac{1}{R},\frac{1}{R}\right)$
u_{BPPD}
$\left(\frac{1}{R},0\right)$
$\bar{H}_{\mathrm{PPD}}(\mathbf{u})$
u_1
$\left(\frac{1}{R},-\frac{1}{R}\right)$
$\mathbf{m}=(1,-1)$
$\mathbf{u}=\mathbf{Vm}$
$\bar{H}_{\mathrm{AHI}}(\mathbf{u}-\mathbf{Vm})$

Fig. 2.5.: Normalized spectrum of the digital halftone.

Next, we upper-bound the strength of the impulses in (2.11)-(2.13). To do that, we define a new function

$$\bar{H}_{\mathrm{PROD}}\left(\mathbf{u};\mathbf{u}_{\mathrm{displace}}\right)=\bar{H}_{\mathrm{PPD}}\left(\mathbf{u}\right)\bar{H}_{\mathrm{AHD}}\left(\mathbf{u}-\mathbf{u}_{\mathrm{displace}}\right) \tag{2.15}$$

as the upper bound of the contribution at frequency $\mathbf{u}$ of a replication of $\bar{H}_{\mathrm{AHD}}\left(\mathbf{u}\right)$ centered at $\mathbf{u}_{\mathrm{displace}}$. We define the threshold δ to be the smallest value for a spectral term that we must include in the summations given by (2.11) and (2.13). When we choose the limits for our summations over $\mathbf{m}$ and $\mathbf{k}$, we may also include terms that have magnitude $\leq \delta$. However, we will never neglect a term that is $\geq \delta$.

Since the absolute value of both terms in the expression for $\bar{H}_{\mathrm{PROD}}(\mathbf{u};\mathbf{u}_{\mathrm{displace}})$ given by (2.15) are bounded from above by 1, $\left|\bar{H}_{\mathrm{PROD}}(\mathbf{u};\mathbf{u}_{\mathrm{displace}})\right| \leq \delta$, whenever either $\bar{H}_{\mathrm{PPD}}(\mathbf{u}) \leq \delta$, or $\bar{H}_{\mathrm{AHD}}(\mathbf{u};\mathbf{u}_{\mathrm{displace}}) \leq \delta$. Thus, we define

$$\begin{aligned} u_{\mathrm{BPPD}} &= \min\{u_{\mathrm{BPPD}} : \left|\bar{H}_{\mathrm{PPD}}(\mathbf{u})\right| < \delta, \forall\, |\mathbf{u}| > u_{\mathrm{BPPD}}\}, \\ u_{\mathrm{BAHI}} &= \min\{u_{\mathrm{BAHI}} : \left|\bar{H}_{\mathrm{AHI}}(\mathbf{u})\right| < \delta, \forall\, |\mathbf{u}| > u_{\mathrm{BAHI}}\}. \end{aligned} \tag{2.16}$$

Combining the two facts above, we conclude that we do not need to consider values for $\mathbf{m}$, for which $\mathbf{u}_{\text{displace}} > \mathbf{u}_{\text{BPPD}} + \mathbf{u}_{\text{BAHI}}$. Therefore, we set $\mathbf{u}_{\text{displace}}^{\text{max\_initial}} = \mathbf{u}_{\text{BPPD}} + \mathbf{u}_{\text{BAHI}}$. However, this results in a range of summation for $\mathbf{m}$ that is too conservative. We seek a tighter upper bound for $\mathbf{u}_{\text{displace}}^{\text{max}}$. Therefore, for each value of $0 \leq \mathbf{u}_{\text{displace}} \leq \mathbf{u}_{\text{displace}}^{\text{max\_initial}}$, we determine the value

$$\bar{H}_{\text{PROD}}^{\text{max}} \left(\mathbf{u}_{\text{displace}} \right) = \max_{0 \leq \mathbf{u} \leq \mathbf{u}^{\text{max}}} \bar{H}_{\text{PROD}} \left(\mathbf{u}; \mathbf{u}_{\text{displace}} \right), \tag{2.17}$$

where $\mathbf{u}^{\text{max}} = \lceil u_{\text{BPPD}} \rceil$. Then, we find the minimum value for $\mathbf{u}_{\text{displace}}$ for which

$$\mathbf{u}_{\text{displace}}^{\text{max\_final}} = \min\{ \mathbf{u}_{\text{displace}} : \left| \bar{H}_{\text{PROD}}^{\text{max}} \left(\mathbf{u}_{\text{displace}} \right) \right| < \delta \}. \tag{2.18}$$

Finally, the limits for $\mathbf{m}$ can be determined from $\mathbf{m} \leq \left\lceil \mathbf{V}^{-1} \mathbf{u}_{\text{displace}}^{\text{max\_final}} \right\rceil$. Here scalar operations or relations are applied to each element of the vectors. Having determined the $\mathbf{m}$ limits in the preceding subsection, we next need to determine, for each fixed value $\mathbf{m} = \mathbf{m}_0$ within those limits, what are the minimum limits of the summation over $\mathbf{k}$ in (2.11)-(2.14) that will satisfy our criterion on the spectral magnitude of each term for it to be included in the summation.

Determining Limits of Summation for k

For the $\mathbf{k}$ limits, we consider the normalized spectrum of the analog halftone image illustrated in Fig. 2.6. There are two factors that delineate the cases to be considered for determining the $\mathbf{k}$-limits. The first factor is that the axes defined by $\mathbf{n_1}'$ and $\mathbf{n_2}'$, shown in Fig. 2.6, can each intersect the u_1 or u_2 axes, respectively, at a positive value or at a negative value. The second factor is that the axes defined by $\mathbf{n_1}'$ and $\mathbf{n_2}'$ can each intersect or not intersect the band region of the perceived printer dot with radius $\mathbf{u}_{\text{BPPD}}$. This is the region that limits the set of frequency components that will be visible to the human viewer, as indicated by (2.13), (2.14). We determined that there are 16 cases for the two factors above. The first 4 cases are presented in Fig. 2.7. And the other 12 cases are defined analogously by searching around the circle with radius $\mathbf{u}_{\text{BPPD}}$, and keeping either the $\mathbf{n_1}'$ or $\mathbf{n_2}'$ axis fixed, while moving along the

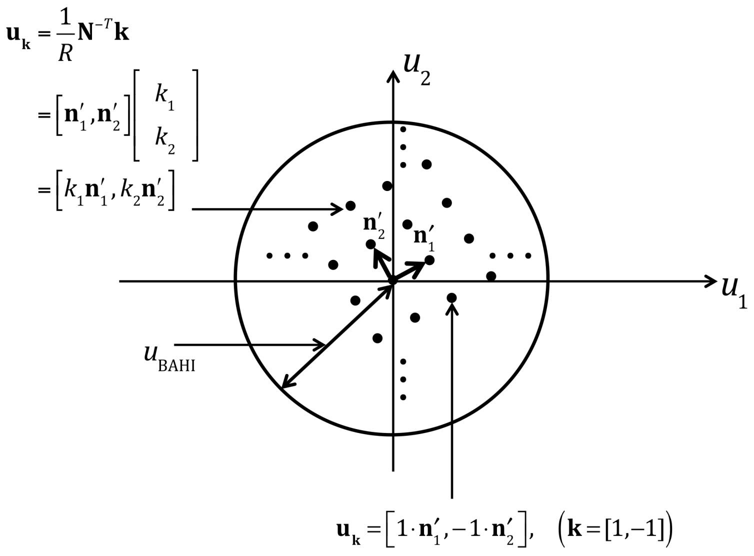

$\mathbf{u}_{\mathbf{k}} = \frac{1}{R}\mathbf{N}^{-T}\mathbf{k}$
$= [\mathbf{n}'_1, \mathbf{n}'_2]\begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$
$= [k_1\mathbf{n}'_1, k_2\mathbf{n}'_2]$
u_2
$\mathbf{n}'_2$
$\mathbf{n}'_1$
u_1
u_{BAHI}
$\mathbf{u}_{\mathbf{k}} = [1 \cdot \mathbf{n}'_1, -1 \cdot \mathbf{n}'_2]$, $(\mathbf{k} = [1, -1])$

Fig. 2.6.: Normalized spectrum of the analog halftone image. The dots indicate the location of impulses.

opposite axis. Hence, out of the 16 cases, there are 13 unique cases, and 3 non-unique cases that occur when making a switch from one direction to the other. Based on these results, we came up with a general approach for determining the **k**-limits.

2.4.2 Experimental results

The perceived analog rendering continuous-space halftone images in the Fourier domain have the following form:

$$\tilde{G}^{\text{norm}}(\mathbf{u}) = H_{\text{HVS}}(\mathbf{u})\,\text{jinc}\left(\mathbf{M}^{-T}\mathbf{u}\right)\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u} - \frac{1}{R}\mathbf{N}^{-\mathbf{T}}\mathbf{k}\right), \tag{2.19}$$

whereas the perceived digital rendering continuous-space halftone images in the Fourier domain have the following form:

$$\begin{aligned}\tilde{G}^{\text{norm}}_{\text{printer}}(\mathbf{u}) = H_{\text{HVS}}(\mathbf{u})\,\text{sinc}\,(R\mathbf{u})\sum_{\mathbf{m}\in Z^2}\text{jinc}\left(\mathbf{M}^{-T}(\mathbf{u} - \mathbf{V}\mathbf{m})\right) \\ \times\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u} - \mathbf{V}\mathbf{m} - \frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right).\end{aligned} \tag{2.20}$$

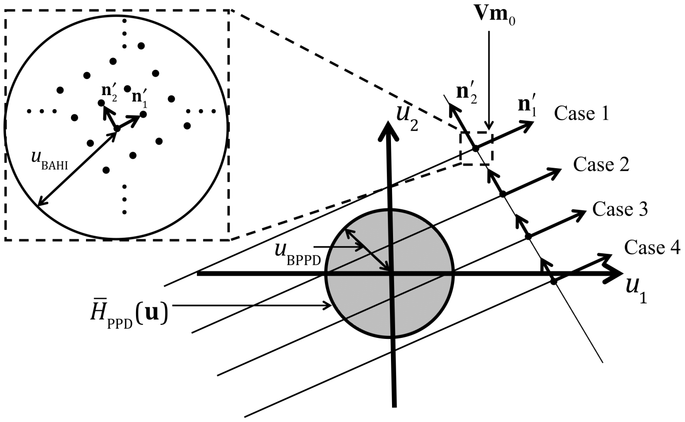


Fig. 2.7.: The first 4 cases for determining the **k**-limits. We keep the $\mathbf{n_2}'$ axis fixed, and move the $\mathbf{n_1}'$: (1) $\mathbf{n_1}'$ intersects u_2 at a positive value and does not intersect the shaded circle; (2) $\mathbf{n_1}'$ intersects u_2 at a positive value and does intersect the shaded circle; (3) $\mathbf{n_1}'$ intersects u_2 at a negative value and does intersect the shaded circle; (4) $\mathbf{n_1}'$ intersects u_2 at a negative value and does not intersect the shaded circle.

We use the expressions in (2.19) and (2.20) in order to determine the locations of all the frequency components, and their corresponding amplitudes. We generate tables of all the data in ascending order with respect to the distance from the origin. Here is the approach for generating the tables of frequency components:

1. Start with the main CSFT equation ((2.19) or (2.20)) and obtain the limits for **m** and **k** as described previously.

2. Generate a table of values of all the frequency components, and their magnitudes for a newly defined region of **m** and **k**.

3. Determine which **m** and **k** result in the same $\mathbf{u_1}$ and $\mathbf{u_2}$, and update the contributions of the desired sources of attenuation.

4. Sort all the obtained values of the locations of frequency components in ascending order in order to determine what **m** and **k** give the additional frequency

components in the digital rendering spectrum that were not present in the analog rendering spectrum.

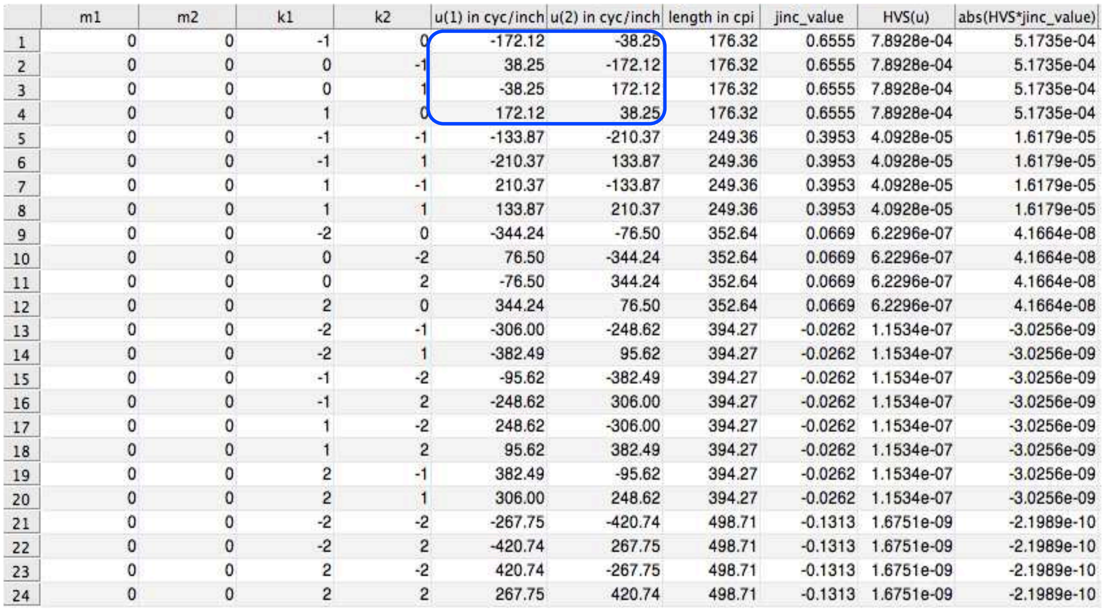
In this section, we use an example of the geometry with periodicity matrix $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$. The limits for $\mathbf{m}$ values for the digital rendering spectrum have turned out to be $-51 \leq m_1 \leq 51, m_1 \in \mathbb{Z}$ and $-51 \leq m_2 \leq 51, m_2 \in \mathbb{Z}$. The limits for $\mathbf{k}$ for the analog rendering spectrum are $-2 \leq k_1 \leq 2, k_1 \in \mathbb{Z}$ and $-2 \leq k_2 \leq 2, k_2 \in \mathbb{Z}$, whereas for the digital rendering the limits for $\mathbf{k}$ are unique for each pair of $\mathbf{m}$ values.

The table of the frequency components and their corresponding amplitudes for the *analog* rendering table is provided in the Fig. 2.8. The table of the frequency components and their corresponding amplitudes for the *digital* rendering table is provided in the Figs. 2.9 and 2.10. From such Fourier analysis, we can identify two sources of attenuation of the frequency domain alias terms. The first is the Fourier spectrum of the ideal halftone dot shape that would be rendered by an analog device. The second is the Fourier spectrum corresponding to the shape of the printer-addressable pixels denoted by digital rendering [45]. Due to the relatively small size of these pixels, this effect is much weaker than the attenuation due to the spectrum of the ideal halftone dot shape. Taking into account both the locations and amplitudes of each Fourier component with a visual weighting function allows us to predict the perceived lack of smoothness in the printed halftone pattern. In the next section, we discuss how we use the information in the tables in order to obtain the measure of the perceived image fluctuation.

2.5 Evaluation of Image Fluctuation with the Use of Analog and Digital Halftones Spectra and Computation of Ratios.

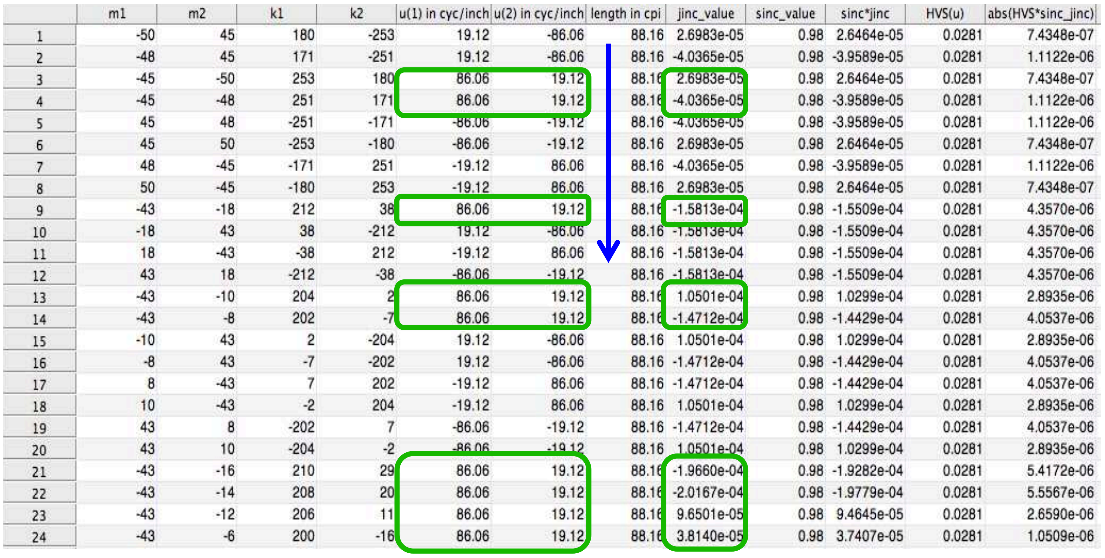
2.5.1 Computation of ratios using the analog and digital halftones spectra

As mentioned before, if the elements of the periodicity matrix are integers, then the screen is regular, and if some or all elements of the periodicity matrix are non-



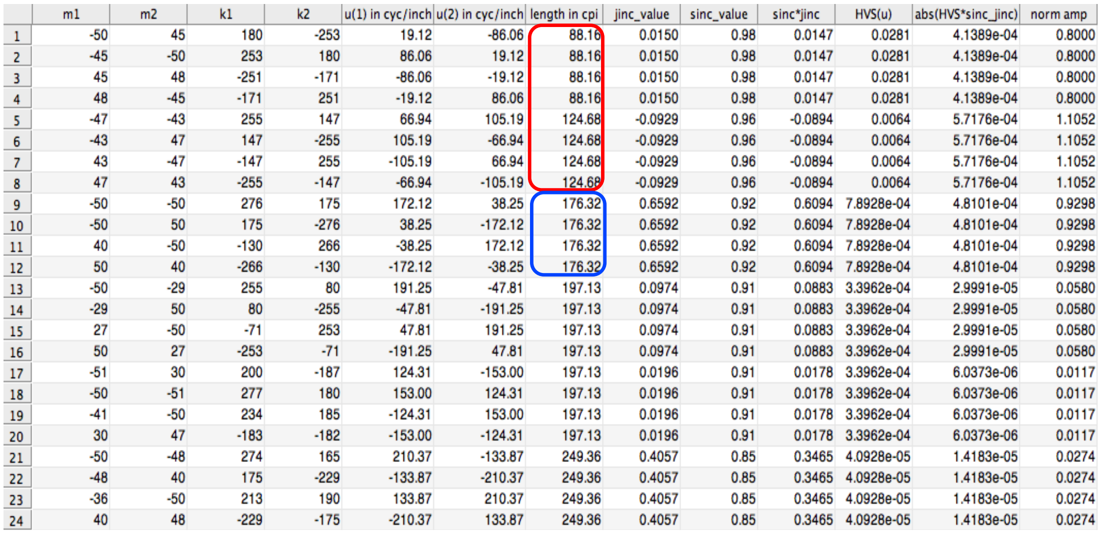
	m1	m2	k1	k2	u(1) in cyc/inch	u(2) in cyc/inch	length in cpi	jinc_value	HVS(u)	abs(HVS*jinc_value)
1	0	0	-1	0	-172.12	-38.25	176.32	0.6555	7.8928e-04	5.1735e-04
2	0	0	0	-1	38.25	-172.12	176.32	0.6555	7.8928e-04	5.1735e-04
3	0	0	0	1	-38.25	172.12	176.32	0.6555	7.8928e-04	5.1735e-04
4	0	0	1	0	172.12	38.25	176.32	0.6555	7.8928e-04	5.1735e-04
5	0	0	-1	-1	-133.87	-210.37	249.36	0.3953	4.0928e-05	1.6179e-05
6	0	0	-1	1	-210.37	133.87	249.36	0.3953	4.0928e-05	1.6179e-05
7	0	0	1	-1	210.37	-133.87	249.36	0.3953	4.0928e-05	1.6179e-05
8	0	0	1	1	133.87	210.37	249.36	0.3953	4.0928e-05	1.6179e-05
9	0	0	-2	0	-344.24	-76.50	352.64	0.0669	6.2296e-07	4.1664e-08
10	0	0	0	-2	76.50	-344.24	352.64	0.0669	6.2296e-07	4.1664e-08
11	0	0	0	2	-76.50	344.24	352.64	0.0669	6.2296e-07	4.1664e-08
12	0	0	2	0	344.24	76.50	352.64	0.0669	6.2296e-07	4.1664e-08
13	0	0	-2	-1	-306.00	-248.62	394.27	-0.0262	1.1534e-07	-3.0256e-09
14	0	0	-2	1	-382.49	95.62	394.27	-0.0262	1.1534e-07	-3.0256e-09
15	0	0	-1	-2	-95.62	-382.49	394.27	-0.0262	1.1534e-07	-3.0256e-09
16	0	0	-1	2	-248.62	306.00	394.27	-0.0262	1.1534e-07	-3.0256e-09
17	0	0	1	-2	248.62	-306.00	394.27	-0.0262	1.1534e-07	-3.0256e-09
18	0	0	1	2	95.62	382.49	394.27	-0.0262	1.1534e-07	-3.0256e-09
19	0	0	2	-1	382.49	-95.62	394.27	-0.0262	1.1534e-07	-3.0256e-09
20	0	0	2	1	306.00	248.62	394.27	-0.0262	1.1534e-07	-3.0256e-09
21	0	0	-2	-2	-267.75	-420.74	498.71	-0.1313	1.6751e-09	-2.1989e-10
22	0	0	-2	2	-420.74	267.75	498.71	-0.1313	1.6751e-09	-2.1989e-10
23	0	0	2	-2	420.74	-267.75	498.71	-0.1313	1.6751e-09	-2.1989e-10
24	0	0	2	2	267.75	420.74	498.71	-0.1313	1.6751e-09	-2.1989e-10

Fig. 2.8.: The summary statistics of the locations of the frequency components and their amplitudes for the analog rendering of an irregular screen with a periodicity matrix $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$. The table is generated using the following equation: $\tilde{G}^{\text{norm}}(\mathbf{u}) = H_{\text{HVS}}(\mathbf{u})\,\text{jinc}\left(\mathbf{M}^{-T}\mathbf{u}\right)\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\frac{1}{R}\mathbf{N}^{-\mathbf{T}}\mathbf{k}\right)$. The entire table has 24 entries, here $-2 \leq k_1 \leq 2, k_1 \in \mathbb{Z}$ and $-2 \leq k_2 \leq 2, k_2 \in \mathbb{Z}$. Note: the entries in the table are sorted based on the distance of the frequency components to the origin. Column 7 contains the distance in units of cycles per inch. The jinc_value in column 8 corresponds to the result of jinc $\left(\mathbf{M}^{-T}\mathbf{u}\right)$. The four main peaks are placed in a blue box.



	m1	m2	k1	k2	u(1) in cyc/inch	u(2) in cyc/inch	length in cpi	jinc_value	sinc_value	sinc*jinc	HVS(u)	abs(HVS*sinc_jinc)
1	-50	45	180	-253	19.12	-86.06	88.16	2.6983e-05	0.98	2.6464e-05	0.0281	7.4348e-07
2	-48	45	171	-251	19.12	-86.06	88.16	-4.0365e-05	0.98	-3.9589e-05	0.0281	1.1122e-06
3	-45	-50	253	180	86.06	19.12	88.16	2.6983e-05	0.98	2.6464e-05	0.0281	7.4348e-07
4	-45	-48	251	171	86.06	19.12	88.16	-4.0365e-05	0.98	-3.9589e-05	0.0281	1.1122e-06
5	45	48	-251	-171	-86.06	-19.12	88.16	-4.0365e-05	0.98	-3.9589e-05	0.0281	1.1122e-06
6	45	50	-253	-180	-86.06	-19.12	88.16	2.6983e-05	0.98	2.6464e-05	0.0281	7.4348e-07
7	48	-45	-171	251	-19.12	86.06	88.16	-4.0365e-05	0.98	-3.9589e-05	0.0281	1.1122e-06
8	50	-45	-180	253	-19.12	86.06	88.16	2.6983e-05	0.98	2.6464e-05	0.0281	7.4348e-07
9	-43	-18	212	38	86.06	19.12	88.16	-1.5813e-04	0.98	-1.5509e-04	0.0281	4.3570e-06
10	-18	43	38	-212	19.12	-86.06	88.16	-1.5813e-04	0.98	-1.5509e-04	0.0281	4.3570e-06
11	18	-43	-38	212	-19.12	86.06	88.16	-1.5813e-04	0.98	-1.5509e-04	0.0281	4.3570e-06
12	43	18	-212	-38	-86.06	-19.12	88.16	-1.5813e-04	0.98	-1.5509e-04	0.0281	4.3570e-06
13	-43	-10	204	2	86.06	19.12	88.16	1.0501e-04	0.98	1.0299e-04	0.0281	2.8935e-06
14	-43	-8	202	-7	86.06	19.12	88.16	-1.4712e-04	0.98	-1.4429e-04	0.0281	4.0537e-06
15	-10	43	2	-204	19.12	-86.06	88.16	1.0501e-04	0.98	1.0299e-04	0.0281	2.8935e-06
16	-8	43	-7	-202	19.12	-86.06	88.16	-1.4712e-04	0.98	-1.4429e-04	0.0281	4.0537e-06
17	8	-43	7	202	-19.12	86.06	88.16	-1.4712e-04	0.98	-1.4429e-04	0.0281	4.0537e-06
18	10	-43	-2	204	-19.12	86.06	88.16	1.0501e-04	0.98	1.0299e-04	0.0281	2.8935e-06
19	43	8	-202	7	-86.06	-19.12	88.16	-1.4712e-04	0.98	-1.4429e-04	0.0281	4.0537e-06
20	43	10	-204	-2	-86.06	-19.12	88.16	1.0501e-04	0.98	1.0299e-04	0.0281	2.8935e-06
21	-43	-16	210	29	86.06	19.12	88.16	-1.9660e-04	0.98	-1.9282e-04	0.0281	5.4172e-06
22	-43	-14	208	20	86.06	19.12	88.16	-2.0167e-04	0.98	-1.9779e-04	0.0281	5.5567e-06
23	-43	-12	206	11	86.06	19.12	88.16	9.6501e-05	0.98	9.4645e-05	0.0281	2.6590e-06
24	-43	-6	200	-16	86.06	19.12	88.16	3.8140e-05	0.98	3.7407e-05	0.0281	1.0509e-06

Fig. 2.9.: The summary statistics of the locations of the frequency components and their amplitudes for the digital rendering of an irregular screen with a periodicity matrix $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$. The table is generated using the following equation: $\tilde{G}^{\text{norm}}_{\text{printer}}(\mathbf{u}) = H_{\text{HVS}}(\mathbf{u})\,\text{sinc}(R\mathbf{u}) \sum_{\mathbf{m}\in Z^2} \text{jinc}\left(\mathbf{M}^{-T}(\mathbf{u}-\mathbf{Vm})\right) \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right)$. The entire table has 211,786 entries, here $-51 \leq m_1 \leq 51, m_1 \in \mathbb{Z}$ and $-51 \leq m_2 \leq 51, m_2 \in \mathbb{Z}$, $\mathbf{k}$ limits are unique for each pair of $\mathbf{m}$ values. Since different $\mathbf{m}$ and $\mathbf{k}$ give the identical u_2 and u_2 (or $u(1)$ and $u(2)$ as labeled in the table), their jinc values needed to be combined. The example of the frequency components that are combined are placed in green boxes. Their jinc components are combined: $2.6983e-5+(-4.0365e-5)+(-1.5813e-4)+\ldots = 0.0150$. The changes are reflected in the next figure. The blue arrow shows that the table is sorted in the ascending order, based on the distance of the frequency components to the origin. Note: the jinc_value in column 8 corresponds to the result of jinc $\left(\mathbf{M}^{-T}(\mathbf{u}-\mathbf{Vm})\right)$, whereas the sinc_value corresponds to the result of sinc $(R\mathbf{u})$, and column 10 corresponds to the product of columns 8 and 9.



	m1	m2	k1	k2	u(1) in cyc/inch	u(2) in cyc/inch	length in cpi	jinc_value	sinc_value	sinc*jinc	HVS(u)	abs(HVS*sinc_jinc)	norm amp
1	-50	45	180	-253	19.12	-86.06	88.16	0.0150	0.98	0.0147	0.0281	4.1389e-04	0.8000
2	-45	-50	253	180	86.06	19.12	88.16	0.0150	0.98	0.0147	0.0281	4.1389e-04	0.8000
3	45	48	-251	-171	-86.06	-19.12	88.16	0.0150	0.98	0.0147	0.0281	4.1389e-04	0.8000
4	48	-45	-171	251	-19.12	86.06	88.16	0.0150	0.98	0.0147	0.0281	4.1389e-04	0.8000
5	-47	-43	255	147	66.94	105.19	124.68	-0.0929	0.96	-0.0894	0.0064	5.7176e-04	1.1052
6	-43	47	147	-255	105.19	-66.94	124.68	-0.0929	0.96	-0.0894	0.0064	5.7176e-04	1.1052
7	43	-47	-147	255	-105.19	66.94	124.68	-0.0929	0.96	-0.0894	0.0064	5.7176e-04	1.1052
8	47	43	-255	-147	-66.94	-105.19	124.68	-0.0929	0.96	-0.0894	0.0064	5.7176e-04	1.1052
9	-50	-50	276	175	172.12	38.25	176.32	0.6592	0.92	0.6094	7.8928e-04	4.8101e-04	0.9298
10	-50	50	175	-276	38.25	-172.12	176.32	0.6592	0.92	0.6094	7.8928e-04	4.8101e-04	0.9298
11	40	-50	-130	266	-38.25	172.12	176.32	0.6592	0.92	0.6094	7.8928e-04	4.8101e-04	0.9298
12	50	40	-266	-130	-172.12	-38.25	176.32	0.6592	0.92	0.6094	7.8928e-04	4.8101e-04	0.9298
13	-50	-29	255	80	191.25	-47.81	197.13	0.0974	0.91	0.0883	3.3962e-04	2.9991e-05	0.0580
14	-29	50	80	-255	-47.81	-191.25	197.13	0.0974	0.91	0.0883	3.3962e-04	2.9991e-05	0.0580
15	27	-50	-71	253	47.81	191.25	197.13	0.0974	0.91	0.0883	3.3962e-04	2.9991e-05	0.0580
16	50	27	-253	-71	-191.25	47.81	197.13	0.0974	0.91	0.0883	3.3962e-04	2.9991e-05	0.0580
17	-51	30	200	-187	124.31	-153.00	197.13	0.0196	0.91	0.0178	3.3962e-04	6.0373e-06	0.0117
18	-50	-51	277	180	153.00	124.31	197.13	0.0196	0.91	0.0178	3.3962e-04	6.0373e-06	0.0117
19	-41	-50	234	185	-124.31	153.00	197.13	0.0196	0.91	0.0178	3.3962e-04	6.0373e-06	0.0117
20	30	47	-183	-182	-153.00	-124.31	197.13	0.0196	0.91	0.0178	3.3962e-04	6.0373e-06	0.0117
21	-50	-48	274	165	210.37	-133.87	249.36	0.4057	0.85	0.3465	4.0928e-05	1.4183e-05	0.0274
22	-48	40	175	-229	-133.87	-210.37	249.36	0.4057	0.85	0.3465	4.0928e-05	1.4183e-05	0.0274
23	-36	-50	213	190	133.87	210.37	249.36	0.4057	0.85	0.3465	4.0928e-05	1.4183e-05	0.0274
24	40	48	-229	-175	-210.37	133.87	249.36	0.4057	0.85	0.3465	4.0928e-05	1.4183e-05	0.0274

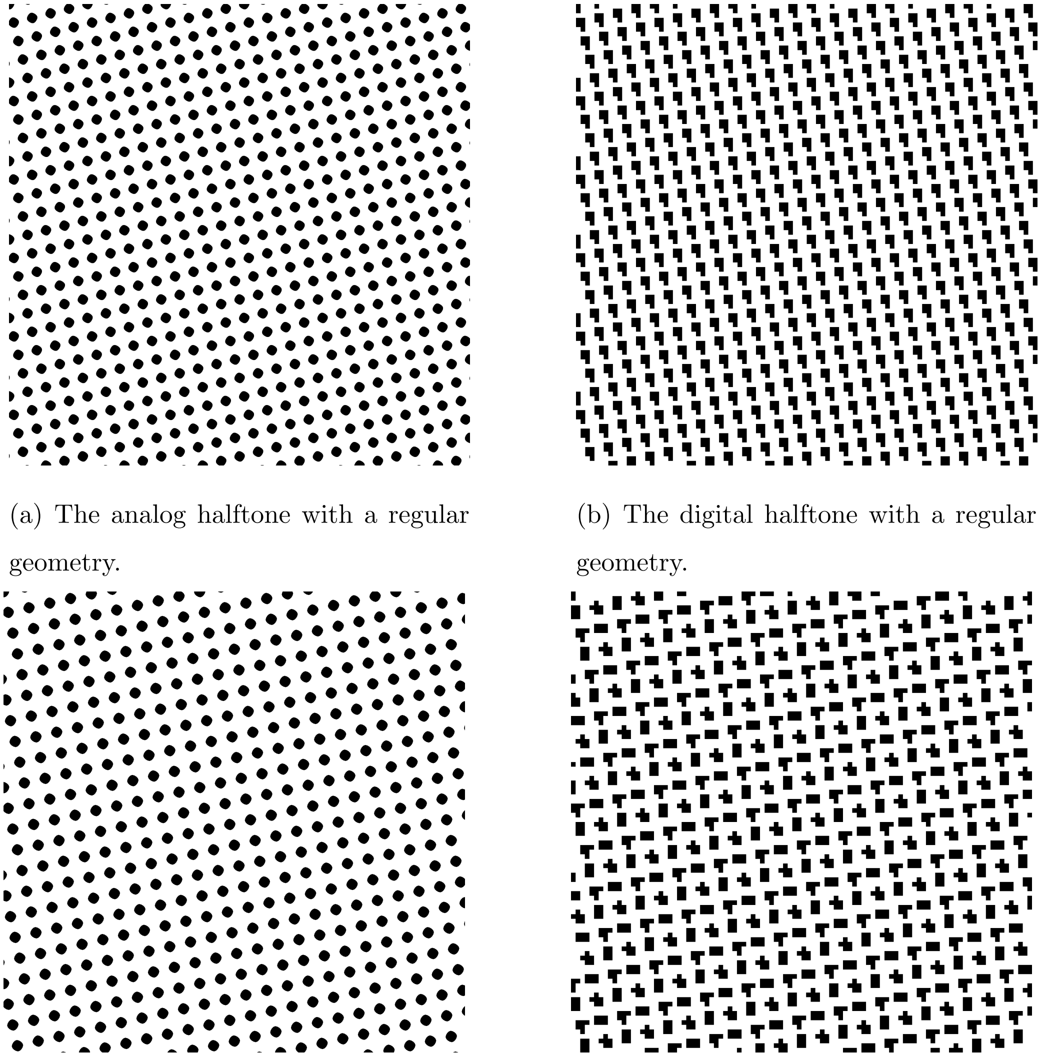
Fig. 2.10.: The summary statistics of the locations of the frequency components and their amplitudes for the digital rendering of an irregular screen with a periodicity matrix $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$ updated. The table is generated using the following equation: $\tilde{G}^{\text{norm}}_{\text{printer}}(\mathbf{u}) = H_{\text{HVS}}(\mathbf{u})\,\text{sinc}\,(R\mathbf{u}) \sum_{\mathbf{m}\in Z^2} \text{jinc}\left(\mathbf{M}^{-T}(\mathbf{u}-\mathbf{V}\mathbf{m})\right) \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u}-\mathbf{V}\mathbf{m}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right)$. This is the updated table shown in Fig. 2.9. The entire table has 80 entries. Column 13 contains the values which are obtained by scaling column 12 by the maximum amplitude obtained from the four main peaks in the analog rendering (percentage). The frequency components in the blue box are the example of the components that have the same locations as those that were present in the analog rendering spectrum, whereas the frequency components in the red box are the example of the new frequency components that were not present in the analog rendering.

integers, then the screen is irregular. The examples of regular and irregular analog and digital halftone textures are provided in Fig. 2.11.

Their corresponding DFT spectra are shown in Fig. 2.12. As can be seen from Fig. 2.12, with the irregular halftones, not only can we see the frequency components that fall on the same locations as in the analog rendering, but we can also observe the additional frequency components that fall on the different locations as shown in Fig. 2.12 d) [67]. The frequency components near the origin will be most visible, and will contribute the most to the perception of noise in the resulting image. Based on this observation, we chose to compute four ratios. The numerator terms for all four ratios are based on the square-root of the sum of the squares of the amplitudes of the selected terms. Ratio 1 summarizes the contributions of these components, which lie in a quadrilateral bounded by, but not including the four lowest frequencies in the analog halftone. Ratio 2 summarizes the contribution of the digital halftone at the same frequencies (fundamentals and harmonics) that were present in the analog halftone. Ratio 3 summarizes the contributions of all components in the digital halftone, except those present in the analog halftone. Finally, Ratio 4 summarizes the contributions of all non-zero frequency terms in the printed digital halftone. We normalize all our ratios by the square-root of the sum of the squares of the amplitudes of the four lowest (fundamental) frequencies in the analog halftone image [65]. An example of how the frequency components are selected for computing ratios is demonstrated in Fig. 2.13.

2.5.2 Experimental results

In order to compare our results we pick the 3 screen tile vectors from Fig. 2.14 and compare their ratios. The target screen parameters for these geometries are the screen frequency of 180 lpi, screen angle of 15°, and printer resolution of 812.8 dpi. Fig. 2.15 contains the values of the four ratios for the three periodicity matrices and halftone patterns. As expected, for the regular screen, Ratios 1 and 3 are zero; and Ratios 2 and 4 are nearly unity. The value of Ratio 2 is close to unity for



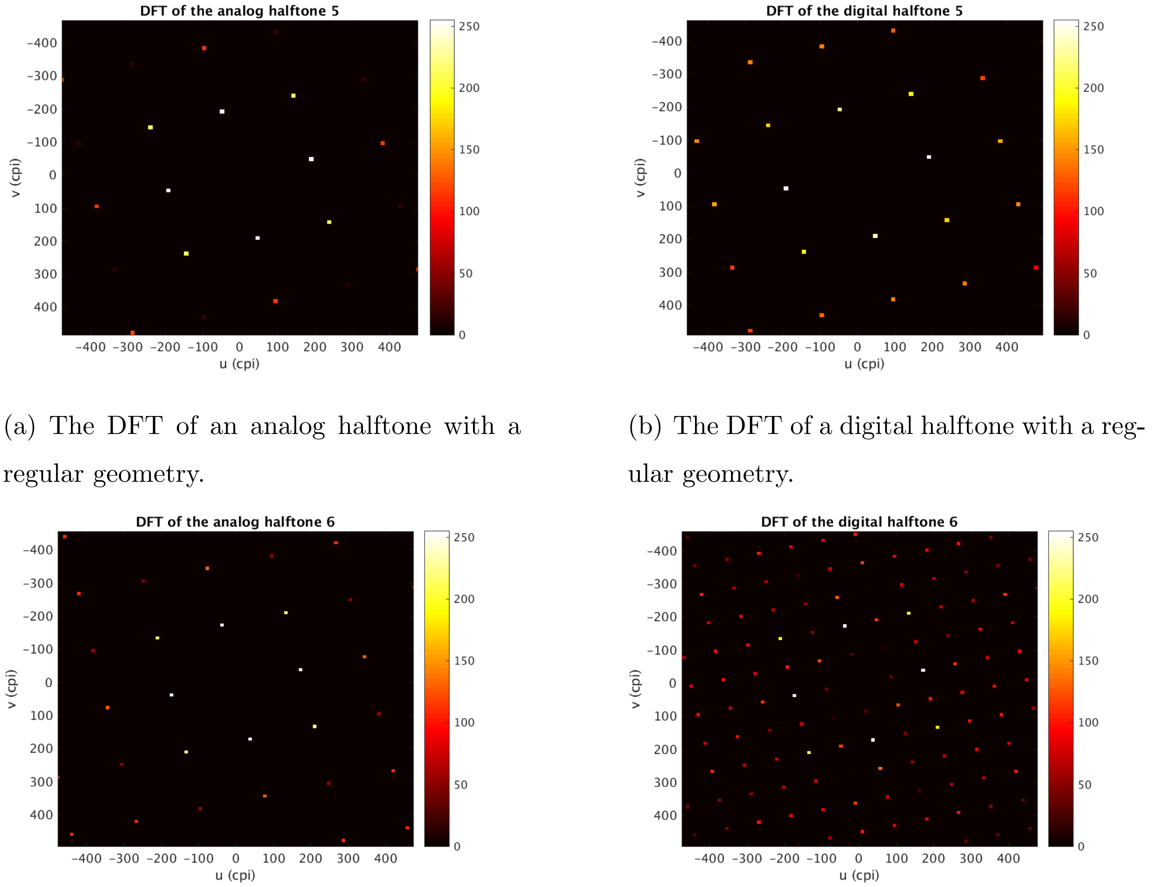
(a) The analog halftone with a regular geometry.

(b) The digital halftone with a regular geometry.

(c) The analog halftone with an irregular geometry.

(d) The digital halftone with an irregular geometry.

Fig. 2.11.: Comparison between analog and digital, regular and irregular halftones: a) and b) are regular halftones that have a periodicity matrix $\mathbf{N} = \begin{bmatrix} 4 & -1 \\ 1 & 4 \end{bmatrix}$; c) and d) are irregular halftones that have a periodicity matrix $\mathbf{N} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$



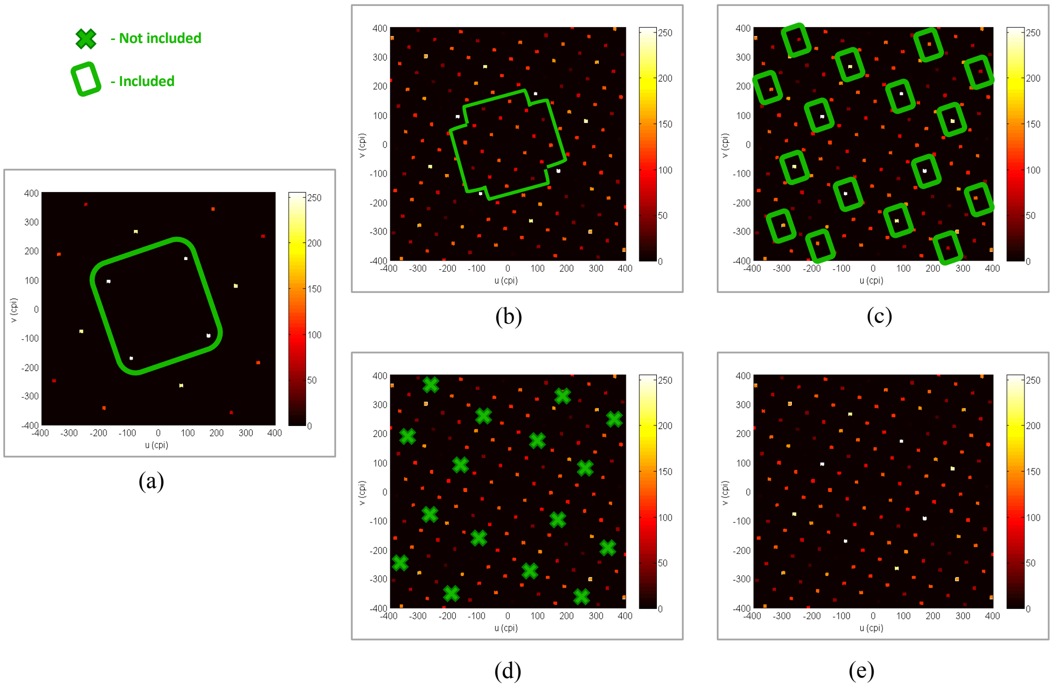
(a) The DFT of an analog halftone with a regular geometry.

(b) The DFT of a digital halftone with a regular geometry.

(c) The DFT of an analog halftone with an irregular geometry.

(d) The DFT of a digital halftone with an irregular geometry.

Fig. 2.12.: Comparison between the DFT of analog and digital, regular and irregular halftones: a) and b) are the DFT spectra of a regular geometry with periodicity matrix $\mathbf{N} = \begin{bmatrix} 4 & -1 \\ 1 & 4 \end{bmatrix}$; c) and d) are the DFT spectra of an irregular geometry with periodicity matrix $\mathbf{N} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$



(a)

(b)

(c)

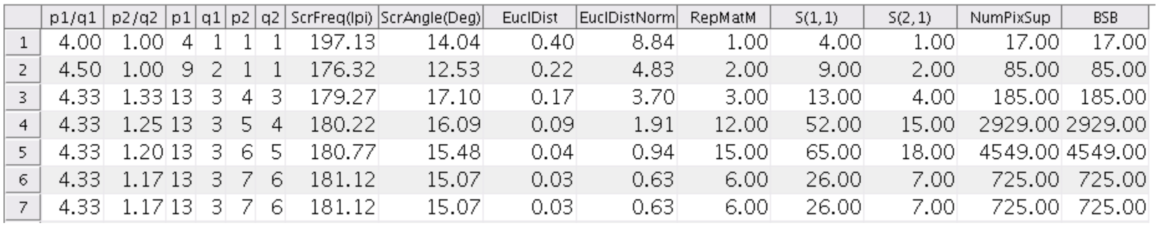
(d)

(e)

Fig. 2.13.: This figure demonstrates which frequency components are considered in order to compute four ratios. Subfigure a) contains an example of the analog rendering spectrum, subfigures (b), (c), (d), and (e) contain the corresponding digital rendering spectrum. If the frequency components are surrounded with green lines, that means that they are the only ones that are used for computing the corresponding ratio. If the frequency components are marked with a green x mark, then they are the frequency components that are not included in computing the corresponding ratio. a) The four fundamental frequency components (peaks) in the analog rendering spectrum are selected. These frequency components are used in computing the denominator for the four ratios. b) The frequency components in the digital rendering whose distance to the origin is smaller than the distance to the fundamental peak in the analog rendering. These frequency components are used in computing the numerator for Ratio 1. c) The frequency components in the digital rendering, which fall on the same locations as those that are in the analog rendering spectrum. These frequency components are used in computing the numerator for Ratio 2. d) All the new frequency components that appear as a result of digitization in the digital rendering, and that are not present in the analog rendering. These frequency components are used in computing the numerator for Ratio 3. e) The entire digital rendering spectrum. These frequency components are used in computing the numerator for Ratio 4.

all three screens, suggesting that the significance of the spectral power change at harmonics of the fundamental frequency of the analog halftone induced by an irregular screen is minimal. In addition, for the two irregular screens, Ratios 1, 3, and 4 have nearly identical values. This suggests that the spectral power within the quadrilateral bounded by, but not including the four lowest frequencies in the analog halftone, will dominate all three of these ratios. The value of these ratios is much larger for $\mathbf{N}_3$ than for $\mathbf{N}_2$. This is in agreement with the visual appearance of the corresponding halftone patterns shown in Fig. 2.15, which is much noisier for $\mathbf{N}_3$ than for $\mathbf{N}_2$.

In addition, Table 2.3 contains ratio results for all candidate tile vectors presented in Fig. 2.14. It can be seen from the results that as we increase the denominator threshold value and allow higher values for digitization, the ratios 1, 3, and 4 increase drastically, which happens due to the fact that as we increase the denominator, the number of additional frequency components that occur in the digital rendering also goes up. Therefore, the number of additional frequency components that are closest to the origin also increases, which leads to higher graininess levels. However, it may also happen that if we increase the denominator only slightly, the ratio values may both increase or decrease, which can be tracked from the data tables that we generate. As for ratio 2, its value is close to 1 most of the time due to the fact there are always frequency components in the digital rendering that fall on the same locations in the analog rendering.

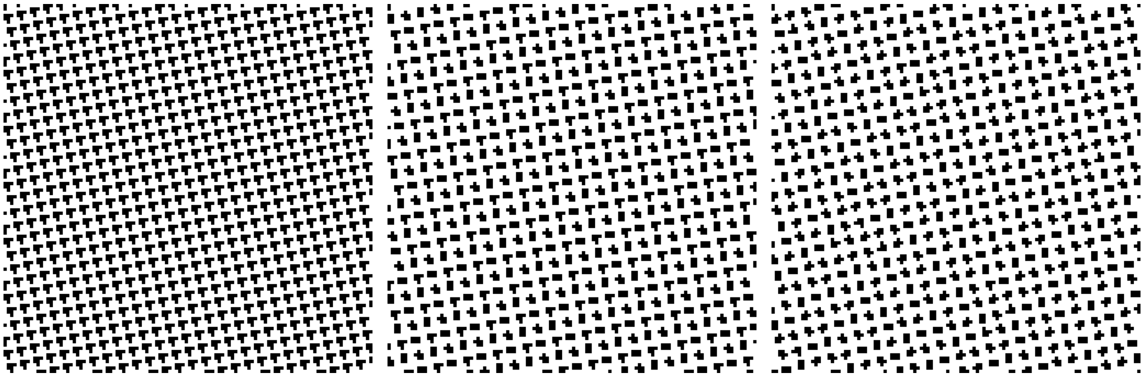


	p1/q1	p2/q2	p1	q1	p2	q2	ScrFreq(lpi)	ScrAngle(Deg)	EuclDist	EuclDistNorm	RepMatM	S(1,1)	S(2,1)	NumPixSup	BSB
1	4.00	1.00	4	1	1	1	197.13	14.04	0.40	8.84	1.00	4.00	1.00	17.00	17.00
2	4.50	1.00	9	2	1	1	176.32	12.53	0.22	4.83	2.00	9.00	2.00	85.00	85.00
3	4.33	1.33	13	3	4	3	179.27	17.10	0.17	3.70	3.00	13.00	4.00	185.00	185.00
4	4.33	1.25	13	3	5	4	180.22	16.09	0.09	1.91	12.00	52.00	15.00	2929.00	2929.00
5	4.33	1.20	13	3	6	5	180.77	15.48	0.04	0.94	15.00	65.00	18.00	4549.00	4549.00
6	4.33	1.17	13	3	7	6	181.12	15.07	0.03	0.63	6.00	26.00	7.00	725.00	725.00
7	4.33	1.17	13	3	7	6	181.12	15.07	0.03	0.63	6.00	26.00	7.00	725.00	725.00

Fig. 2.14.: The set of the candidate screen tile-vectors for 180 lpi, 15° and 812.8 dpi. The target tile vector is $\mathbf{v}_{\mathbf{1}}^{\mathbf{target}} = \begin{bmatrix} 4.36 & 1.17 \end{bmatrix}$.

Table 2.3.: Ratio results for the candidate tile vectors presented in Fig. 2.14. As we increase the denominator value (going from row 1 to row 6), and allow higher values for digitization, the ratios 1, 3, and 4 increase drastically, which happens due to the fact that as we increase the denominator, the number of additional frequency components that occur in the digital rendering also goes up. As for ratio 2, its value is close to 1 most of the time due to the fact there are always frequency components in the digital rendering that fall on the same locations in the analog rendering.

Periodicity matrix	**Ratio 1**	**Ratio 2**	**Ratio 3**	**Ratio 4**
$\mathbf{N}_1 = \begin{bmatrix} 4 & -1 \\ 1 & 4 \end{bmatrix}$	0	0.99	0	0.99
$\mathbf{N}_2 = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$	1.36	0.93	1.36	1.65
$\mathbf{N}_3 = \begin{bmatrix} 13/3 & -4/3 \\ 4/3 & 13/3 \end{bmatrix}$	10.05	0.91	10.05	10.09
$\mathbf{N}_4 = \begin{bmatrix} 13/3 & -5/4 \\ 5/4 & 13/3 \end{bmatrix}$	17.77	0.92	17.77	17.79
$\mathbf{N}_5 = \begin{bmatrix} 13/3 & -6/5 \\ 6/5 & 13/3 \end{bmatrix}$	21.31	0.91	21.31	21.33
$\mathbf{N}_6 = \begin{bmatrix} 13/3 & -7/6 \\ 7/6 & 13/3 \end{bmatrix}$	20.22	0.91	20.22	20.24



(a)

$$\mathbf{N}_1 = \begin{bmatrix} 4 & -1 \\ 1 & 4 \end{bmatrix}$$

(b)

$$\mathbf{N}_2 = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$$

(c)

$$\mathbf{N}_3 = \begin{bmatrix} \frac{13}{3} & \frac{-7}{6} \\ \frac{7}{6} & \frac{13}{3} \end{bmatrix}$$

Fig. 2.15.: Comparison of the ratio results: the three halftones (a), (b), and (c) correspond to the same target tile vector with a screen frequency of 180 lpi, screen angle of 15°, and printer resolution of 812.8 dpi. The halftone in (a) is a regular halftone, whereas the halftones in (b) and (c) are irregular halftones. a) Ratio 1 = 0, Ratio 2 = 0.99, Ratio 3 = 0, Ratio 4 = 0.99. (b) Ratio 1 = 1.36, Ratio 2 = 0.93, Ratio 3 = 1.36, Ratio 4 = 1.65. (c) Ratio 1 = 21.31, Ratio 2 = 0.91, Ratio 3 = 21.31, Ratio 4 = 21.33

2.6 Conclusion

In this chapter, we proposed a procedure for finding the optimal screens that will minimize image graininess. Our algorithm was based on developing a set of alternative tile-vectors that will best fit the specified target screen set: screen frequency, screen angle, and printer resolution. We also, generated the tool for looking at the frequency spectrum of any halftone, and finding all the information about its frequency components. Finally, we provided a measure for computing the image fluctuation for a single separation based only the periodicity matrix. Our ratio metrics also assist us in making better choices when selecting between several halftones without generating the halftones themselves.

3. DIGITAL SIMULATION-BASED ANALYSIS OF HALFTONING WITH AN ARBITRARY NUMBER OF SEPARATIONS (REGULAR OR IRREGULAR) USING THE SPATIOCHROMATIC HVS MODEL

3.1 Introduction

The purpose of our research is to design a predictor that will help us assign colorants to different screens that will minimize the perceived error. In addition, we would like to find out what are the advantages and disadvantages in using regular and irregular halftones for the superposition. In our previous work, which was based on a single separation, we made a conclusion that choosing an irregular halftone screen has an advantage of getting closer to the target screen tile vector. However, it results in higher graininess levels. Whereas with a regular screen, the distance from the target screen tile vector is increased, but the graininess levels may improve [65]. Another challenge that we are facing is that as we explore the superposition of two irregular color halftones, we need not only take into account the effects of superposition but also the effects of digitization. In order to eliminate dot-on-dot printing in the superposition image, the screen of each colorant is rotated to a different angle. However, this causes other problems such as moire and rosette artifacts, which are the result of the interaction between two lattices. The moire phenomenon refers to a low-frequency structure, which has a very visible pattern and can be observed at the superposition of two halftones, and not in the individual halftones separately [48]. In contrast, a rosette pattern has a circular or polygonal pattern, which is formed as a result of rotating halftone screens [48]. The superposition of two color halftones is a very important question to study because the artifacts such as moire, rosette and misregistration can lead to color shifts. Baqai and Allebach presented a systematic method for designing moire- and rosette- free clustered-dot color screens for discrete raster color systems [48]. Whereas Ufuk and Allebach developed a model-based color halftoning method which minimizes the perceived error between the continuous tone original color image and the color halftone image [57]. In our research, we integrate the HVS based model that takes into account the spectral representation of color and the difference in the responses of the human viewer to luminance and chrominance information in order to obtain the perceived error metric. Specifically, we are interested in investigating the new challenges that occur as a result of superimposing the two irregular halftone screens in comparison to the two regular halftone screens.

3.2 HVS-Based Model for Superposition of Two Color Halftones

3.2.1 Method for obtaining the average squared perceived error

The procedure implemented in our research consists of three parts. The first part involves generating the three $g_X[\mathbf{m}]$, $g_Y[\mathbf{m}]$, and $g_Z[\mathbf{m}]$ images of the superposition halftone. In the second part, in order to account for the difference in the responses of the human viewer to luminance and chrominance information, we use a linearized version of the $L^{\star}a^{\star}b^{\star}$ uniform color space, which is $Y_yC_zC_z$ [70]. And lastly, in part three we apply separate luminance and chrominance frequency responses; and we obtain the average squared perceived error.

Color device model

We start by considering the superposition of two colorants. The halftone patterns of the cyan and magenta colorants associated with their periodicity matrices $\mathbf{N}_c$ and $\mathbf{N}_m$ are represented as $g_c[\mathbf{m}]$ and $g_m[\mathbf{m}]$ in Fig. 3.1. The superposition of these two halftones generates three colorant areas (Neugebauer primaries) c, m, cm, and a white w area. In order to transform to a device-independent space CIE XYZ, we use the Neugebauer primaries generator to obtain a $g_{NP}[\mathbf{m}]$ image. Next, we obtain the

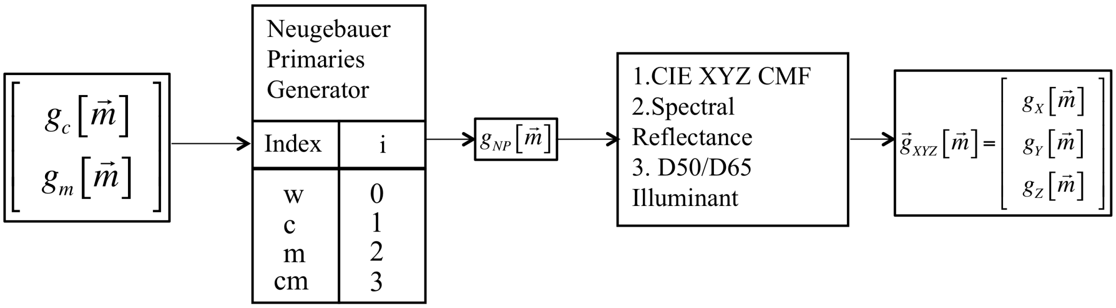

$g_c[\vec{m}]$
$g_m[\vec{m}]$
Neugebauer Primaries Generator
Index
i
w
c
m
cm
0
1
2
3
$g_{NP}[\vec{m}]$
1.CIE XYZ CMF
2.Spectral Reflectance
3. D50/D65 Illuminant
$\vec{g}_{XYZ}[\vec{m}] = \begin{bmatrix} g_X[\vec{m}] \\ g_Y[\vec{m}] \\ g_Z[\vec{m}] \end{bmatrix}$

Fig. 3.1.: The procedure for generating images in CIE XYZ.

XYZ matrix using the spectral reflectance, the CIE XYZ color matching functions (CMF's), and a D65 Illuminant, depicted in Figs. 3.2, 3.3, and 3.4. X, Y, and Z values are obtained using the following equations:

$$\begin{aligned} X &= \int_{400}^{700} S(\lambda)\bar{x}(\lambda)\mathrm{d}\lambda \\ Y &= \int_{400}^{700} S(\lambda)\bar{x}(\lambda)\mathrm{d}\lambda \\ Z &= \int_{400}^{700} S(\lambda)\bar{x}(\lambda)\mathrm{d}\lambda, \end{aligned} \tag{3.1}$$

where

$$S(\lambda) = I(\lambda)R(\lambda). \tag{3.2}$$

Here, we define $I(\lambda)$ to be the Illuminant function, for example D50 Illuminant or D65 Illuminant, and $R(\lambda)$ to be the Reflectance function corresponding to the specific Neugebauer primary, including the media on which the colorant is placed. It is unique to the target printer. In this chapter, we use the Ideal Block Reflectance functions,

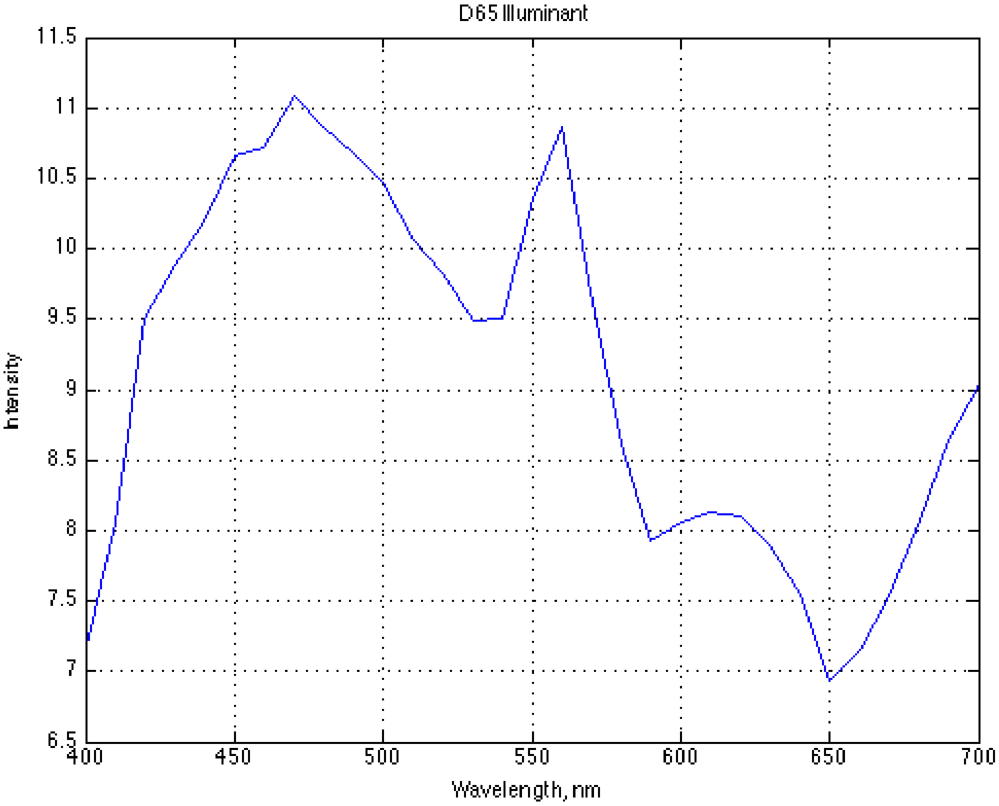


Fig. 3.2.: D65 Illuminant.

shown in Fig. 3.4 for the cyan and magenta colorants only, and the D65 Illuminant to illustrate our results. The final XYZ matrix has the following form:

$$XYZ = \begin{bmatrix} X_c & X_m & X_{cm} & X_w \\ Y_c & Y_m & Y_{cm} & Y_w \\ Z_c & Z_m & Z_{cm} & Z_w \end{bmatrix}. \tag{3.3}$$

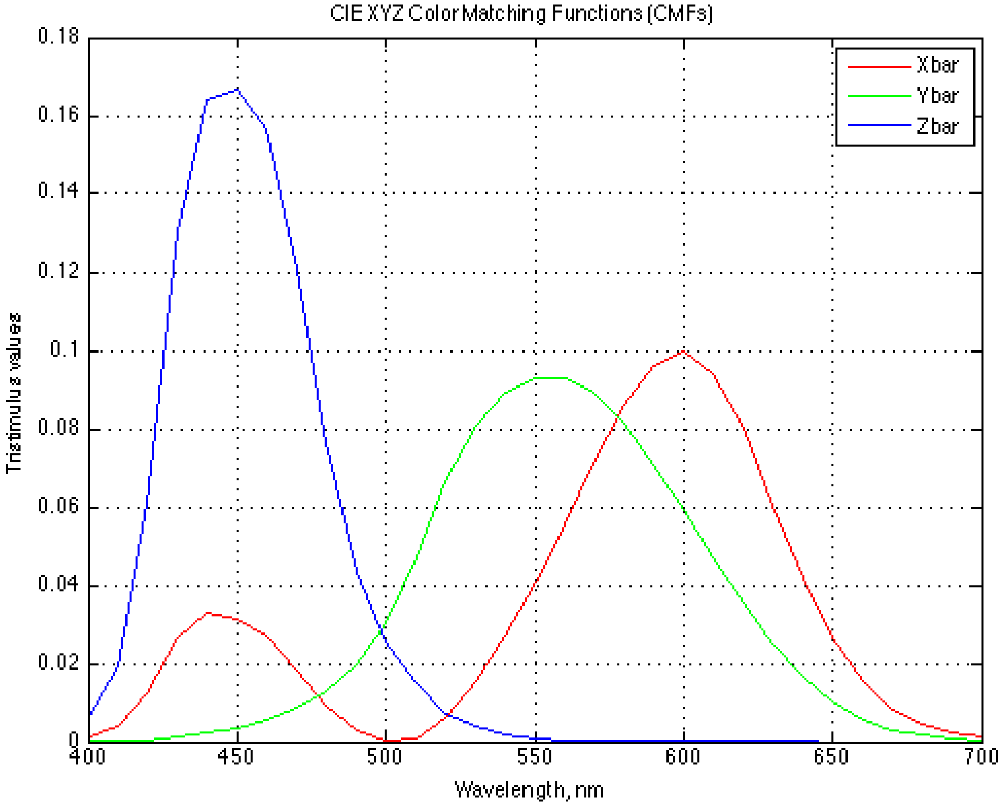


Fig. 3.3.: CIE **XYZ** color matching functions.

Lastly, to get the $g_X[\mathbf{m}]$, we use the previously obtained $g_{NP}[\mathbf{m}]$ and the Xvalues for c, m, cm, and w:

$$g_X[\mathbf{m}] = \begin{cases} X_w, & g_{NP}[\mathbf{m}] = 0 \\ X_c, & g_{NP}[\mathbf{m}] = 1 \\ X_m, & g_{NP}[\mathbf{m}] = 2 \\ X_{cm}, & g_{NP}[\mathbf{m}] = 3. \end{cases} \tag{3.4}$$

The $g_Y[\mathbf{m}]$ and $g_Z[\mathbf{m}]$ images are similarly obtained.

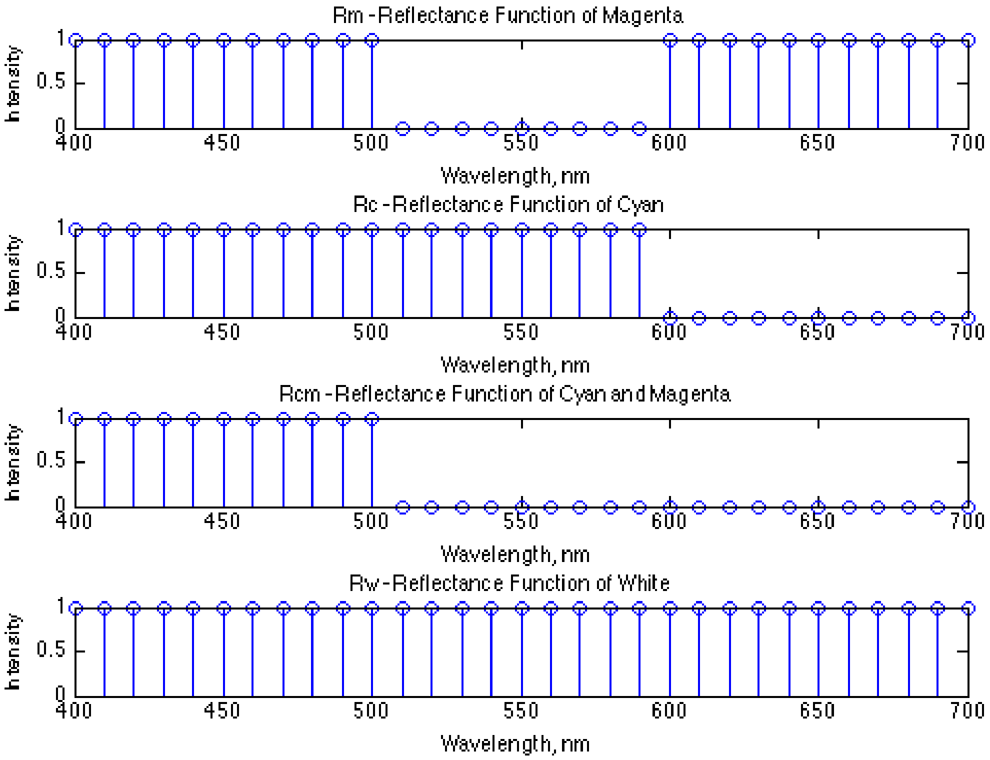


Fig. 3.4.: Ideal block reflectance for cyan and magenta colorants.

Opponent color space

Next, we implement the human visual model that accounts for the difference in the responses of the human viewer to luminance and chrominance information. We use a linearized version of $L^{\star}a^{\star}b^{\star}$ color space in order to preserve local averages, which are crucial for faithful tone reproduction [70]. The transformation that we use is:

$$\begin{aligned} Y_y &= 116\frac{Y}{Y_n} \\ C_x &= 500\left[\frac{X}{X_n}-\frac{Y}{Y_n}\right] \\ C_z &= 200\left[\frac{Y}{Y_n}-\frac{Z}{Z_n}\right], \end{aligned} \tag{3.5}$$

where (X_n, Y_n, Z_n) is the D65 white point for the XYZ color space. The Y_y component represents luminance, and C_x and C_z components represent the $R-G$ and $B-Y$ opponent-color chrominance components. In order to look at each channel separately, we convert each channel to sRGB by setting the other two channels to 0 or a constant value [70]. For example, in order to look at the Y channel, we set C_x, and C_z channels to 0 resulting in $(Y_y, 0, 0)$. After that we transform back to XYZ and to sRGB [70]. Figure 3.5 summarizes the procedure. It turned out after the transformation of $(Y_{mid}, C_x, 0)$ and $(Y_{mid}, 0, C_z)$ from XYZ to sRGB, there were negative values in sRGB at some pixels. In order to not lose any data, we developed a procedure, that will fit all the data inside the range $[0, 1]$. Figure 3.6 depicts the procedure for visualizing $(Y_{mid}, 0, C_z)$. A similar approach is used to display $(Y_{mid}, C_x, 0)$. In addition to Fig. 3.6, a detailed derivation for obtaining the α parameter is provided in the Appendix.

3.2.2 Spatial frequency response and error metric

The next step is to apply separate luminance and chrominance frequency responses to Y_y, C_x and C_z, which are chosen based on the spatial sensitivity of the human eye to them. For the luminance channel, we use Daly HVS, which is given as:

$$H(\bar{\rho}) = \begin{cases} a\left(b+c\bar{\rho}\right)\exp\left(-\left(c\bar{\rho}\right)^{d}\right), & \bar{\rho} > \bar{\rho}_{max} \\ 1, & \text{else,} \end{cases} \tag{3.6}$$

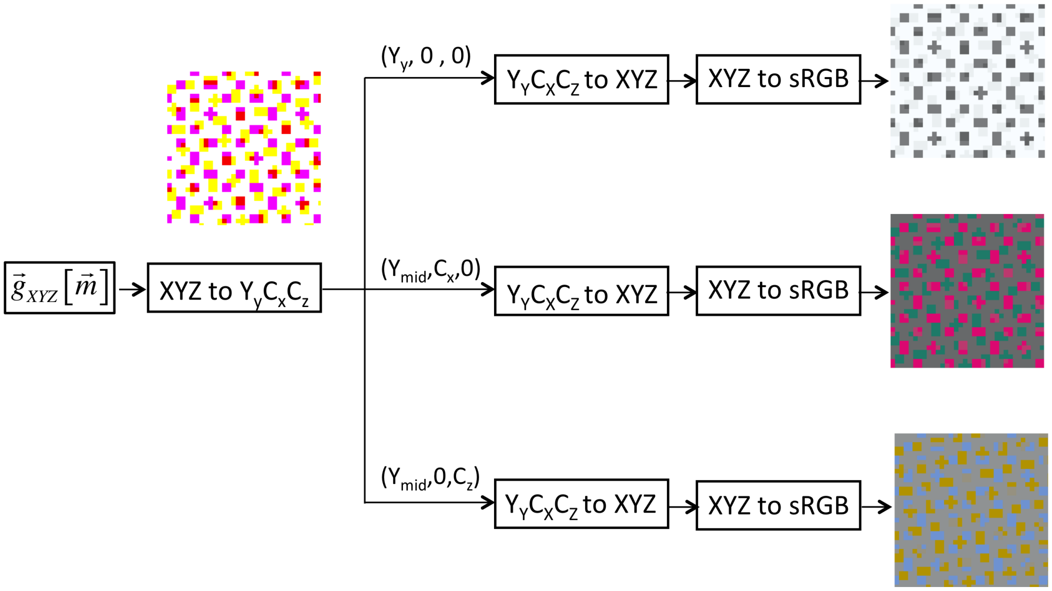


Fig. 3.5.: Visualization of Y_y, C_x and C_z channels in sRGB. Y_{mid} is the midpoint between the maximum and the minimum values in the Y_y component .

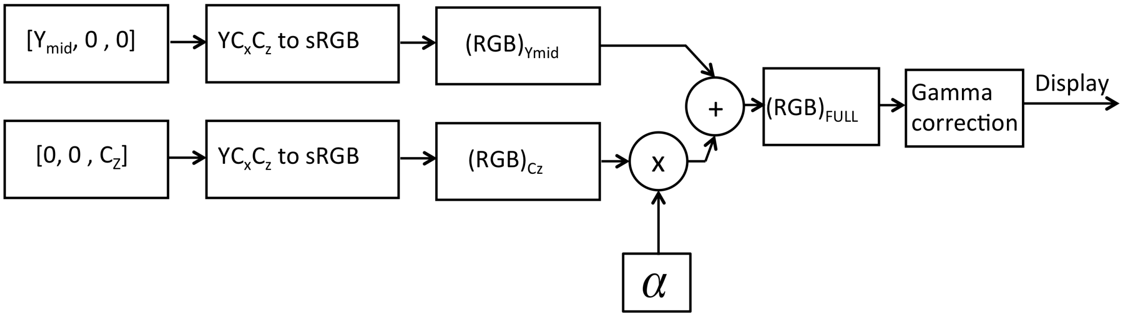


Fig. 3.6.: Procedure for visualizing $(Y_{mid}, 0, C_z)$.

where $a = 2.2, b = 0.192, c = 0.114, d = 1.1, \bar{\rho}_{max} = 6.6$, and $\bar{\rho} = \sqrt{u^2 + v^2}$ [69]. For the chrominance channel, we use the approximation by Kolpatzik and Bouman to experimental data collected by Mullen, which is given as:

$$W(\bar{\rho}) = A \exp(-\alpha||\bar{\rho}||), \tag{3.7}$$

where $\alpha = 0.419$, and $A = 100$ [70], [48]. The procedure is demonstrated in Fig. 3.7. After obtaining $\epsilon_{Y_y}(i,j)$, $\epsilon_{C_x}(i,j)$, $\epsilon_{C_z}(i,j)$ for each pixel of the image located at (i,j), and scaling $\epsilon_{Y_y}(i,j)$ by 4, we calculate $\Delta E(i,j)$ in the following way:

$$\Delta E(i,j) = \sqrt{\left(\epsilon_{Y_y}(i,j)^2 + \epsilon_{C_x}(i,j)^2 + \epsilon_{C_z}(i,j)^2\right)}. \tag{3.8}$$

Finally, we obtain the $\Delta E_{fluctuation}$ value, using:

$$\Delta E_{fluctuation} = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\Delta E(i,j), \tag{3.9}$$

where the size of the image is mxn. Knowing $\Delta E_{fluctuation}$ helps us identify which color should be assigned to which screen in order to decrease the graininess of the superposition image.

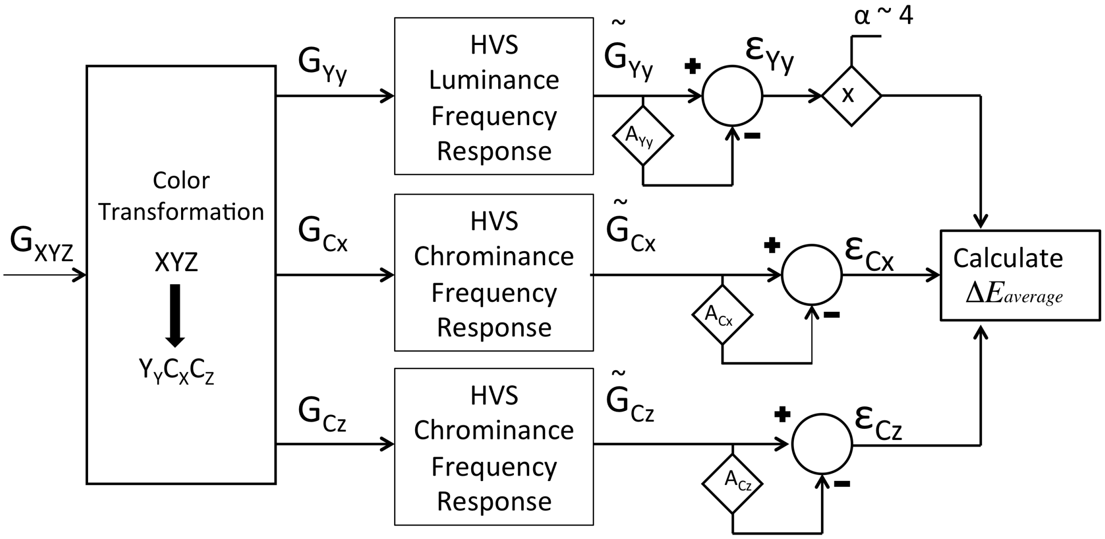


Fig. 3.7.: The procedure for obtaining the main components for calculating $\Delta E_{fluctuation}$. $\tilde{G}_{Y_y}(\mathbf{u})$, $\tilde{G}_{C_x}(\mathbf{u})$, and $\tilde{G}_{C_z}(\mathbf{u})$ represent the perceived Fourier Transforms of their corresponding images $\tilde{g}_{Y_y}[\mathbf{m}]$, $\tilde{g}_{C_x}[\mathbf{m}]$, and $\tilde{g}_{C_x}[\mathbf{m}]$.

3.2.3 Experimental results

Since the scope of our research lies in investigating the behavior of irregular clustered dot halftones, we chose to work with two irregular halftone screens and 3 colorants: cyan, magenta, and yellow. The first screen is an irregular screen with a periodicity matrix $\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$, and the second screen is also an irregular screen with a periodicity matrix $\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$. For the HP Indigo press with resolution 812.8 dpi, these screens have frequencies 176.32 and 172.42 lpi, respectively. Their angles are 12.53 and 45 degrees, respectively. Since we have 2 halftone screens and 3 colorants, we need to compare 6 different sets of data, defined and indexed in Table 3.1. Based on Table 3.1, with Superposition #1 the halftone with periodicity matrix $\mathbf{N_1}$ is cyan, and the halftone with periodicity matrix $\mathbf{N_2}$ is magenta. It is presented in Fig. 3.9. Figure 3.10 reveals the visualization of the opponent channels in sRGB. In order get a better insight of the opponent color channels, the zoomed-in view of Y_y, C_x, and C_z is provided in Fig. 3.8. The $Y_yC_xC_z$ matrix corresponding to

Table 3.1.: Six configurations for two halftone screens and three colorants: cyan, magenta, and yellow

Index (#)	$\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$	$\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$
1	Cyan	Magenta
2	Magenta	Cyan
3	Yellow	Magenta
4	Magenta	Yellow
5	Cyan	Yellow
6	Yellow	Cyan

the XYZ matrix mentioned in Equation 3.3 has the following form:

$$\begin{bmatrix} Y_y \\ C_x \\ C_z \end{bmatrix} = \begin{bmatrix} 101.51 & 33.03 & 11.75 & 122.80 \\ -124.05 & 140.59 & 44.21 & -27.67 \\ -50.96 & -145.85 & -177.08 & -19.73 \end{bmatrix} \tag{3.10}$$

Similar to Equation 3.3, columns 1 through 4 correspond to the cyan, magenta, cyan and magenta, and white pixels in the superposition image. From Fig. 3.8, it can be seen that the magenta pixel, which is the sum of red and blue, has a large red component in the C_x channel with its corresponding value of 140.59 and a large blue component in the C_z image with its corresponding value of -145.85 based on Equation 3.10. Whereas the cyan pixel, which is the sum of green and blue, has a large green component in the C_x channel with its value of -124.05, but a much smaller blue component in the C_z channel with its value of -50.96. Figure 3.11 is a visualization of the opponent color channels filtered with their corresponding luminance and chrominance

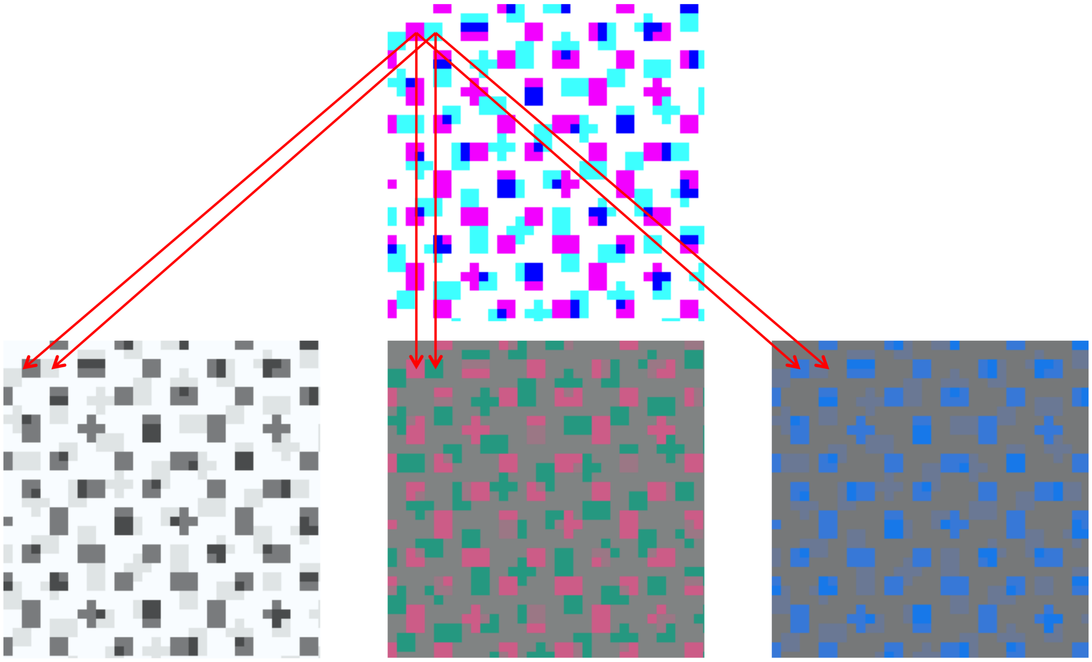


Fig. 3.8.: Zoomed-in view of Y_y, C_x, and C_z channels for the Superposition #1.

frequency responses depicted in the Methods section. The filtered $(Y_y, 0, 0)$ image has a strongly visible low frequency pattern, whereas the filtered C_x, and C_z images have no visible pattern due to the nature of the chrominance filter. As a comparison to the Superposition #1 results, the detailed results for the Superposition #2, when the halftone with periodicity matrix $\mathbf{N_2}$ is cyan, and the halftone with periodicity matrix $\mathbf{N_1}$ is magenta, are presented in Figs. 3.12, 3.13, and 3.14. Based on Fig. 3.14, it can be deduced that the filtered $(Y_y, 0, 0)$ image has a less visible low frequency pattern than the $(Y_y, 0, 0)$ image of Superposition #1, which is also verified by the calculated $\Delta E_{fluctuation}$ values, which are provided in Table 3.2, and supported by their corresponding images in Fig. 3.15. Our results emphasize that in the case of Ideal Block Reflectance and D65 Illuminant, we get a higher $\Delta E_{fluctuation}$ when applying the magenta colorant to a more irregular screen if choosing between either magenta and yellow, or magenta and cyan configurations. In this chapter, the more irregular screen is $\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$. That's why the $\Delta E_{fluctuation}$ for configurations #1 and #3 is higher than their complementary pairs. Similarly, $\Delta E_{fluctuation}$ is higher for configuration #6, when the cyan colorant is applied to a more irregular halftone rather than the yellow colorant. The results may seem intuitive for these simple cases. However, when considering different sets of regular and irregular screens, and different sets of Illuminants and Spectral Reflectance Functions, the results may no longer be easily predicted.

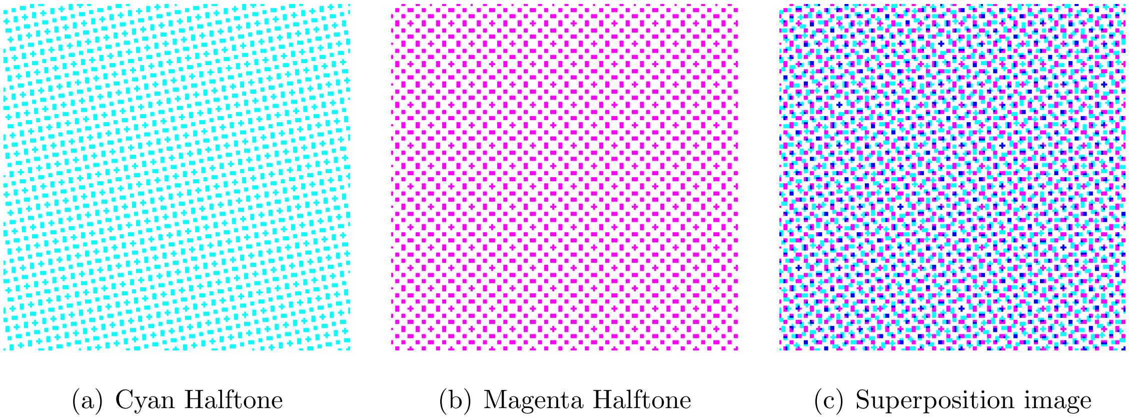


Fig. 3.9.: Superposition when the cyan halftone has a periodicity matrix $\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$ and the magenta halftone has a periodicity matrix $\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$

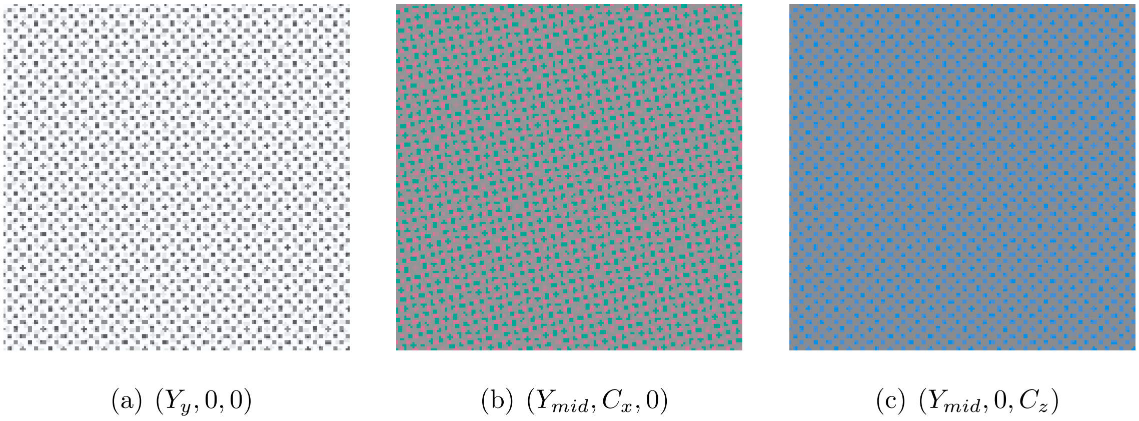
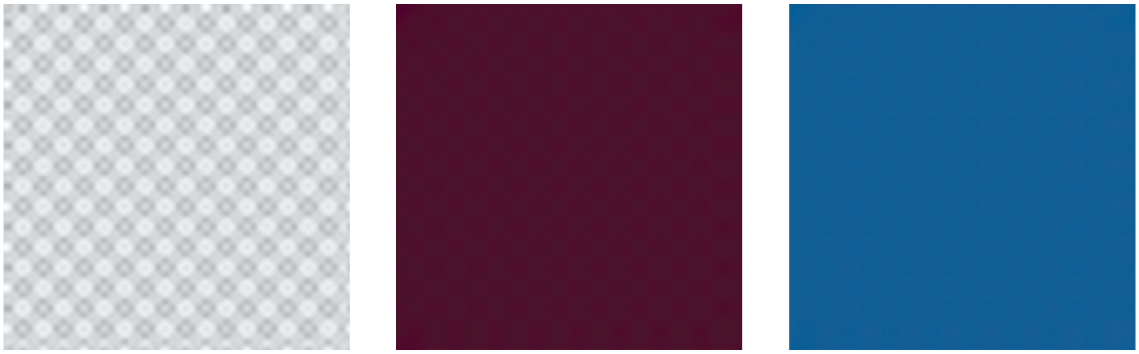
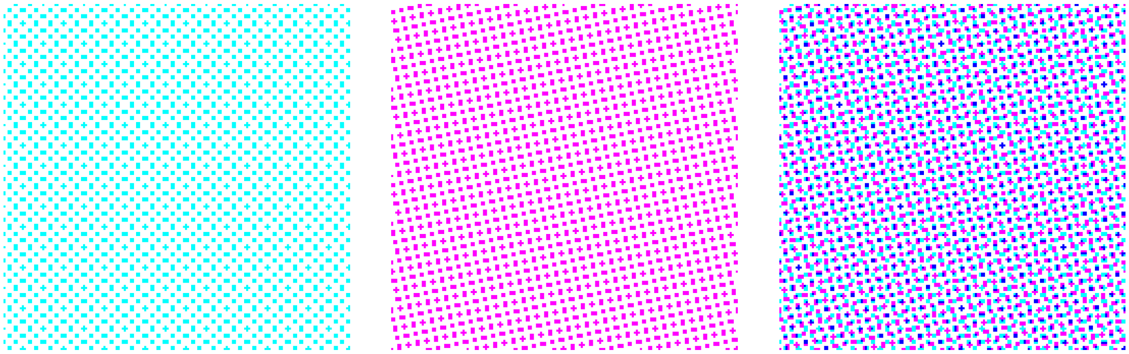


Fig. 3.10.: Visualization of Y_y, C_x, and C_z channels of the superposition image presented in Fig. 3.9 c).



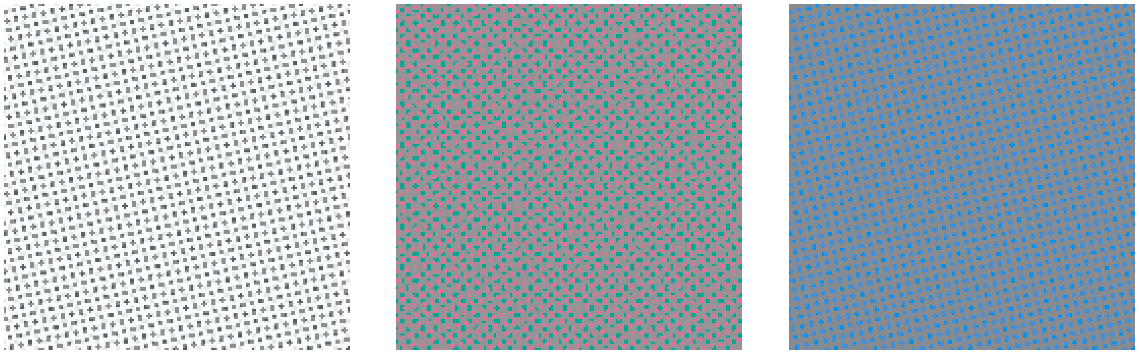
(a) $(Y_y, 0, 0)$ (b) $(Y_{mid}, C_x, 0)$ (c) $(Y_{mid}, 0, C_z)$

Fig. 3.11.: Visualization of Y_y, C_x, and C_z channels of the superposition image presented in Fig. 3.9 c), filtered with luminance and chrominance frequency responses



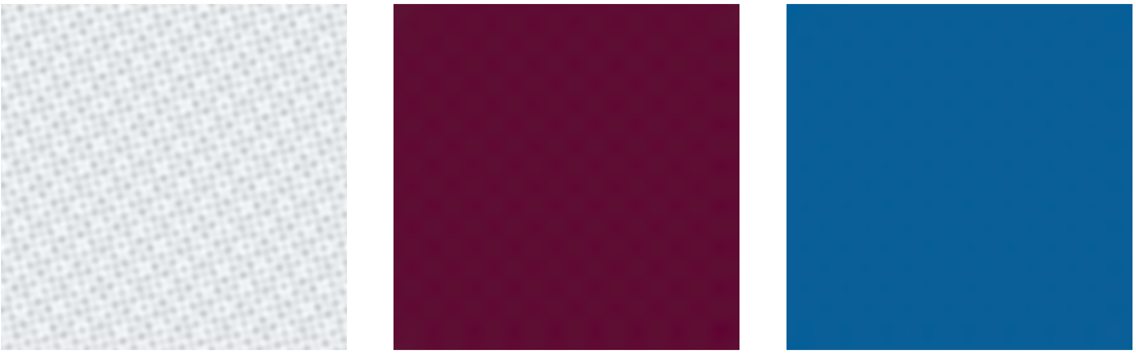
(a) Cyan Halftone (b) Magenta Halftone (c) Superposition image

Fig. 3.12.: Superposition when the cyan halftone has a periodicity matrix $\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$ and the magenta halftone has a periodicity matrix $\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$



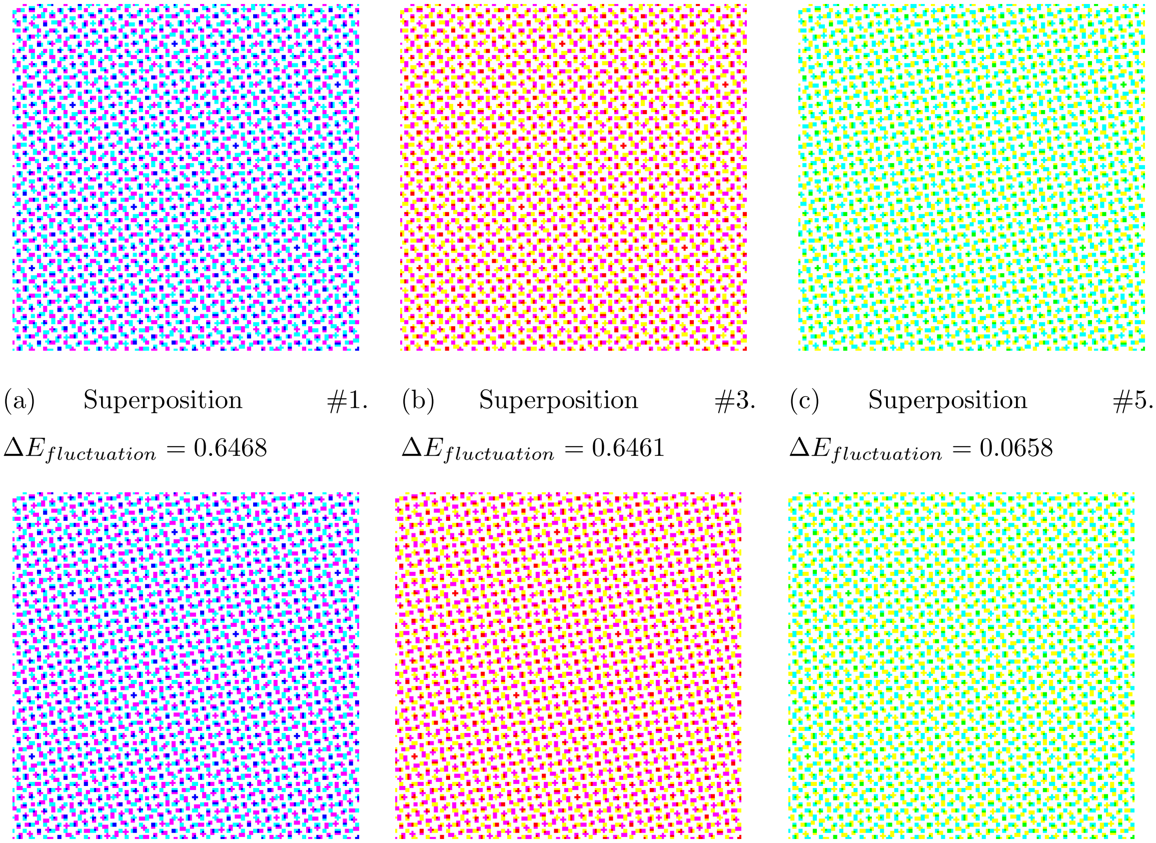
(a) $(Y_y, 0, 0)$ (b) $(Y_{mid}, C_x, 0)$ (c) $(Y_{mid}, 0, C_z)$

Fig. 3.13.: Visualization of Y_y, C_x, and C_z channels of the superposition image presented in Fig. 3.12 c).



(a) $(Y_y, 0, 0)$ (b) $(Y_{mid}, C_x, 0)$ (c) $(Y_{mid}, 0, C_z)$

Fig. 3.14.: Visualization of Y_y, C_x, and C_z channels of the superposition image presented in Fig. 3.12 c), filtered with luminance and chrominance frequency responses



(a) Superposition #1. $\Delta E_{fluctuation} = 0.6468$

(b) Superposition #3. $\Delta E_{fluctuation} = 0.6461$

(c) Superposition #5. $\Delta E_{fluctuation} = 0.0658$

(d) Superposition #2. $\Delta E_{fluctuation} = 0.1917$

(e) Superposition #4. $\Delta E_{fluctuation} = 0.1463$

(f) Superposition #6. $\Delta E_{fluctuation} = 0.1099$

Fig. 3.15.: Visualization of six configurations based on cyan, magenta, and yellow colorants and two periodicity matrices $\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$ and $\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$. The detailed information is provided in Table 3.2

Table 3.2.: Comparison between six different configurations for cyan, magenta, and yellow.

Superposition Index	$\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$	$\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$	$\Delta E_{fluctuation}$
1	Cyan	Magenta	0.6468
2	Magenta	Cyan	0.1917
3	Yellow	Magenta	0.6461
4	Magenta	Yellow	0.1463
5	Cyan	Yellow	0.0658
6	Yellow	Cyan	0.1099

3.2.4 Conclusion

We have presented the HVS-based model for the superposition of two clustered dot color halftones, which assists us in making the right choices when assigning colorants to a variety of irregular or regular screens. We used the $Y_yC_xC_z$ color space in order to account for the difference in the responses of the human viewer to luminance and chrominance information. Our model can be used to improve the predictions concerning the visibility of color halftone textures. In the next section, we'll show the result of expanding our model to the superposition of three and four clustered dot color halftones as well as looking at different absorptance combinations.

3.3 Analysis of Impact of Screen Configuration on Image Fluctuation and Color Shift (for misregistration) across all Absorptance Combinations

In this section, we present the results of the analysis of impact of screen configuration on image fluctuation and colorshift across all absorptance combinations. We start with the HVS-based model for the superposition of two clustered dot color halftones and we expand it to the superposition of three or four halftones. We use the $\Delta E_{fluctuation}$ metric as our measure for image fluctuation. The model helps us decide what are the best color assignments for the superposition of regular or irregular halftones that will minimize the perceived error.

In the previous section, we assumed one absorptance value for all separations. In this section, we have no restriction on absorptance value. In order to generate irregular clustered-dot halftones with all possible absorptance values, we started with the systematic approach for regular clustered-dot halftones that Baqai and Allebach proposed in [48], and extended it to the case of irregular clustered-dot halftones. We present our approach and example in the first subsection.

After applying our model to the superposition of three and four clustered-dot color halftones with different absorptance values, it was concluded that for different combinations of colorant absorptance values, their corresponding best color assignments

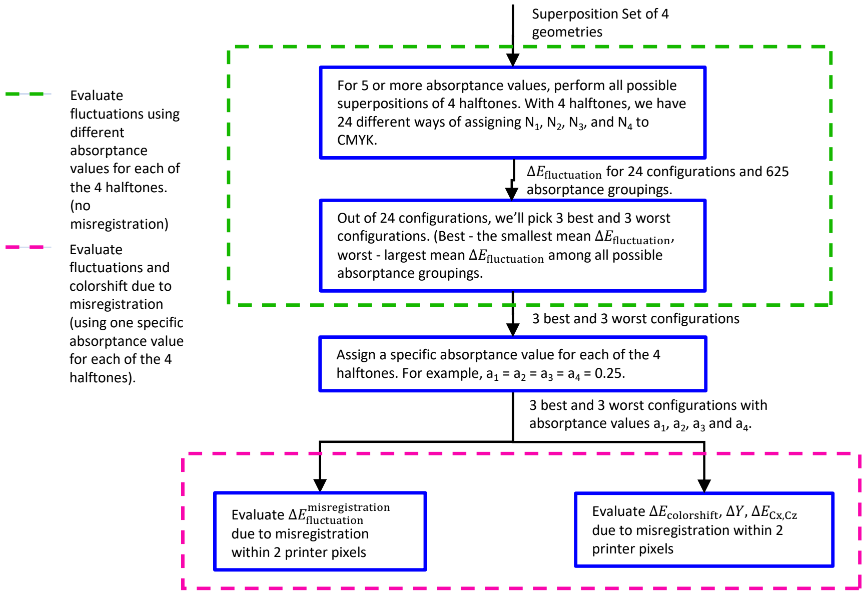
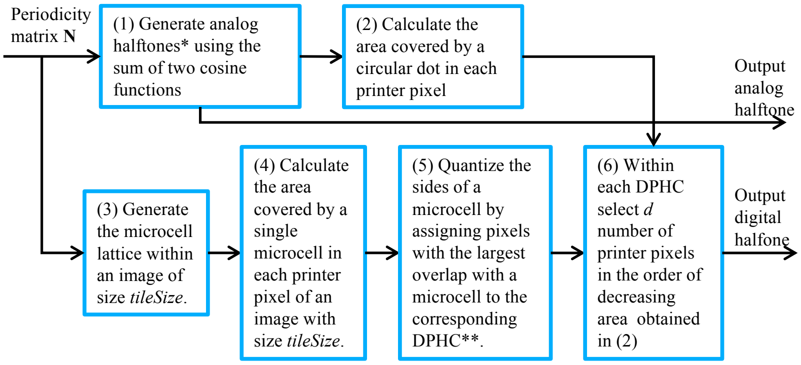

Superposition Set of 4 geometries
For 5 or more absorptance values, perform all possible superpositions of 4 halftones. With 4 halftones, we have 24 different ways of assigning N_1, N_2, N_3, and N_4 to CMYK.
$\Delta E_{\text{fluctuation}}$ for 24 configurations and 625 absorptance groupings.
Out of 24 configurations, we'll pick 3 best and 3 worst configurations. (Best - the smallest mean $\Delta E_{\text{fluctuation}}$, worst - largest mean $\Delta E_{\text{fluctuation}}$ among all possible absorptance groupings.
3 best and 3 worst configurations
Assign a specific absorptance value for each of the 4 halftones. For example, $a_1 = a_2 = a_3 = a_4 = 0.25$.
3 best and 3 worst configurations with absorptance values a_1, a_2, a_3 and a_4.
Evaluate $\Delta E_{\text{fluctuation}}^{\text{misregistration}}$ due to misregistration within 2 printer pixels
Evaluate $\Delta E_{\text{colorshift}}$, ΔY, $\Delta E_{\text{Cx,Cz}}$ due to misregistration within 2 printer pixels
Evaluate fluctuations using different absorptance values for each of the 4 halftones. (no misregistration)
Evaluate fluctuations and colorshift due to misregistration (using one specific absorptance value for each of the 4 halftones).

Fig. 3.16.: Block diagram for performing analysis of impact of screen configuration on image fluctuation and colorshift (for misregistration) across all absorptance combinations.

turn out to be different. Hence, it was decided that for a given superposition set, we need to look at different absorptance combinations and different color assignments in order to determine the best color assignment. On the other hand, given a specific set of absorptance values for each colorant, we evaluate the fluctuations and colorshift due to misregistration within 2 printer-addressable pixels. The block diagram for this section is provided in Fig. 3.16. We start with a superposition set of four geometries and five absorptance values, and perform all possible superpositions of four halftones. In the case of 4 periodicity matrices and 4 colorants, we have 24 different ways to assign colors to the periodicity matrices. In the case of 5 absorptance values and 4 periodicity matrices, we have 625 different combinations of absorptance values. After performing all possible combinations, we select 3 best and 3 worst configurations that'll be evaluated for image fluctuation and colorshift due to misregistration.

3.3.1 Screen design for irregular clustered-dot halftones.

The main challenge with the screen design of irregular clustered-dot halftones is that, unlike with regular halftones, the microcells within the tile are not identical. Hence, we have to generate all microcell growing sequences for all microcells within a tile, and then tile them. The full block diagram for our approach is presented in Fig. 3.17. Similarly to our theoretical analysis of Fourier spectra of continuous-space analog and digital halftones, we start by producing analog circular-dot halftones. In order to produce a checkerboard pattern at the 50% absorptance and circular patterns at all other levels, we use the approach described in [79]. The examples of three circular-dot halftones are presented in Fig. 3.18.



Periodicity matrix **N**
(1) Generate analog halftones* using the sum of two cosine functions
(2) Calculate the area covered by a circular dot in each printer pixel
Output analog halftone
(3) Generate the microcell lattice within an image of size *tileSize*.
(4) Calculate the area covered by a single microcell in each printer pixel of an image with size *tileSize*.
(5) Quantize the sides of a microcell by assigning pixels with the largest overlap with a microcell to the corresponding DPHC**.
(6) Within each DPHC select *d* number of printer pixels in the order of decreasing area obtained in (2)
Output digital halfone

d – number of printer pixels that need to be switched on based on a specific absorptance value.
*analog halftone is represented as a digital image, which is highly upsampled by a factor of 20.
**DPHC – discrete parameter halftone cell.

Fig. 3.17.: Block diagram for generating analog and digital halftones.

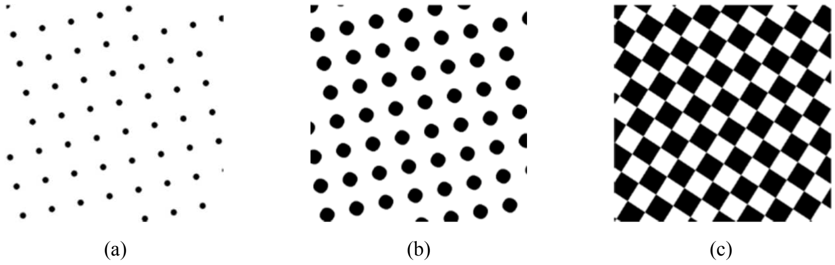
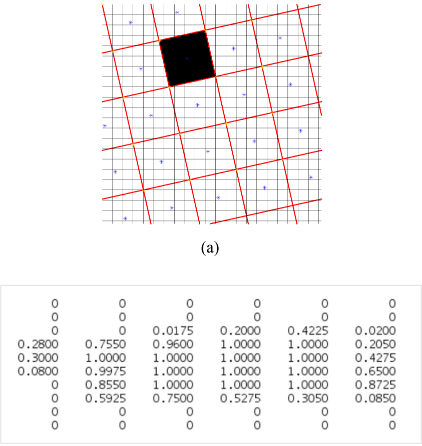


Fig. 3.18.: Example of producing analog rendering halftones. For this example, the screen frequency is 176.32 lpi, screen angle is 12.53°, and printer resolution is 812.8 dpi. (a) Absorptance = 0.05; (a) Absorptance = 0.25; (c) Absorptance = 0.50. The equation for producing circular dots is $t(x, y) = \left(-\frac{1}{2}\right)(\cos(2\pi f_1 x) + \cos(2\pi f_2 x))$, where f_1 and f_2 are derived from the periodicity matrix.

Next we, build the continuous parameter halftone cell lattice (CPHC) based on the periodicity matrix tile vectors, and then quantize the CPHC to obtain the DPHC [48], [47]. We quantize the sides of the CPHC by assigning printer pixels with the largest overlap with a CPHC to the corresponding DPHC. An example of computing CPHC overlaps with the printer grid is shown in Fig. 3.19. An example of quantizing the CPHC provided in Fig. 3.19 is shown in Fig. 3.20. Once all the DPHCs within the tile are obtained, we can select the desired number of printer pixels in the descending order of the area coverages of a circular dot in that DPHC. An example of generating digital halftones with absorptance values of 0.30 and 0.40 is shown in Fig. 3.21. For all of the figures in this section, we used the geometry with the screen frequency is 176.32 lpi, screen angle is 12.53°, and printer resolution is 812.8 dpi. The periodicity matrix for this geometry is $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$, hence the microcell area = 21.25 printer pixels. The continuous-tone ramp image halftoned using this specific geometry is shown in Fig. 3.22.

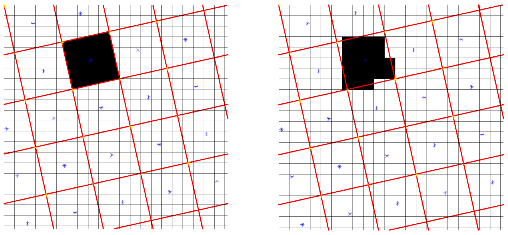


(a)

```
      0         0         0         0         0         0
      0         0         0         0         0         0
      0         0    0.0175    0.2000    0.4225    0.0200
 0.2800    0.7550    0.9600    1.0000    1.0000    0.2050
 0.3000    1.0000    1.0000    1.0000    1.0000    0.4275
 0.0800    0.9975    1.0000    1.0000    1.0000    0.6500
      0    0.8550    1.0000    1.0000    1.0000    0.8725
      0    0.5925    0.7500    0.5275    0.3050    0.0850
      0         0         0         0         0         0
      0         0         0         0         0         0
```

(b)

Fig. 3.19.: Example of computing area coverage by a single CPHC in each printer pixel. For this example, the screen frequency is 176.32 lpi, screen angle is 12.53°, and printer resolution is 812.8 dpi. (a) The red lines correspond to the microcell lattice, the background grid is the printer lattice, and the shaded region is the continuous parameter halftone cell (CPHC). (b) The area coverage (overlap) values of the printer-addressable pixels in a shaded region in (a).



(a)

(b)

Fig. 3.20.: Example of quantizing the sides of the CPHC by assigning pixels with the largest overlap to the corresponding DPHC. (a) The shaded cell is the CPHC. (b) The quantized CPHC - DPHC (discrete parameter halftone cell).

3.3.2 Dependence on absorptance grouping combination

In this section, we use two terms *configuration number* and *absorptance grouping number*. Periodicity matrix configuration number is a 4-digit number, where each

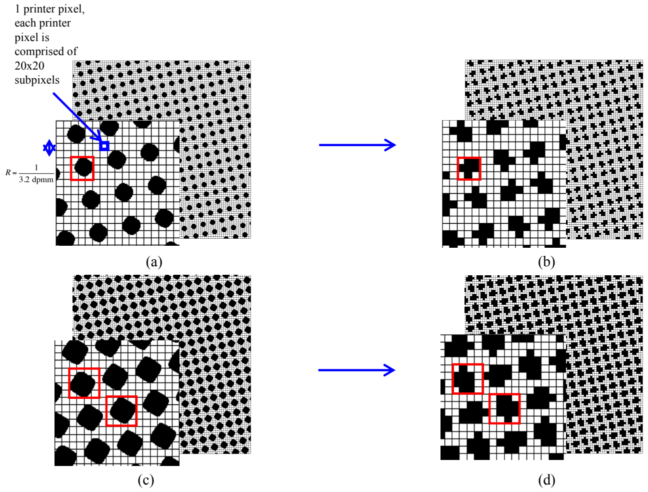

1 printer pixel, each printer pixel is comprised of 20x20 subpixels
$R = \frac{1}{3.2\ \text{dpmm}}$
(a)
(b)
(c)
(d)

Fig. 3.21.: Example of selecting a desired number of printer-addressable pixels in the order of decreasing area coverages of a circular dot in that DPHC. For this example, the screen frequency is 176.32 lpi, screen angle is 12.53°, and printer resolution is 812.8 dpi. The periodicity matrix is $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$, hence the microcell area = 21.25 printer pixels. Subfigures (a) and (b) correspond to the absorptance of 0.30. (a) The analog halftone for absorptance = 0.30, (b) The resulting digital halftone with 6 printer pixels switched on in each dot; Subfigures (c) and (d) correspond to the absorptance of 0.40. (c) The analog halftone for absorptance = 0.40, (d) The resulting digital halftone with 9 printer pixels switched on in each dot. The red boxes are placed in order to visualize the decreasing area coverages of a circular dot with the printer grid.

digit corresponds to a periodicity matrix number (1, 2, 3, and 4 correspond to periodicity matrices $\mathbf{N_1}, \mathbf{N_2}, \mathbf{N_3}$, and$\mathbf{N_4}$). The four digits are ordered in the order of their assignment to the (C, M, Y, K) colorants. For example, configuration number 4321 represents the assignment of C to $\mathbf{N_4}$, M to $\mathbf{N_3}$, Y to $\mathbf{N_2}$, and K to $\mathbf{N_1}$; 4312 means C is $\mathbf{N_4}$, M is $\mathbf{N_3}$, Y is $\mathbf{N_1}$, and K is $\mathbf{N_2}$.

Absorptance grouping numbers are indices for each possible combination of 5 absorptance values assigned to four separations. For the examples, provided in this section, the 5 absorptance values are 0.1, 0.3, 0.5, 0.7, 0.9. For example, absorp-

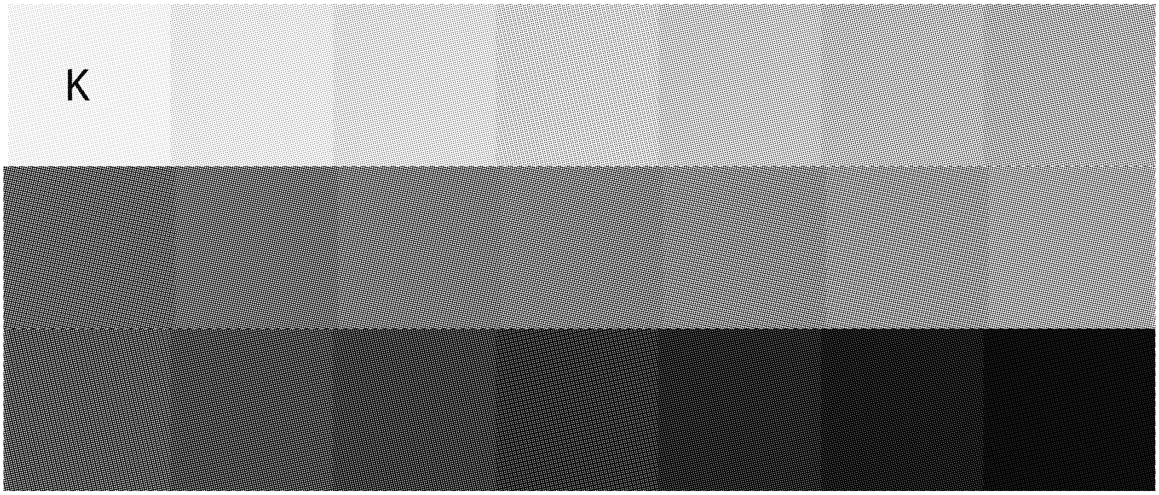


Fig. 3.22.: Example of generating all possible gray levels for the geometry with the screen frequency is 176.32 lpi, screen angle is 12.53°, and printer resolution is 812.8 dpi. The periodicity matrix for this geometry is $\mathbf{N} = \begin{bmatrix} \frac{9}{2} & -1 \\ 1 & \frac{9}{2} \end{bmatrix}$.

tance grouping number 1 is for $(a_1, a_2, a_3, a_4) = (0.1, 0.1, 0.1, 0.1)$, number 2 is for $(a_1, a_2, a_3, a_4) = (0.1, 0.1, 0.1, 0.3)$. Every value in the absorptance grouping number corresponds to a 4-tuple of absorptance values which are assigned to the (C, M, Y, K) colorants. For each of the 24 periodicity matrix configuration numbers, we obtain the minimum, maximum, and mean $\Delta E_{fluctuation}$ values among all 625 absorptance grouping numbers. Based on the mean $\Delta E_{fluctuation}$ over all absorptance grouping numbers, we select 3 best and 3 worst cases of periodicity matrix configuration numbers. The visualization of the images that we work with is illustrated in Fig. 3.23.

3.3.3 Dependence on misregistration

In this subsection, we investigate the effect of misregistration on image fluctuation and color shift, which are very common in electrophotographic printers. We perform the misregistration within 2 printer pixels. Due to the fact that we shift three halftones relative to the fourth, we have $9^3 = 729$ cases of misregistration (no shift, shift by 1, shift by 2 printer-addressable pixels). For each case, we'll calculate $\Delta E_{fluctuation}^{misregistration}$ and $\Delta E_{colorshift}$. The block diagram for computing $\Delta E_{fluctuation}^{misregistration}$ is shown in Fig. 3.25. For the human visual systems (HVSs), we used the Daly HVS for luminance and Mullen Chrominance HVS for chrominance channels [69], [70]. Their frequency responses are shown in Fig. 3.24.

The block diagram for computing $\Delta E_{colorshift}$ is shown in Fig. 3.26. Similarly to obtaining $\Delta E_{fluctuation}^{misregistration}$, we perform 729 cases of misregistration in order to obtain the $\Delta E_{colorshift}, \Delta Y$, and $\Delta E_{Cx,Cz}$. As for the reference image, we use the superposition image where no shifts are performed.

3.3.4 Experimental results

In order to demonstrate our results, we use the superposition set of four geometries, provided in Table 3.3. In order to generate the results in Table 3.3, we use three different sets of absorptance values and 4 halftones. Hence, for each set of absorptance

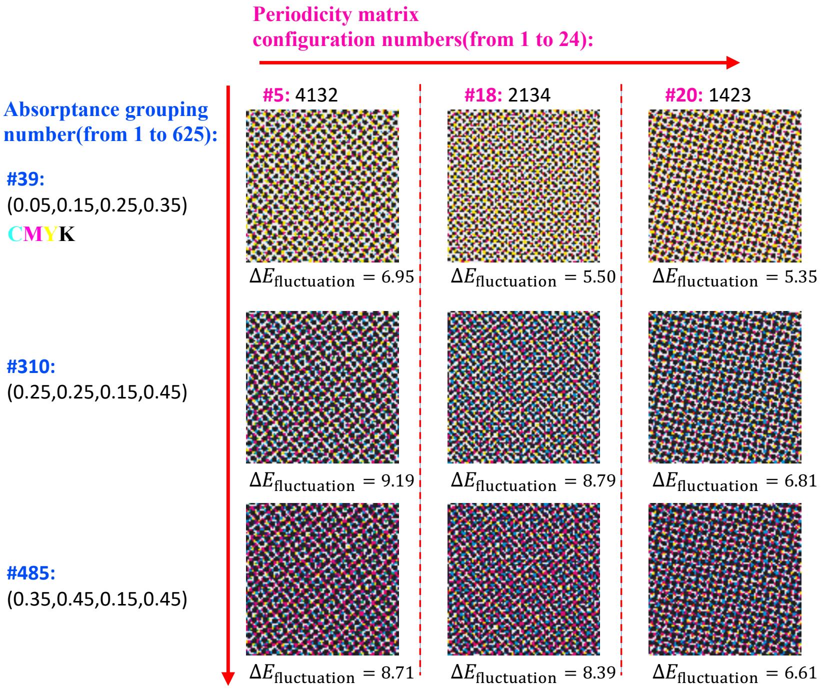
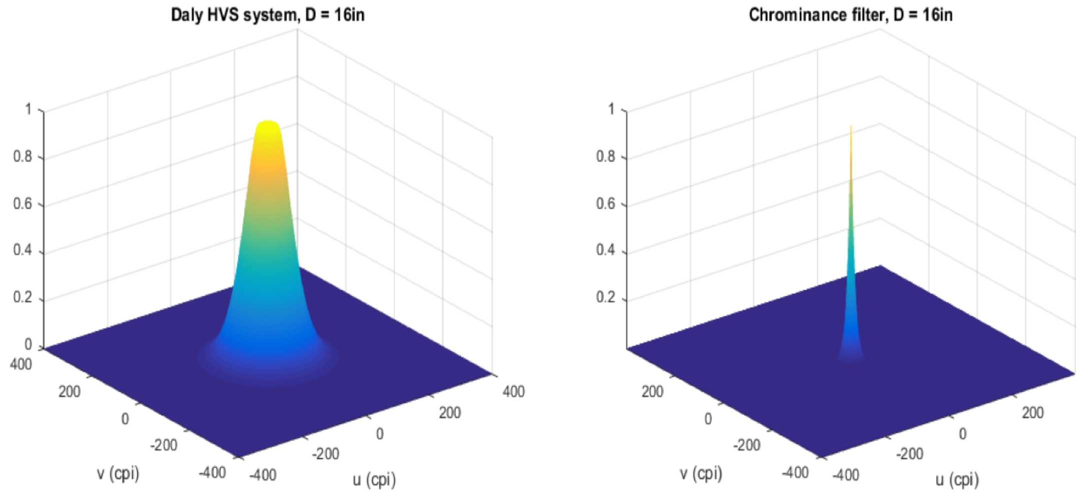

Periodicity matrix
configuration numbers(from 1 to 24):
#5: 4132
#18: 2134
#20: 1423
Absorptance grouping
number(from 1 to 625):
#39:
(0.05,0.15,0.25,0.35)
CMYK
#310:
(0.25,0.25,0.15,0.45)
#485:
(0.35,0.45,0.15,0.45)
$\Delta E_{\text{fluctuation}} = 6.95$
$\Delta E_{\text{fluctuation}} = 5.50$
$\Delta E_{\text{fluctuation}} = 5.35$
$\Delta E_{\text{fluctuation}} = 9.19$
$\Delta E_{\text{fluctuation}} = 8.79$
$\Delta E_{\text{fluctuation}} = 6.81$
$\Delta E_{\text{fluctuation}} = 8.71$
$\Delta E_{\text{fluctuation}} = 8.39$
$\Delta E_{\text{fluctuation}} = 6.61$

Fig. 3.23.: Visualization of the images that are used in order to obtain the best color assignment depending on different absorptance values. Absorptance grouping numbers vary from row to row, whereas periodicity matrix configuration numbers vary from column to column. In order to select the 3 best and 3 worst configurations, the mean $\Delta E_{fluctuation}$ over all absorptance grouping numbers in each configuration set is computed.



(a) (b)

Fig. 3.24.: Daly and Mullen Chrominance human visual systems frequency responses. (a) Daly HVS; (b) Mullen Chrominance HVS.

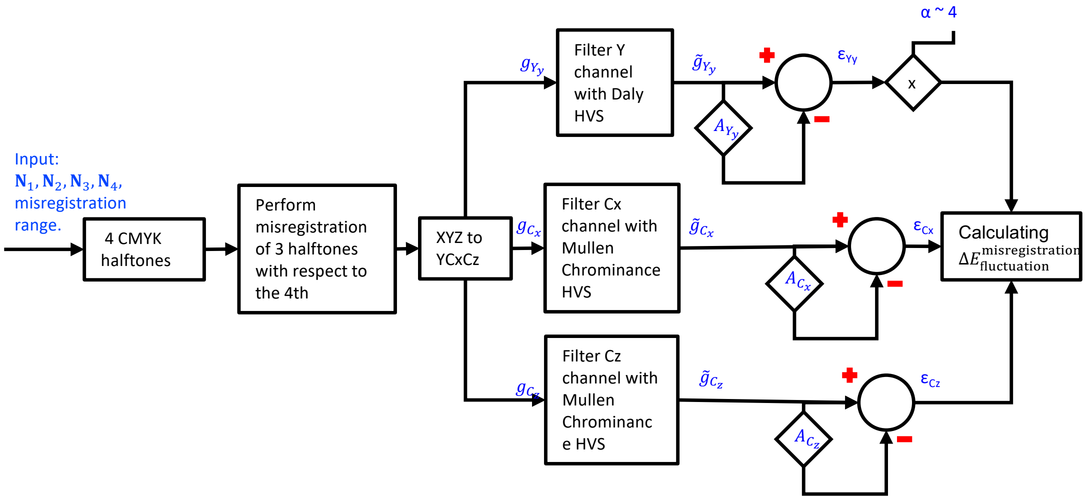
values, we have $5^4 = 625$ possible absorptance grouping numbers. We obtained mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers for each of the 24 configurations, and sorted the results in increasing order from the minimum mean $\Delta E_{fluctuation}$ to the maximum mean $\Delta E_{fluctuation}$ to obtain the 3 best and 3 worst cases. We used three different sets of absorptance values: a) highlight levels 0.05 - 0.45 with increment of 0.1, b) shadow levels 0.55 - 0.95 with increment of 0.1; c) all levels 0.1 - 0.9 with increment of 0.2. The result is provided in Fig. 3.27.

From Fig. 3.27, it can be concluded that configuration #17 is the best. Configuration #17 is 2143, which means that we need to assign the highest frequency halftone (in this case, geometry with $\mathbf{N_4}$) to Yellow. Configuration #18 is the worst.

After performing misregistration and evaluating $\Delta E_{fluctuation}^{misregistration}$ and $\Delta E_{colorshift}$ on the three best and worst cases described in Fig. 3.27, we derived two main observations regarding the effect of misregistration on image fluctuation and color shift for this superposition set. The first observation is that the three best configurations that are obtained by looking at different absorptance combinations are more robust to the

Table 3.3.: Superposition set used for the experimental results.

Periodicity matrix	**Periodicity matrix (fractional)**	**Screen Frequency (lpi)**	**Screen Angle (degree)**	**Tile based BSB**	**Supercell based BSB**
$\mathbf{N}_1 = \begin{bmatrix} 4.42 & -0.32 \\ 0.32 & 4.42 \end{bmatrix}$	$\mathbf{N}_1 = \begin{bmatrix} \frac{3240}{733} & -\frac{240}{733} \\ \frac{240}{733} & \frac{3240}{733} \end{bmatrix}$	183.38	4.24	120	87960
$\mathbf{N}_2 = \begin{bmatrix} 3.72 & -2.53 \\ 2.53 & 3.72 \end{bmatrix}$	$\mathbf{N}_2 = \begin{bmatrix} \frac{2640}{709} & -\frac{1800}{709} \\ \frac{1800}{709} & \frac{2640}{709} \end{bmatrix}$	180.35	34.29	120	85080
$\mathbf{N}_3 = \begin{bmatrix} 2 & -4 \\ 4 & 2 \end{bmatrix}$	$\mathbf{N}_3 = \begin{bmatrix} 2 & -4 \\ 4 & 2 \end{bmatrix}$	181.75	63.44	80	10
$\mathbf{N}_4 = \begin{bmatrix} 4 & -1.33 \\ 1.33 & 4 \end{bmatrix}$	$\mathbf{N}_3 = \begin{bmatrix} 4 & -\frac{4}{3} \\ \frac{4}{3} & 4 \end{bmatrix}$	192.77	18.43	80	40


Input:
$\mathbf{N}_1, \mathbf{N}_2, \mathbf{N}_3, \mathbf{N}_4$,
misregistration
range.
4 CMYK halftones
Perform misregistration of 3 halftones with respect to the 4th
XYZ to YCxCz
g_{Y_y}
g_{C_x}
g_{C_z}
Filter Y channel with Daly HVS
Filter Cx channel with Mullen Chrominance HVS
Filter Cz channel with Mullen Chrominanc e HVS
$\tilde{g}_{Y_y}$
$\tilde{g}_{C_x}$
$\tilde{g}_{C_z}$
A_{Y_y}
A_{C_x}
A_{C_z}
ε_{Yy}
ε_{Cx}
ε_{Cz}
α ~ 4
x
Calculating $\Delta E^{\text{misregistration}}_{\text{fluctuation}}$

Note: A_{Yy}, A_{Cx},A_{Cz} – the mean value of corresponding Y_y, C_x, and C_z

Fig. 3.25.: The block diagram for computing the perceived error due to misregistration. Given $\epsilon_Y, \epsilon_{Cx}, \epsilon_{Cz}$ for each pixel i, we calculate $\Delta E(i) = \sqrt{\epsilon_Y^2 + \epsilon_{Cx}^2 + \epsilon_{Cz}^2}$, and then $\Delta E^{misregistration}_{fluctuation} = \frac{1}{n}\Sigma_{i=1}^{n}\Delta E(i)$. Here n is the number of pixels in the image.

image fluctuation than the three worst cases. In order to visualize the impact of the first observation, we use the three best and three worst cases obtained after evaluating mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers for absorptance values 0.05-0.45 (increment = 0.1) shown in the first column of Fig. 3.27. As mentioned before, for each separation, we choose one absorptance value, and perform all possible shifts within 2 printer-addressable pixels. In this case, we choose to work with the absorptance of 0.25 for each separation. For each configuration of the three best and three worst cases, we compute the mean, min, max and stdev of $\Delta E^{misregistration}_{fluctuation}$. The results are summarized in Fig. 3.28. It can be seen that configurations with smaller

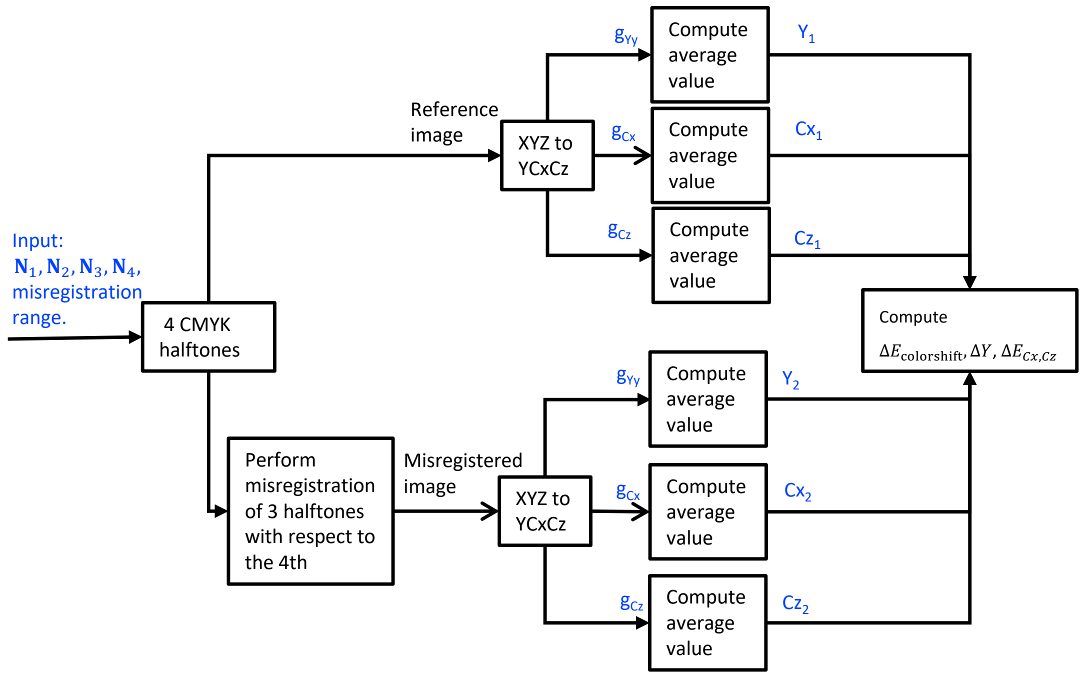
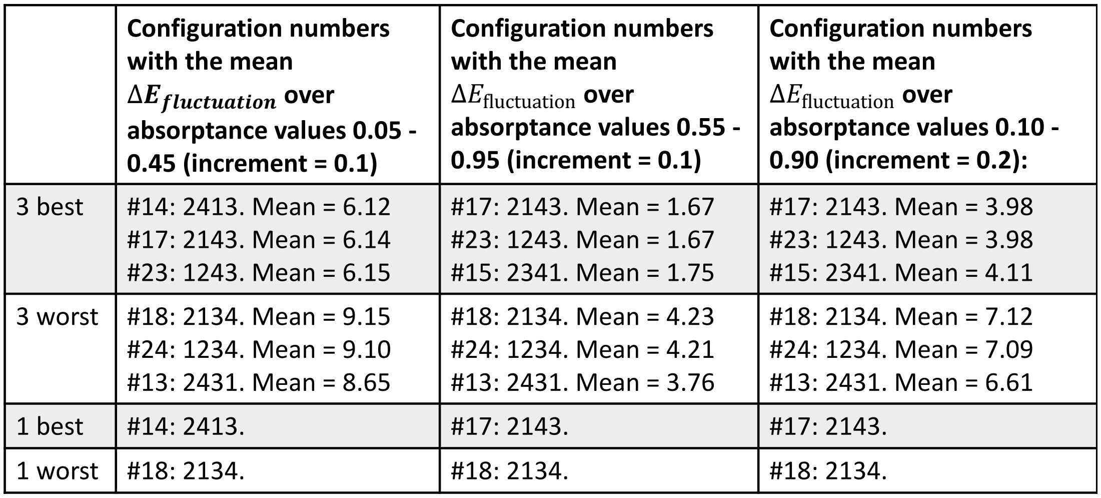


Fig. 3.26.: The block diagram for computing the global colorshift due to misregistration. Here, $\Delta E_{colorshift} = \sqrt{(Y_1 - Y_2)^2 + (Cx_1 - Cx_2)^2 + (Cz_1 - Cz_2)^2}, \Delta Y = Y_1 - Y_2$, and $\Delta E_{Cx,Cz} = \sqrt{(Cx_1 - Cx_2)^2 + (Cz_1 - Cz_2)^2}$.

mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers (the best 3 cases, in this example #14, #17 and #23) are more robust to misregistration than the 3 worst cases. It may that happen that in some cases, the maximum $\Delta E_{fluctuation}^{misregistration}$for the best case may be larger than the maximum $\Delta E_{fluctuation}^{misregistration}$ for the worst case. However, the mean $\Delta E_{fluctuation}^{misregistration}$ for the best case has only been recorded to be smaller than the mean $\Delta E_{fluctuation}^{misregistration}$ for the worst case. Hence, it can be concluded that the best case that we choose while evaluating $\Delta E_{fluctuation}$ among different absorptance groupings, will be the most robust to misregistration. In addition, the

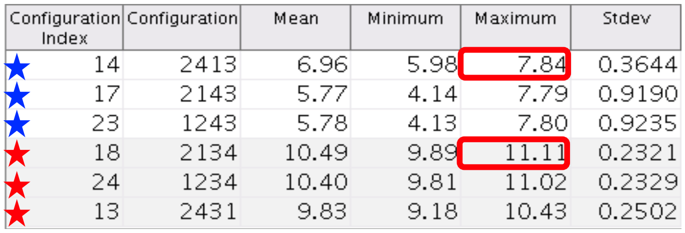


	Configuration numbers with the mean $\boldsymbol{\Delta E_{fluctuation}}$ over absorptance values 0.05 - 0.45 (increment = 0.1)	**Configuration numbers with the mean $\Delta E_{\text{fluctuation}}$ over absorptance values 0.55 - 0.95 (increment = 0.1)**	**Configuration numbers with the mean $\Delta E_{\text{fluctuation}}$ over absorptance values 0.10 - 0.90 (increment = 0.2):**
3 best	#14: 2413. Mean = 6.12 #17: 2143. Mean = 6.14 #23: 1243. Mean = 6.15	#17: 2143. Mean = 1.67 #23: 1243. Mean = 1.67 #15: 2341. Mean = 1.75	#17: 2143. Mean = 3.98 #23: 1243. Mean = 3.98 #15: 2341. Mean = 4.11
3 worst	#18: 2134. Mean = 9.15 #24: 1234. Mean = 9.10 #13: 2431. Mean = 8.65	#18: 2134. Mean = 4.23 #24: 1234. Mean = 4.21 #13: 2431. Mean = 3.76	#18: 2134. Mean = 7.12 #24: 1234. Mean = 7.09 #13: 2431. Mean = 6.61
1 best	#14: 2413.	#17: 2143.	#17: 2143.
1 worst	#18: 2134.	#18: 2134.	#18: 2134.

Fig. 3.27.: Impact of screen configuration on image fluctuation across different absorptance combinations. The results in this figure are generated for the superposition set shown in Table 3.3. It can be concluded that configuration #17 is the best. Configuration #17 is 2143, which means that we need to assign the highest frequency halftone (in this case, geometry with $\mathbf{N_4}$) to Yellow. Configuration #18 is the worst.

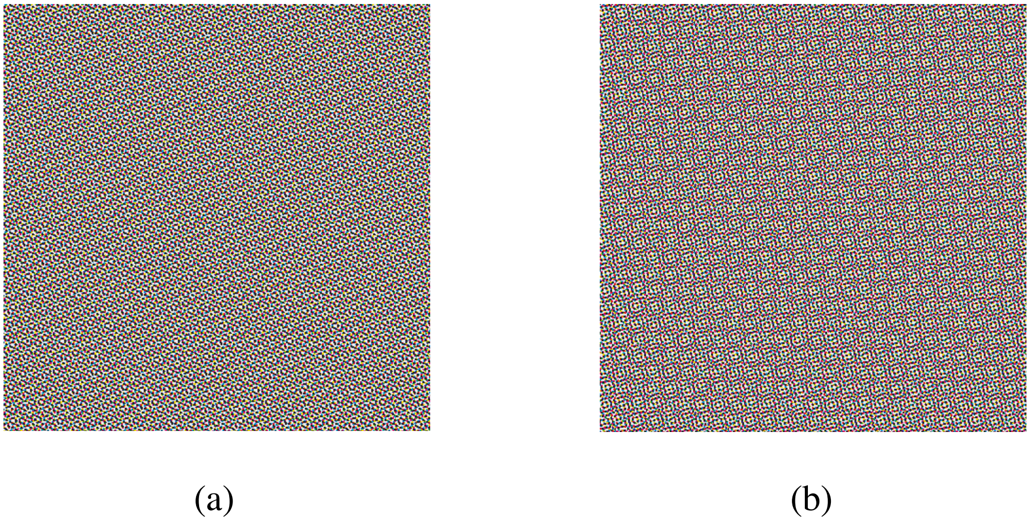
visualization of the halftones with the maximum $\Delta E^{misregistration}_{fluctuation}$ for the best and the worst configurations is provided in Fig 3.29.

The second observation is related to the effect of misregistration on color shift. In order to visualize the impact of the second observation, we use the three best and three worst cases obtained after evaluating mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers for absorptance values 0.55-0.95 (increment = 0.1) shown in the second column of Fig. 3.27. For each separation, we choose one absorptance



Configuration Index	Configuration	Mean	Minimum	Maximum	Stdev
★ 14	2413	6.96	5.98	7.84	0.3644
★ 17	2143	5.77	4.14	7.79	0.9190
★ 23	1243	5.78	4.13	7.80	0.9235
★ 18	2134	10.49	9.89	11.11	0.2321
★ 24	1234	10.40	9.81	11.02	0.2329
★ 13	2431	9.83	9.18	10.43	0.2502

Fig. 3.28.: The table contains the values of the mean, min, max and stdev of $\Delta E_{fluctuation}^{misregistration}$ due to misregistration within 2 printer pixels (729 cases of misregistration) for the superposition set shown in Table 3.3. Absorptance is 0.25 for each separation. Configuration indices in the 1st column are the 3 best and 3 worst cases obtained after evaluating mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers for absorptance values 0.05-0.45 (increment = 0.1) shown in Fig. 3.27. Configurations with smaller mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers (the best 3 cases, in this example #14, #17 and #23) are more robust to misregistration than the 3 worst cases. The maximum $\Delta E_{fluctuation}^{misregistration}$ for the best case is 7.84, whereas the maximum $\Delta E_{fluctuation}^{misregistration}$ for the worst case is 11.11 (both results are highlighted in red and their visualization is provided in Fig 3.29).



(a)

(b)

Fig. 3.29.: The example with the maximum image fluctuation due to misregistration within the range of two printer-addressable pixels: (a) Configuration #14 (2413) with max $\Delta E_{fluctuation}^{misregistration} = 7.84$. Superposition image with the following misregistration: geometry 1 has shift coordinates $[0, 0]$, geometry 2 has shift coordinates $[2, 0]$, geometry 3 has shift coordinates $[0, 0]$, and geometry 4 has shift coordinates $[1, 2]$. (b) Configuration #18 (2134) with max $\Delta E_{fluctuation}^{misregistration} = 11.11$. Superposition image with the following misregistration: geometry 1 has shift coordinates $[0, 0]$, geometry 2 has shift coordinates $[0, 2]$, geometry 3 has shift coordinates $[1, 0]$, and geometry 4 has shift coordinates $[2, 1]$.

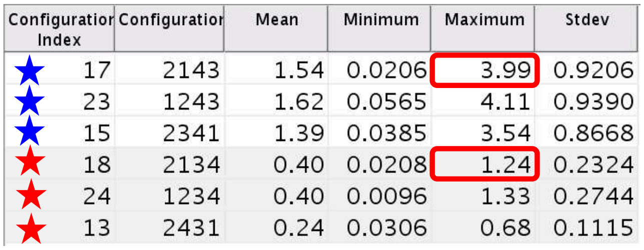
value, and perform all possible shifts within 2 printer-addressable pixels. In this case, we choose to work with the absorptance of 0.75 for each separation. For each configuration of the three best and three worst cases, we compute the mean, min, max and stdev of $\Delta E_{fluctuation}^{misregistration}$. The results are summarized in Fig. 3.30.

We observed that in some cases, not always, configurations with larger mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers (the worst 3 cases, in this example #18, #24 and #13) are more robust to colorshifts due to misregistration than the 3 best cases. However, even though the maximum $\Delta E_{colorshift}$ for the best cases are larger than the maximum $\Delta E_{colorshift}$ for the 3 worst cases, the $\Delta E_{fluctuation}^{misregistration}$ for the worst cases is usually larger than the $\Delta E_{fluctuation}^{misregistration}$ for the best cases. Hence, there is a trade-off between the image fluctuation and color shift.

3.3.5 Conclusion

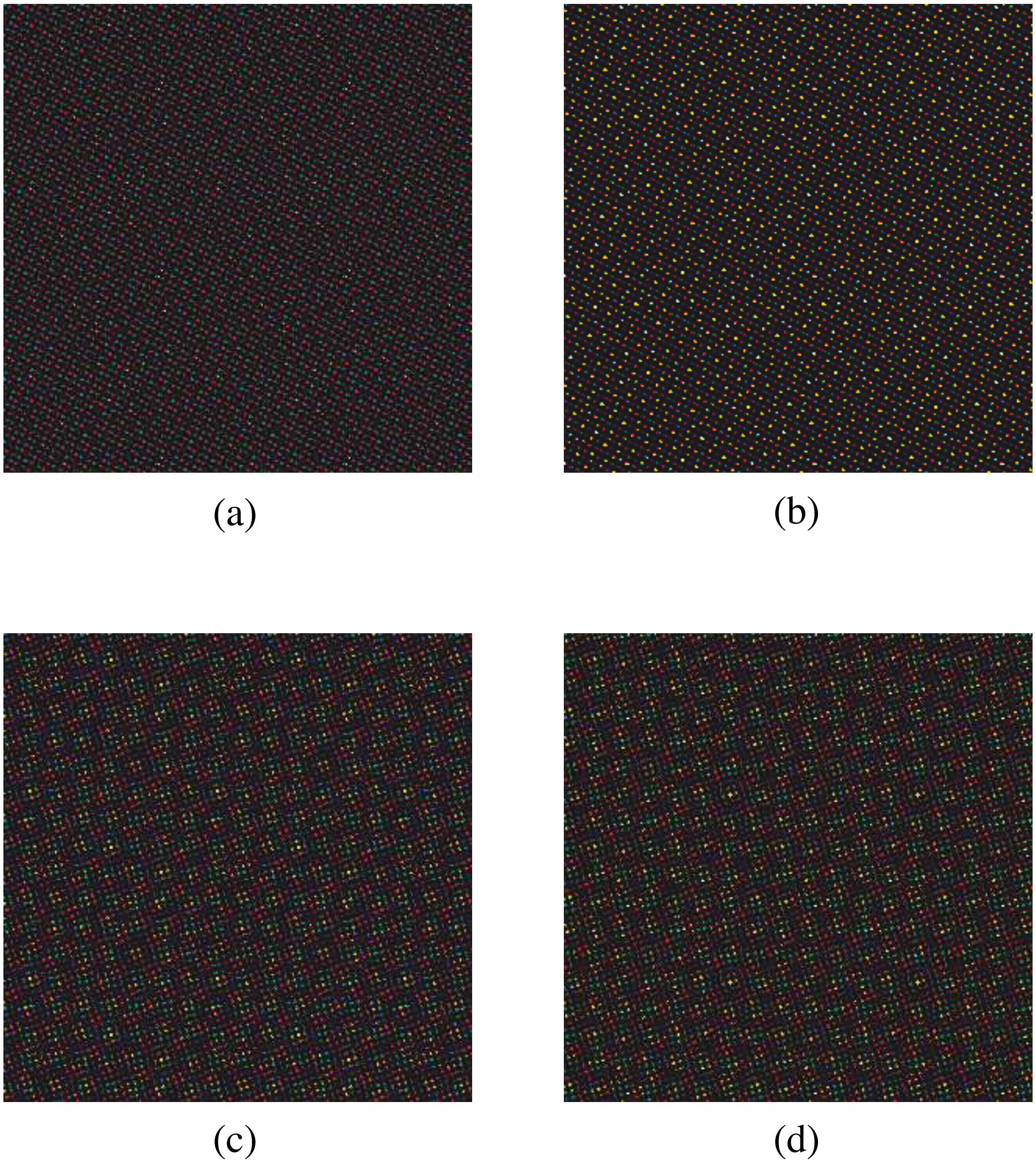
In this section, we investigated the impact of screen configuration on image fluctuation and colorshift across all absorptance combinations. We started with the HVS-based model for the superposition of two clustered dot color halftones and we expanded it to the superposition of four halftones. We used the $\Delta E_{fluctuation}$ metric as our measure for image fluctuation and obtained the best and the worst configurations across different combinations of absorptance values. We concluded that for different combinations of absorptance values their best and worst configurations maybe different. Hence, in the next chapter, we propose to apply content-color-dependent screening in order to generate the best quality halftones.

After determining on average the best and worst configurations across different absorptance combinations, we investigated the effect of misregistration within the range of 2 printer-addressable pixels on image fluctuation and color shift. We concluded that the three best configurations were more robust to misregistration than the three worst configurations, i.e. their $\Delta E_{fluctuation}^{misregistration}$ values were smaller than those for the worst configurations. On the other hand, we found out that sometimes



Configuratior Index	Configuratior	Mean	Minimum	Maximum	Stdev
★ 17	2143	1.54	0.0206	3.99	0.9206
★ 23	1243	1.62	0.0565	4.11	0.9390
★ 15	2341	1.39	0.0385	3.54	0.8668
★ 18	2134	0.40	0.0208	1.24	0.2324
★ 24	1234	0.40	0.0096	1.33	0.2744
★ 13	2431	0.24	0.0306	0.68	0.1115

Fig. 3.30.: The table contains the values of the mean, min, max and stdev of $\Delta E_{colorshift}$ due to misregistration within 2 printer pixels (729 cases of misregistration) for the superposition set shown in Table 3.3. Absorptance is 0.75 for each separation. Configuration indices in the 1st column are the 3 best and 3 worst cases obtained after evaluating mean $\Delta E_{fluctuation}$ among all 625 absorptance grouping numbers for absorptance values 0.55-0.95 (increment = 0.1) shown in Fig. 3.27. The maximum $\Delta E_{colorshift}$ for the best case is 3.99, whereas the maximum $\Delta E_{colorshift}$ for the worst case is 1.24(both results are highlighted in red and their visualization is provided in Fig. 3.31).


(a)

(b)

(c)

(d)

Fig. 3.31.: The example with the maximum colorshift due to misregistration within the range of two printer-addressable pixels: (a) Configuration #17 (2143) is the reference image; b) Configuration #17 (2143) with max $\Delta E_{colorshift} = 3.99$(in addition its $\Delta Y = 1.54$ and $\Delta E_{CxCz} = 3.68$); (c) Configuration #18 (2134) is the reference image; d) Configuration #18 (2134) with max $\Delta E_{colorshift} = 1.24$ (in addition its $\Delta Y = 0.12$ and $\Delta E_{CxCz} = 1.23$). Superposition image in (b) has the following shifts: geometry 1 has shift coordinates $[0,0]$, geometry 2 has shift coordinates $[1,1]$, geometry 3 has shift coordinates $[2,0]$, and geometry 4 has shift coordinates $[0,1]$. Superposition image in (d) has the following shifts: geometry 1 has shift coordinates $[0,0]$, geometry 2 has shift coordinates $[2,2]$, geometry 3 has shift coordinates $[2,0]$, and geometry 4 has shift coordinates $[0,1]$.

the worst configurations are more robust to the color shift than the best configurations, however the worst configurations look less smooth than the best configuration images. Hence, there is a tradeoff between image fluctuation and color shift.

4. CONTENT-COLOR-DEPENDENT SCREENING (CCDS) USING REGULAR OR IRREGULAR CLUSTERED-DOT HALFTONES

4.1 Introduction

The purpose of our research is to improve print quality in high end digital presses. In this chapter, we focus on improving the quality of color images. Most color printers use four different colorants, typically cyan, magenta, yellow, and black (CMYK). In order to produce a color image, each colorant plane is halftoned independently, as if it were a separate monochrome image. Superimposing the four colorant planes halftoned with screens rotated to different angles will give the final image. Superposition of two or more lattices at different angles can produce moire and rosette artifacts. The moire phenomenon refers to a low-frequency structure, which is easily seen at a normal viewing distance. The rosette pattern has circular or polygonal patterns, which is formed as a result of rotating the halftone screens [68]. The conventional screen angles that minimize visible moire and rosette patterns are 75°, 15°, 0°, and 45°for cyan, magenta, yellow, and black, respectively [68], [45]. A slight change in the screen angle or frequency can result in more visible moire and rosette patterns. There are a number of methods for choosing a set of, say four, regular and/or irregular screens that will not result in artifacts when halftone patterns generated with these screens are superimposed [56], [50]. But how to assign a fixed set of screens to individual colorants has been less thoroughly investigated. In our previous work, we presented a method for choosing the best color assignments to two regular or irregular halftones in order to minimize the perceived error. We developed a model based on the human visual system. In order to account for the difference in the responses of the human viewer to luminance and chrominance information, we used the YyCxCz color space. The perceived error helped us to identify the configuration of colors and screens that will improve the appearance of the superposition image [66]. For example, given two irregular halftones with periodicity matrices $\mathbf{N_1} = \begin{bmatrix} 9/2 & -1 \\ 1 & 9/2 \end{bmatrix}$, and $\mathbf{N_2} = \begin{bmatrix} 10/3 & -10/3 \\ 10/3 & 10/3 \end{bmatrix}$, two colorants cyan and magenta with absorptance of 0.25, it was concluded that assigning $\mathbf{N_1}$ to magenta and $\mathbf{N_2}$ to cyan will yield a better result than assigning $\mathbf{N_2}$ to magenta and $\mathbf{N_1}$ to cyan [66].

In the case of 4 periodicity matrices and 4 colorants, we have 24 different ways to assign colors to the periodicity matrices. In addition, if colorants have different absorptance values, then we have to look at all possible combinations of absorptance values. Since any image can contain any color content, i.e. different absorptance values for each colorant, it was decided that the image needed to be clustered. Next, for each cluster in the image, the corresponding best color assignment can be determined and the image can be halftoned with the best color assignments depending on the color content. It turned out that when we switched color assignments inside smooth areas, the transition from one color assignment to another caused an artifact. Hence, an edge detection algorithm was added. Finally, using the cluster-map and the edge-map, we were able to segment the image based on the color content, and halftone each segment with its optimal color assignment, while not segmenting smooth regions with similar color content into separate regions.

The concept of performing image segmentation based on the content and using different halftoning techniques in different regions of the image was investigated before. Park et al developed a method in which they divided a document into smooth and detail objects, and used low-frequency, periodic clustered-dot halftoning in smooth areas to promote stable development, and high-frequency, periodic clustered-dot halftoning in detail areas to provide better rendering of the detail in the image [58]. Ostromoukhov and Nehab checked the local gradient at each pixel, and chose a basic dither matrix based on the magnitude of the gradient [59]. Huang and Bhattacharjya described a process for switching between a periodic, clustered-dot screen in smooth areas, and error diffusion with a screen in detail areas [60]. The screen used for both detail and smooth regions is the same. References [59] and [60] address only halftoning of monochrome images. Reference [58] does consider color. But it is targeted to home/office laser electrophotographic printers, not high-end digital presses. The main novelty of our approach, which is targeted to printing with high-end digital presses, is the application of different color assignments within the image depending on the local color and content of the image, without changing the overall set of screens that are used.

In this chapter, we will use irregular clustered-dot halftones to demonstrate our results. In order to generate irregular clustered-dot halftones, we followed the method proposed by Baqai and Allebach [48]. Baqai and Allebach presented a systematic method for designing regular clustered-dot halftones based on the periodicity matrix [48]. We extended their approach to design of irregular clustered-dot halftones.

4.2 The procedure for performing content-color-dependent screening

The procedure implemented in our research consists of four parts. In Sec. 4.2.1, the K-means algorithm to cluster the color content is described. In Sec. 4.2.2, an algorithm for obtaining the segmented edge-map is described. In Sec. 4.2.3, merging of the cluster-map and the segmented edge-map into a final map is explained. In Sec. 4.2.4, the approach for obtaining the best color assignment is reviewed and two examples are provided. Finally, the image can be halftoned using the final map of clusters and their corresponding best color assignments. The complete block diagram is presented in Fig. 4.1.

4.2.1 Generation of the cluster-map using K-means

K-means clustering is a type of unsupervised learning, which can be used when we have unlabeled data that needs to be clustered or categorized into groups based

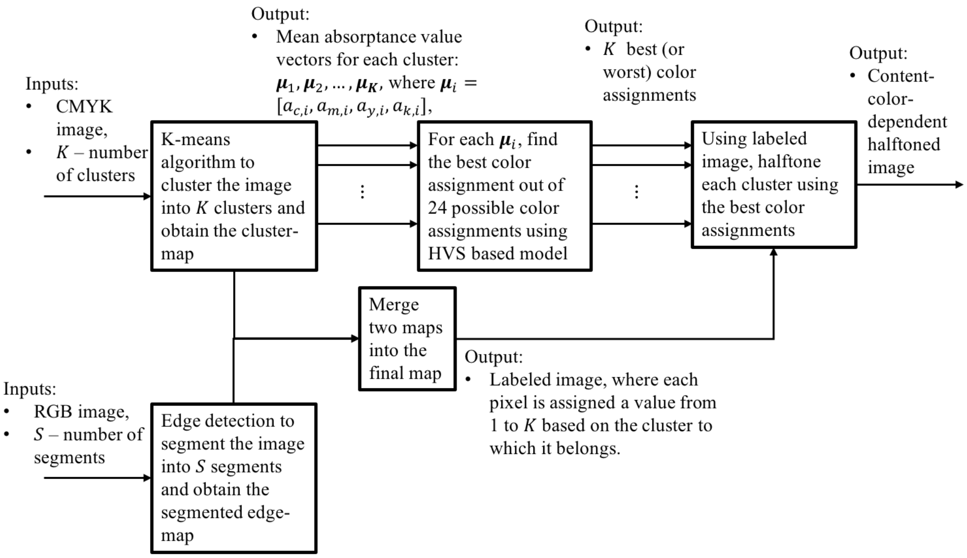

Inputs:
• CMYK image,
• K – number of clusters
K-means algorithm to cluster the image into K clusters and obtain the cluster-map
Output:
• Mean absorptance value vectors for each cluster: $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K$, where $\boldsymbol{\mu}_i = [a_{c,i}, a_{m,i}, a_{y,i}, a_{k,i}]$,
For each $\boldsymbol{\mu}_i$, find the best color assignment out of 24 possible color assignments using HVS based model
Output:
• K best (or worst) color assignments
Using labeled image, halftone each cluster using the best color assignments
Output:
• Content-color-dependent halftoned image
Merge two maps into the final map
Output:
• Labeled image, where each pixel is assigned a value from 1 to K based on the cluster to which it belongs.
Inputs:
• RGB image,
• S – number of segments
Edge detection to segment the image into S segments and obtain the segmented edge-map

Fig. 4.1.: Block diagram of content-color-dependent screening (CCDS).

on a certain similarity feature [71]. In our case, we start with a CMYK image, and our goal is to cluster all pixels in the image based on the absorptance values of the C, M, Y and K separations.

The number of clusters is usually represented by the variable K. The algorithm starts with initial estimates for the K centroids. In our approach, the initial centroids were randomly selected from the image. Each centroid represents one of the clusters. Each pixel of the image is then assigned to its nearest centroid's cluster, based on the squared Euclidean distance. Next, by taking the mean of all pixels assigned to that centroid's cluster, the new centroids are obtained. The algorithm iterates until the maximum number of iterations is reached [71]. Based on our experiments, K-means clustering converged before 10 iterations. Therefore, we chose to use the value of 10 as the maximum number of iterations.

The outputs of the K-means clustering algorithm are: a) the final centroids of the K clusters; b) the cluster-map, which is the image indexed with values 1 through K that represent each pixel being assigned to a single cluster. We'll later use the final centroids values in order to obtain the best color assignments out of 24 possible color assignments (i.e. given 4 periodicity matrices, and 4 colorants C, M, Y and K, there are 24 ways to make color assignments). In addition, we will use the cluster-map along with the edge-map in order to build the final map for halftoning the image.

4.2.2 Generation of the segmented edge-map

After obtaining the cluster-map and halftoning the image based solely on the cluster map, it was concluded that the transition between two color assignments in smooth areas of the image was very visible. Hence, we need an additional step in which the smooth areas will be identified. In order to do that, it was decided to first generate an edge map of the image, and then use the connected components algorithm to partition the edge image into segments. Since the number of segments may be too large, it was decided that we should only focus on the largest segments and constrain the number of segments to some number S. Therefore, $S-1$ segments will be selected in the order of their decreasing size. The remaining segments will be combined in the last segment. The complete block diagram for obtaining the segmented image is provided in Fig. 4.3.

As shown in Fig. 4.3, after converting from RGB to L*a*b* space, we used a bilateral filter in order to smooth the image while preserving large-scale edges without blurring. The expression for the bilateral filter with the CIE L*a*b* color difference model is presented in (4.1) [72], [73].

$$\begin{aligned}
&\mathcal{BF}\{I_k[m_0,n_0]\} = \\
&\quad \frac{1}{M}\sum_{m=m_0-w}^{m_0+w}\sum_{n=n_0-w}^{n_0+w}\exp\left(-\frac{(m-m_0)^2+(n-n_0)^2}{2\sigma_d^2}\right) \\
&\quad \times \exp\left(-\frac{\Delta E^2(I_{L^\star a^\star b^\star}[m,n], I_{L^\star a^\star b^\star}[m_0,n_0])}{2\sigma_r^2}\right) I_k[m,n],
\end{aligned} \tag{4.1}$$

where $\mathcal{BF}\{I\}$ is the bilateral filtered image in CIE L*a*b* color space; $k \in \{L^\star, a^\star, b^\star\}$ refers to one of the channels in the CIE L*a*b* color space; $[m_0, n_0]$ is the center pixel of a $(2w+1)\times(2w+1)$ convolution window; σ_d is the standard deviation of spatial smoothing; and σ_r indicates the range of tolerance in color difference. We used $\sigma_r = 6$, and for σ_d we used 2% of image diagonal. The color difference component is calculated as the Euclidean distance between the two colors in the CIE L*a*b* space:

$$\begin{aligned}
&\Delta E^2(I_{L^\star a^\star b^\star}[m,n], I_{L^\star a^\star b^\star}[m_0,n_0]) \\
&\quad = (I_{L^\star}[m,n] - I_{L^\star}[m_0,n_0])^2 + \\
&\quad (I_{a^\star}[m,n] - I_{a^\star}[m_0,n_0])^2 + \\
&\quad (I_{b^\star}[m,n] - I_{b^\star}[m_0,n_0])^2 \,,
\end{aligned} \tag{4.2}$$

and the normalization factor M is computed as

$$\begin{aligned}
M = &\sum_{m=m_0-w}^{m_0+w} \sum_{n=n_0-w}^{n_0+w} \exp\left(-\frac{(m-m_0)^2+(n-n_0)^2}{2\sigma_d^2}\right) \\
&\times \exp\left(-\frac{\Delta E^2(I_{L^\star a^\star b^\star}[m,n], I_{L^\star a^\star b^\star}[m_0,n_0])}{2\sigma_r^2}\right) I_k[m,n].
\end{aligned} \tag{4.3}$$

Next, we used a Sobel edge detector in the L*a*b* space to obtain the magnitude of the gradients in L*, a*, and b* channels, denoted as $|\nabla g_{L^\star}|$, $|\nabla g_{a^\star}|$ and $|\nabla g_{b^\star}|$ [74], [75]. We used the following Sobel operator for the horizontal direction

$$S_H = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \tag{4.4}$$

and the following Sobel operator for the vertical direction

$$S_V = S'_H. \tag{4.5}$$

In order to compute each of the three magnitudes of the gradients, we first convolved the images in L*, a*, and b* channels with S_H and S_V, to obtain the gradients in horizontal and vertical directions, and then computed their magnitudes. The magnitude of the color gradient is then computed as

$$|\nabla g_{L^\star a^\star b^\star}| = \sqrt{|\nabla g_{L^\star}|^2 + |\nabla g_{a^\star}|^2 + |\nabla g_{b^\star}|^2}. \tag{4.6}$$

The initial edge map can then be obtained by applying hysteresis thresholding [76]. The hysteresis thresholding algorithm uses two thresholds T_{high} and T_{low}. A pixel is

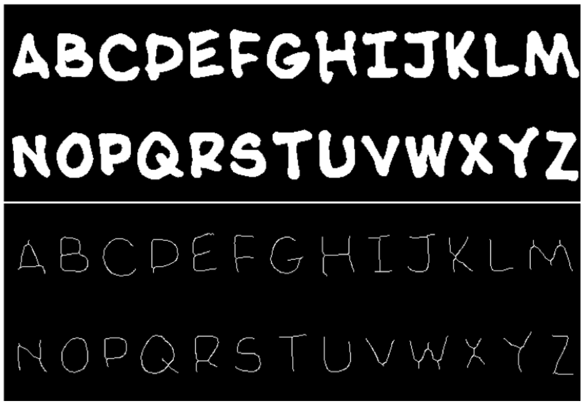


Fig. 4.2.: An example of applying Zhang-Suen thinning algorithm.

called a strong pixel if $|\nabla g_{L^\star a^\star b^\star}|[m, n] > T_{high}$ at that pixel. A pixel is called a weak pixel if $|\nabla g_{L^\star a^\star b^\star}|[m, n] \leq T_{low}$ at that pixel. All other pixels are called candidate pixels. We chose T_{high} and T_{low} empirically. In order to obtain the initial edge map, the steps below need to be iterated until there are no more changes:

1. Discard the pixel if it is weak, output the pixel if it is strong.

2. If the candidate pixel is connected to a strong pixel through the 8-point connectivity, output this candidate pixel; otherwise, do not output the candidate pixel.

In order to thin the edges, we used the well-known Zhang-Suen thinning algorithm [77]. The goal of thinning algorithms is to take a binary image and draw a 1 pixel wide skeleton of that image while retaining the shape and structure of the full image. The Zhang-Suen Thinning algorithm is probably the most used thinning algorithm. Devised in 1984, the algorithm is what is called a 2-pass algorithm, meaning that for each iteration it performs two sets of checks to remove pixels from the image. An example of thinning with Zhang-Suen thinning algorithm is shown in Fig. 4.2.

After thinning the edges, the connected components algorithm with a 4-point connectivity was used [78]. Finally, S segments were selected based on the order of decreasing size producing the segmented edge-map.

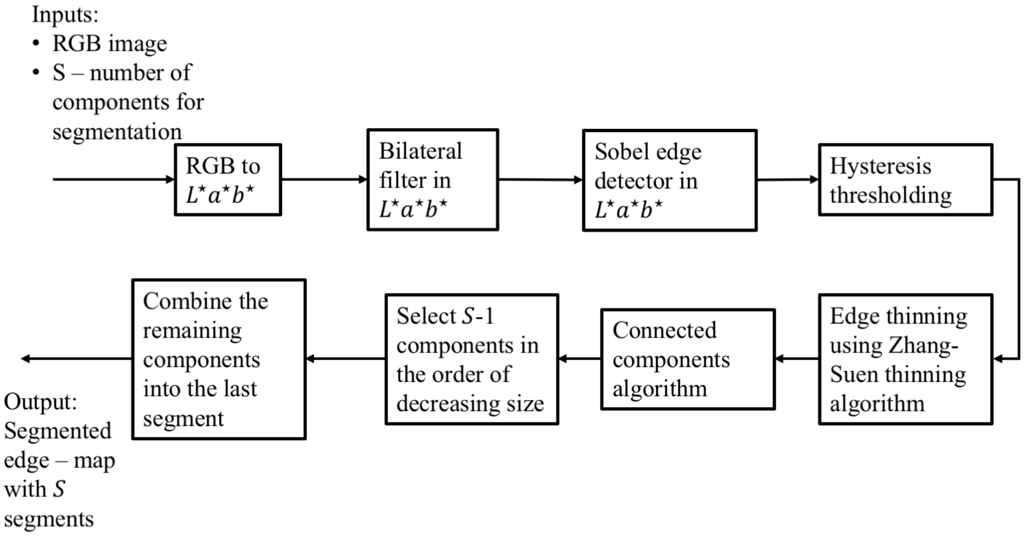

Inputs:
• RGB image
• S – number of components for segmentation
RGB to $L^\star a^\star b^\star$
Bilateral filter in $L^\star a^\star b^\star$
Sobel edge detector in $L^\star a^\star b^\star$
Hysteresis thresholding
Edge thinning using Zhang-Suen thinning algorithm
Connected components algorithm
Select S-1 components in the order of decreasing size
Combine the remaining components into the last segment
Output: Segmented edge – map with S segments

Fig. 4.3.: Block diagram for obtaining segmented edge-map.

4.2.3 Merging the cluster-map and the segmented edge-map

After obtaining the cluster-map and the segmented edge-map, the final map needs to be generated. In order to accomplish this goal, the following approach was used. We start with the segmented edge-map, and for each segment $s \in 1, 2, ..., S$, compute the number of pixels that were assigned to each cluster $k \in 1, 2, ..., K$ within this segment. We then determine the cluster number, which occurred the maximum number of times among the pixels in that segment. Finally, we assign that number to all the pixels in the segment. As a result, we obtained the final map with K clusters that would be used for halftoning the image with different color assignments.

4.2.4 Selection of the best color assignments

In order to select the best color assignment for any set of absorptance values $a_{cmyk} = (a_c, a_m, a_y, a_k)$, the HVS-based model for the superposition of color halftones was used [66]. In our case, we've narrowed down all image absorptance values to the mean absorptance values of the K clusters obtained in Sec. 4.2.1, specified as $\mu_{\mathbf{1}}, ..., \mu_{\mathbf{k}}$, where each $\mu_{\mathbf{i}} = (a_{c,i}, a_{m,i}, a_{y,i}, a_{k,i})$. Hence, for each of the K vectors of mean absorptance values, the best color assignment was obtained. The metric for obtaining the best color assignment was presented in [66] , and is denoted as $\Delta E_{fluctuation}$ (previously $\Delta E_{average}$) .

Next, two examples comparing different color assignments are presented. For both examples, the following geometries were used:

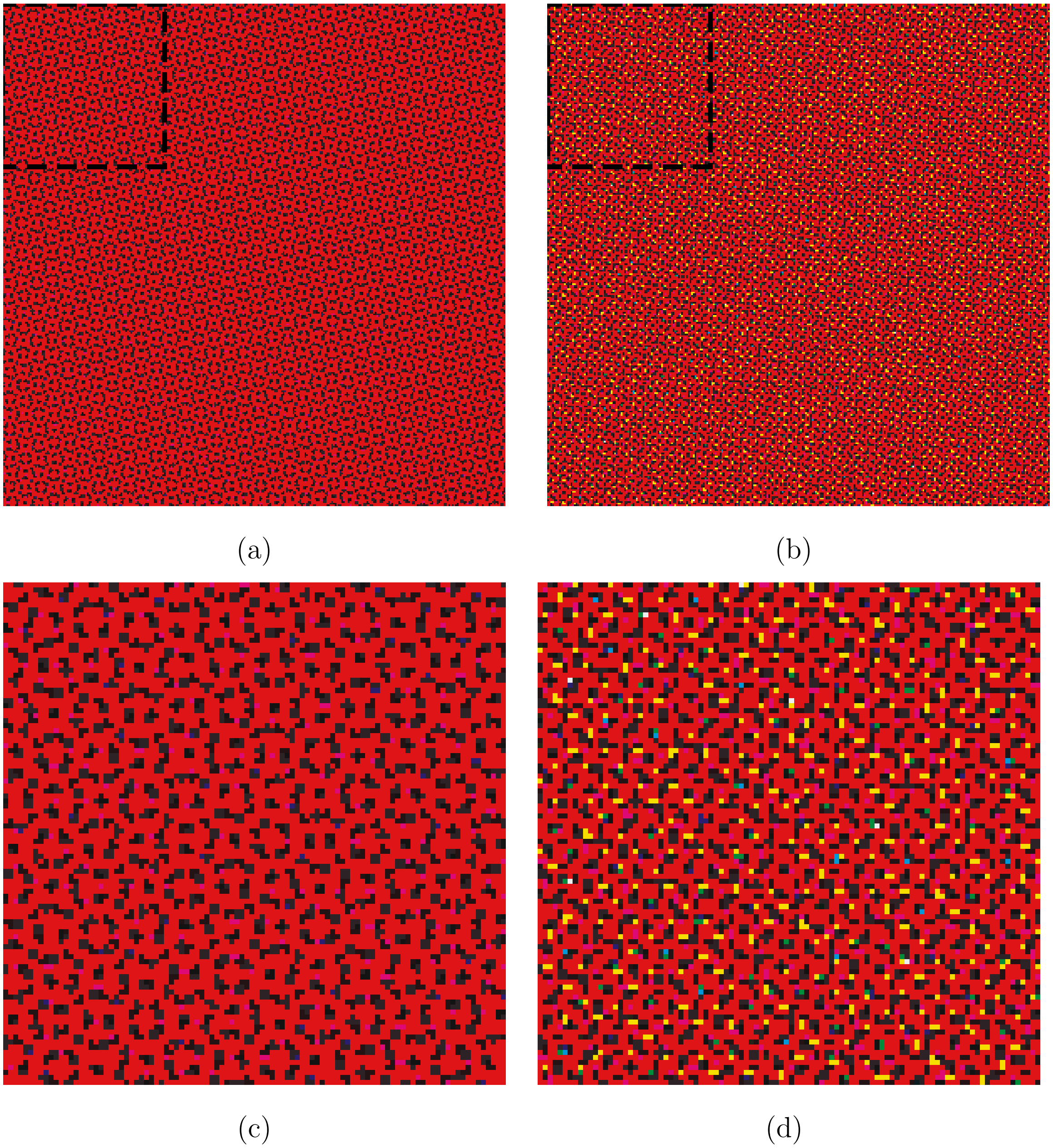
$$\mathbf{N_1} = \begin{bmatrix} 4.56 & -1.19 \\ 1.19 & 4.56 \end{bmatrix}, \mathbf{N_2} = \begin{bmatrix} 3.44 & -3.26 \\ 3.26 & 3.44 \end{bmatrix} \tag{4.7}$$
$$\mathbf{N_3} = \begin{bmatrix} 1.30 & -4.45 \\ 4.45 & 1.30 \end{bmatrix}, \mathbf{N_4} = \begin{bmatrix} 2.50 & -2.40 \\ 2.40 & 2.50 \end{bmatrix}.$$

For the HP Indigo press with resolution 812.8 dpi, the parameters for the 4 geometries above are: 172.5 lpi and 14.62°, 171.56 lpi and 43.45°, 175.17 lpi and 73.74°, and 234.73 lpi and 43.83°. In order to represent the color assignment, a 4-digit number is used. Each digit in a color assignment number represents the periodicity matrix number being assigned to C, M, Y, and K. For example, color assignment number 3214 should be interpreted as $\mathbf{N_3}$ is cyan, $\mathbf{N_2}$ is magenta, $\mathbf{N_1}$ is yellow, and $\mathbf{N_4}$ is black.

For Example 1 in Fig. 4.4, the set of absorptance values is $a_{cmyk} = (0.20, 0.93, 0.96, 0.13)$ with the color assignments 3421 and 3214, and the resulting $\Delta E_{fluctuation}$ values are 0.92 and 3.89, accordingly. It can be seen that the superposition image with color assignment 3421 is much smoother than the superposition image with color assignment 3214, and hence its $\Delta E_{fluctuation}$ is smaller. Similarly, a second example is demonstrated with superposition images in Fig. 4.5. The set of absorptance values for these images is $a_{cmyk} = (0.29, 0.31, 0.30, 0.02)$, and the color assignments are 3412 and 4231. The resulting $\Delta E_{fluctuation}$ values are 4.78 and 6.91, respectively.

4.3 Experimental results

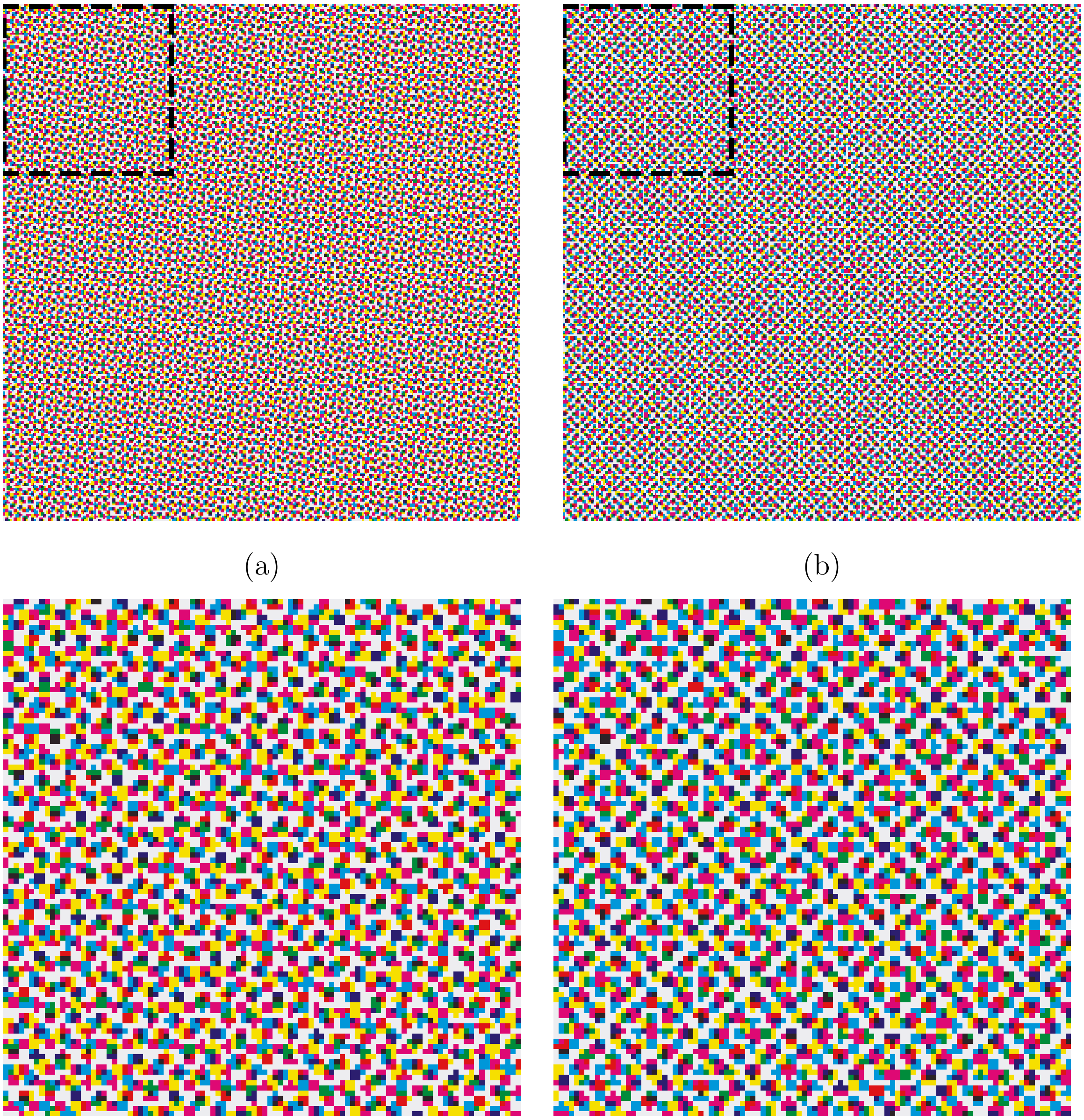
In order to demonstrate a result of CCDS, it was decided to use the image shown in Fig. 4.6 (a). The K-means clustering algorithm was applied with the number of clusters $K = 4$. The cluster-map can be observed in Fig. 4.6 (b). Based on the cluster-map, it was concluded that since smooth areas of the image, such as the woman's face and arms, got clustered into two clusters, the additional edge-detection and segmentation step was important. Otherwise, we will see artifacts from the transition between two color assignments after halftoning. The result of the segmented edge-map with $S = 4$ is presented in Fig. 4.6 (c). After that, the final map, which involves merging the maps in Fig. 4.6 (b) and (c), was obtained and is displayed in Fig. 4.6 (d). It can be seen that each smooth area of the image is assigned a single cluster. Hence, after halftoning, the visible artifacts in those areas will no longer be present. After that, we used our HVS-based model to determine the color assignments that will minimize the perceived error for the four clusters [66]. The best color assignments with their corresponding clusters are presented in Fig. 4.7. Finally, using the geometries presented in Sec. 4.2.4, a part of the halftoned image is presented in Fig. 4.8 (a). The image in Fig. 4.8 (b) is generated by halftoning using the single color assignment of 3214 for the entire image. The assignment of 3214 was chosen randomly out of 24 possible color assignments. By comparing the images in Figs. 4.8 (a) and (b), it can be concluded that the image in Fig. 4.8 (a) is much smoother than the image in Fig. 4.8 (b). Therefore, applying content-color-dependent screening yields much smoother images than the images obtained by using a single color assignment for the entire image.



(a) (b)

(c) (d)

Fig. 4.4.: Example 1: $a_{cmyk} = (0.20, 0.93, 0.96, 0.13)$ (a) color assignment 3421, $\Delta E_{fluctuation} = 0.92$; (b) color assignment 3214, $\Delta E_{fluctuation} = 3.89$; (c) zoomed-in part of the image outlined in (a); (d) zoomed-in part of the image outlined in (b)



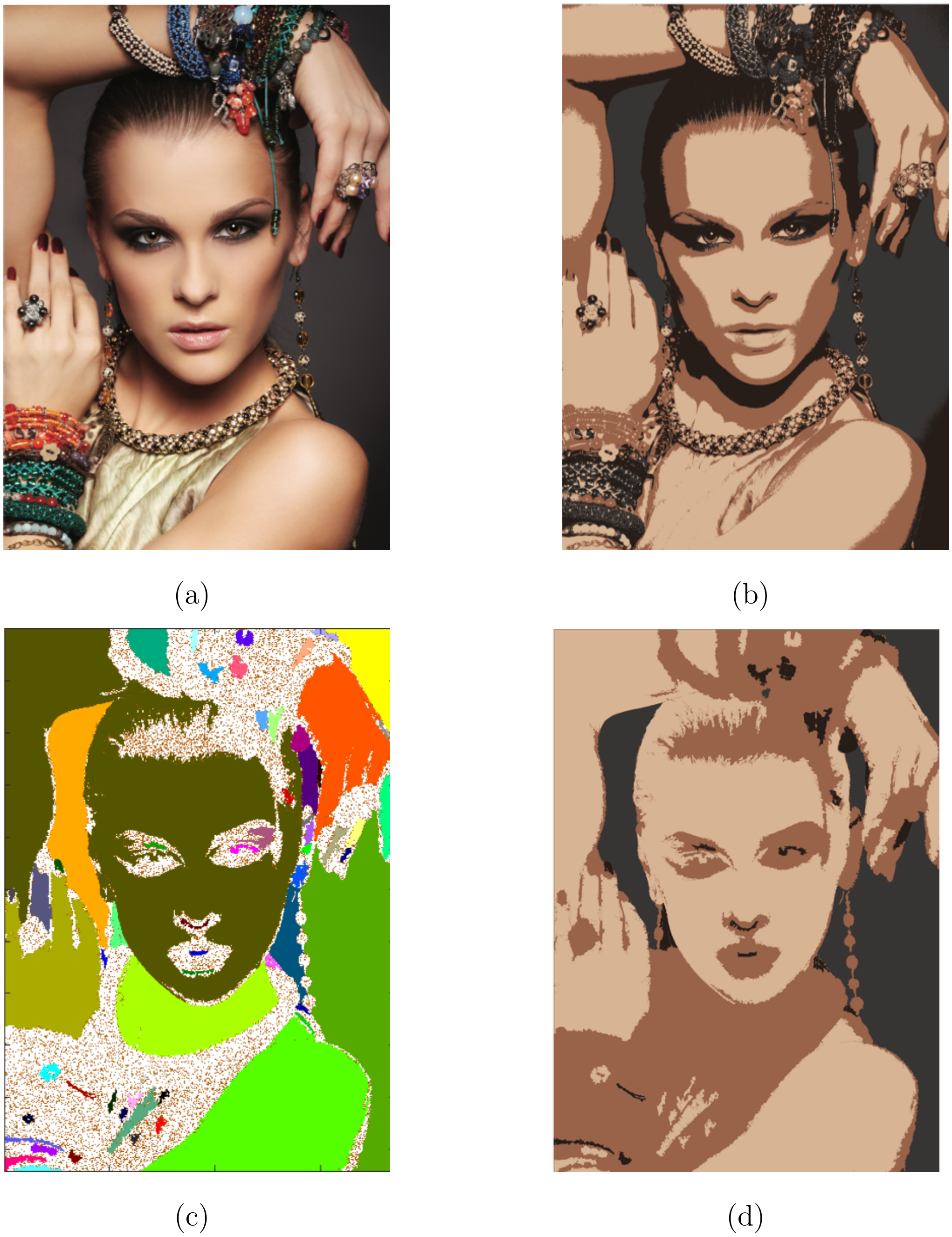
(a)

(b)

(a)

(b)

Fig. 4.5.: Example 2: $a_{cmyk} = (0.29, 0.31, 0.30, 0.02)$ (a) color assignment 3412, $\Delta E_{fluctuation} = 4.78$; (b) color assignment 4231, $\Delta E_{fluctuation} = 6.91$; (c) zoomed-in part of the image outlined in (a); (d) zoomed-in part of the image outlined in (b)



(a)

(b)

(c)

(d)

Fig. 4.6.: Example of obtaining the map for selecting best color assignments in a given image: (a) Original image; (b) Cluster-map; (c) Segmented edge-map; d) Final map: merging of cluster-map and segmented edge-map.

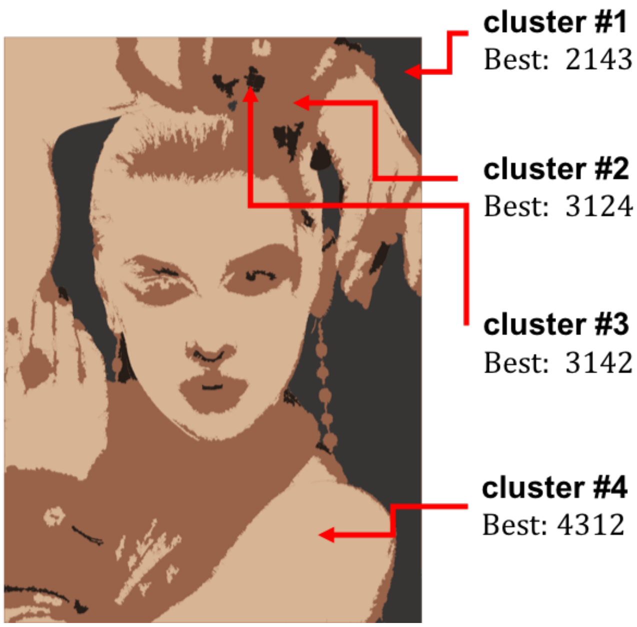
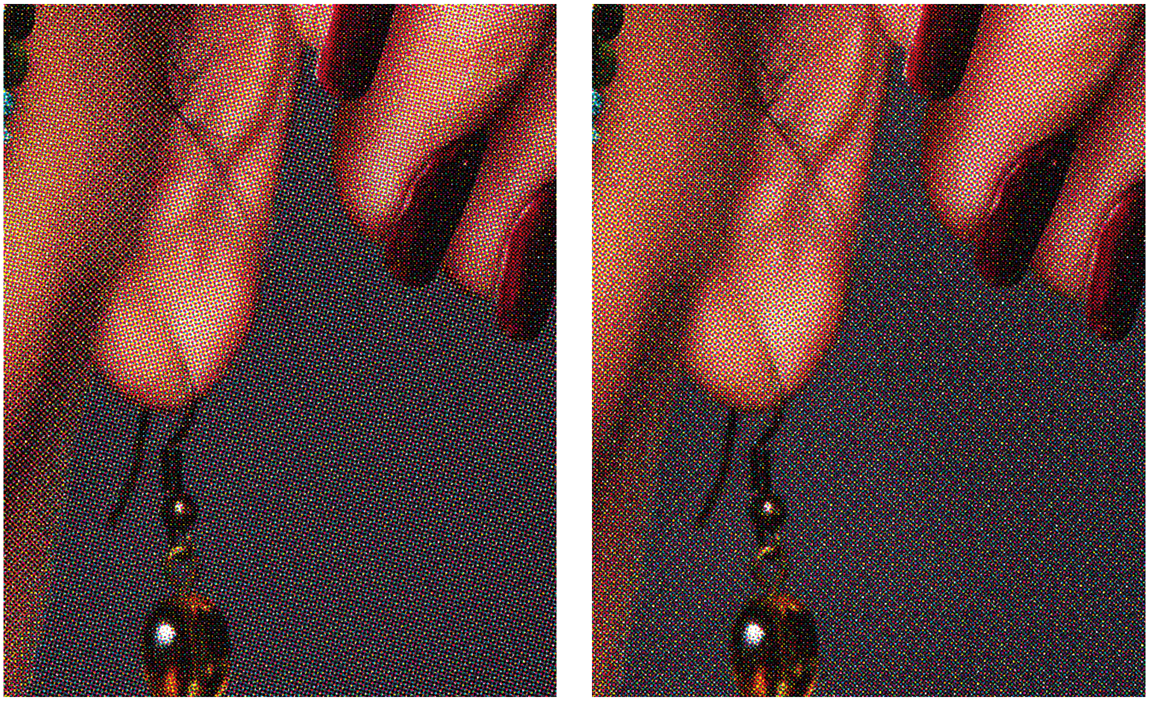


Fig. 4.7.: Best color assignments for the four clusters comprising the image.

4.4 Conclusion

We have presented a content-color-dependent screening method using clustered-dot color halftones, which helps us produce prints with better quality. We used the K-means algorithm along with edge detection to segment an image depending on its color content. Then, we used an HVS-based model to select the best color assignment for each of the clusters in the image. Since the HVS-based model determines the color assignments that will minimize the perceived error, and the entire image will be halftoned with the best color assignments based on the color content, we believe that the CCDS approach can move the quality of color prints generated by limited-resolution digital presses closer to that of the much higher resolution analog offset printing presses with which the digital presses are competing.



(a) (b)

Fig. 4.8.: (a) Halftoning with CCDS applied: optimal color assignments were used for the clusters shown in Fig. 4.7; (b) Halftoning with the single color assignment 3214 for the entire image. (The reader is advised to zoom into to 300% magnification in order to obtain a more accurate impression of these two halftone images.)

5. DESIGN OF STANDARD AND HIGH RESOLUTION, PERIODIC, CLUSTERED-DOT COLOR SCREEN SETS WITH FOCUS ON IMPACT OF HYBRID SCREENING METHOD ON HIGHLIGHTS

5.1 Introduction

In chapter 3, we presented the screen design that can generate both regular or irregular clustered-dot halftones. With this screen design, the number of distinct gray levels is equal to the number of printer addressable pixels in the halftone cell (cell area) plus an additional level. If the cell area is insufficient, then we'll observe contouring artifacts when we change from level to level. In order to remove the contouring artifacts, a supercell approach is used [80]. In the supercell approach, microcells are grouped together to form supercells. The clusters in the microcells of the supercell can grow sequentially or a few at a time, which eliminates the contouring artifacts. The dot turn-on sequence was denoted as the *microcell growing sequence* and the order in which the microcell clusters grow inside the supercell was denoted as the *macrodot growing sequence* by Lee and Allebach [47]. It has become a popular practice to use a stochastic, blue noise screen for the macrodot growing sequence. However, using the macrodot growing sequence alone results in maze-like artifacts. Lin and Allebach showed that in order to remove the maze-like artifacts, the hybrid screen design can be used [47], [61].

The hybrid screen is a modified supercell approach that removes the maze-like artifacts in the highlights and shadows and preserves the advantage of the clustered-dot screen in the midtones [47]. The hybrid screen design involves the use of cores, which are small regions inside each microcell. Lee and Allebach suggested to use a core with the size of 2 × 2 pixels [47]. In order to design the overall macrodot growing sequence as well as the microcell growing sequence within the cores DBS is used [47]. By allowing the first dot in each microcell to move around within the core from microcell to microcell within the supercell, the maze-like artifact is eliminated. After the first dot in each core is turned on, that sequence is used for the macrodot growing sequence. When every core is filled, the original microcell growing sequence is used. That way the periodic, clustered-dot texture is preserved in the midtones. Due to the dot-hole complementary symmetry, the same shadow core strategy is applied in the shadows.

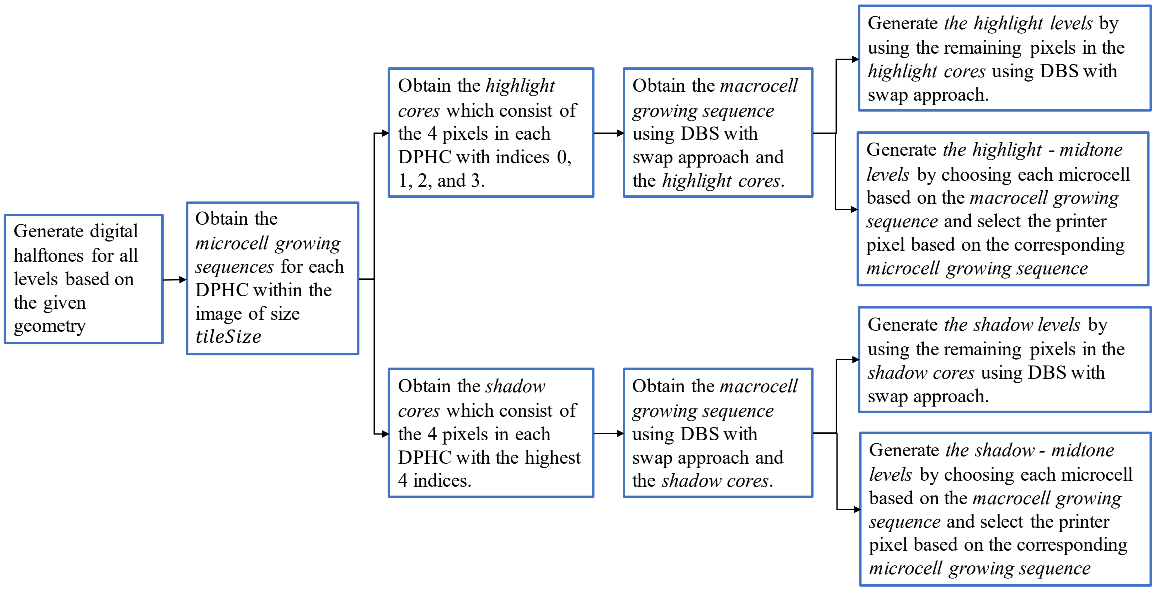
In our work, the novelty lies in applying the hybrid screen design to the irregular clustered-dot halftones, where microcell growing sequences are unique for each microcell within the tile. Therefore, cores do not have identical shapes. In addition, we extended the hybrid screen design to high-resolution periodic clustered-dot halftones, where instead of using a single pixel at the center of a microcell we use a cluster of 2×2 pixels.

5.2 Design of standard resolution periodic irregular clustered-dot halftones with the hybrid screen design

5.2.1 Procedure

We started by investigating the research work presented by Lee and Allebach in [47] and Lin and Allebach in [61]. As mentioned in the introduction, their work is based on periodic regular clustered-dot halftones. In our research, we focus on the use of periodic irregular clustered-dot halftones, which provide a larger space of available screen geometries, and therefore, we can design better screen sets with respect to smoothness and robustness. The procedure presented in this chapter can be applied to both regular or irregular halftones.

The block diagram for the algorithm is presented in Fig. 5.1. Given any geometry, we start by generating digital halftones for base levels using the screen design procedure presented in Chapter 2. The number of levels depends of the microcell



Generate digital halftones for all levels based on the given geometry
Obtain the *microcell growing sequences* for each DPHC within the image of size *tileSize*
Obtain the *highlight cores* which consist of the 4 pixels in each DPHC with indices 0, 1, 2, and 3.
Obtain the *macrocell growing sequence* using DBS with swap approach and the *highlight cores*.
Generate *the highlight levels* by using the remaining pixels in the *highlight cores* using DBS with swap approach.
Generate *the highlight - midtone levels* by choosing each microcell based on the *macrocell growing sequence* and select the printer pixel based on the corresponding *microcell growing sequence*
Obtain the *shadow cores* which consist of the 4 pixels in each DPHC with the highest 4 indices.
Obtain the *macrocell growing sequence* using DBS with swap approach and the *shadow cores*.
Generate *the shadow levels* by using the remaining pixels in the *shadow cores* using DBS with swap approach.
Generate *the shadow - midtone levels* by choosing each microcell based on the *macrocell growing sequence* and select the printer pixel based on the corresponding *microcell growing sequence*

Fig. 5.1.: Block diagram for the hybrid screen design for regular or irregular clustered dot halftones. In the figure, DPHC stands for the discrete parameter halftone cell obtained by quantizing a CPHC (continuous parameter halftone cell). The core is the region where the original microcell growing sequence is ignored and is chosen by DBS.

area of the geometry. For example, for the geometry with the screen frequency of 172.48 lpi, screen angle of 14.62°, and printer resolution of 812.8 dpi, the periodicity matrix is $\mathbf{N} = \begin{bmatrix} 4.56 & -1.19 \\ 1.19 & 4.56 \end{bmatrix}$, and the microcell area $= \det \mathbf{N} = 22.21$ printer pixels. For irregular screens, the microcell area is non-integer, hence, the number of base levels L is computed according to $L = \lceil \det \mathbf{N} \rceil + 1$. In this case, the number of base levels will be $L = 24$. The continuous-tone ramp image halftoned using this specific geometry is shown in Fig. 5.2. As mentioned in the introduction, due to the fact that the number of levels is small, we can see contouring artifacts when going from level to level. Hence, we add the hybrid screen design in order to remove the contouring artifacts.

After generating all base levels, we combine the dot turn on sequences into microcell growing sequences for each DPHC (discrete parameter halftone cells) within the image of size T. T is the smallest square that can be tiled in the vertical and horizontal direction and is denoted as *tile size*. The image containing all the microcell growing sequences is called an *index image*. An example of a part of an *index image* for the geometry with the screen frequency of 172.48 lpi, screen angle of 14.62°, and printer resolution of 812.8 dpi is shown in Fig. 5.3. For this geometry, the *tile size* $T = 112$, therefore, the full size of an *index image* is 112×112. Each value in the image corresponds to the microcell growing sequence of the corresponding pixel.

The next step is to obtain the highlight and shadow cores. The highlight cores consist of the four pixels in each DPHC with indices 0, 1, 2, and 3. The shadow cores consist of the four pixels in each DPHC with the highest four indices. Due to the fact that with irregular halftones, the number of pixels in each DPHC may vary, the shadow cores have to consist of the four pixels with the highest values. Otherwise, the number of pixels in shadow cores will vary from 1 to 4, which will perturb the symmetry of highlight and shadows. Another big difference from the regular halftones is that the shapes of the cores may not be identical. Examples of highlight and shadow

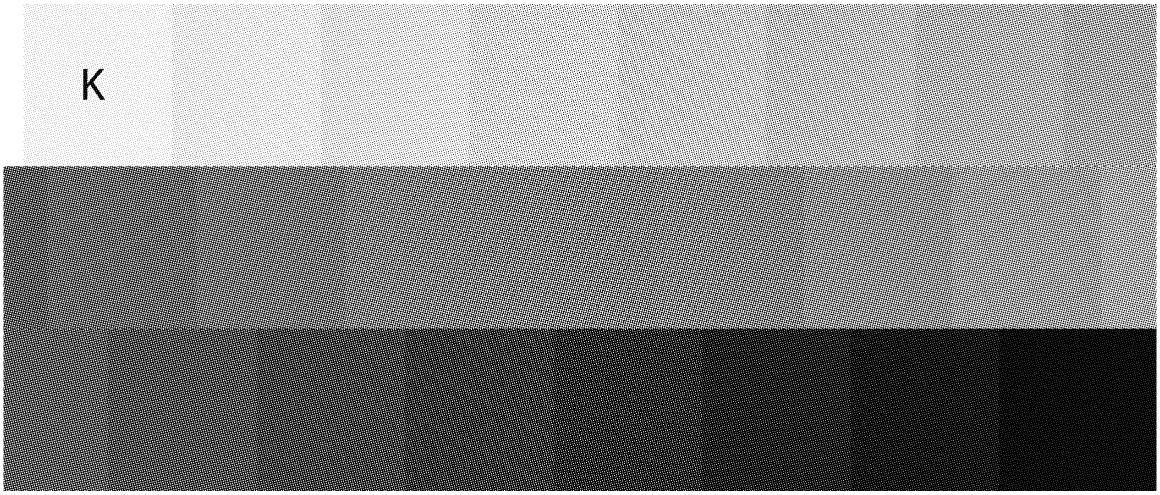


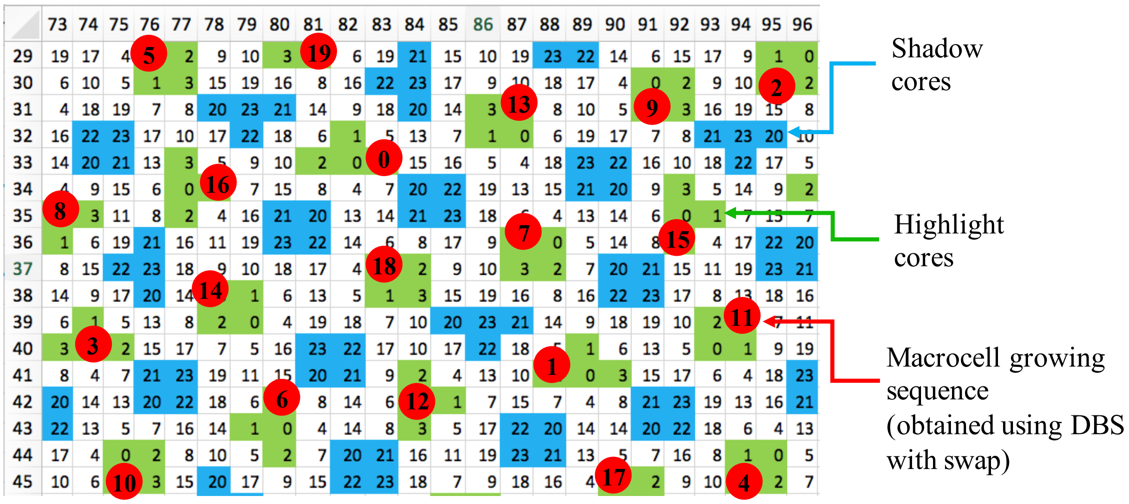
Fig. 5.2.: Example of generating all base levels for the geometry with the screen frequency of 172.48 lpi, screen angle of 14.62°, and printer resolution of 812.8. The periodicity matrix for this geometry is $\mathbf{N} = \begin{bmatrix} 4.56 & -1.19 \\ 1.19 & 4.56 \end{bmatrix}$. The number of base levels is 24.

cores are presented in Fig. 5.3. The highlight cores are highlighted in green, and the shadow cores are highlighted in blue.

Next, the *macrocell growing sequences* for both highlight cores and shadow cores can be obtained using DBS with swap. An example of the macrocell growing sequence for the highlight cores is shown in Fig. 5.3. After determining the macrocell growing sequence, one pixel in each core will have a corresponding sequence index. Hence, there will still be 3 pixels remaining in each core. In order to design the remaining highlight levels, we use macrocell growing sequence to choose each core one by one and perform DBS by toggling the remaining locations. When switching on additional dots inside the core, the 4-point connectivity needs to be preserved. Finally, we generate the highlight - midtone levels by choosing each microcell based on the macrocell growing sequence and selecting the printer pixel based on the corresponding microcell growing sequence as was done in [47]. The shadow levels and the shadow-midtone levels can be obtained in a similar way.

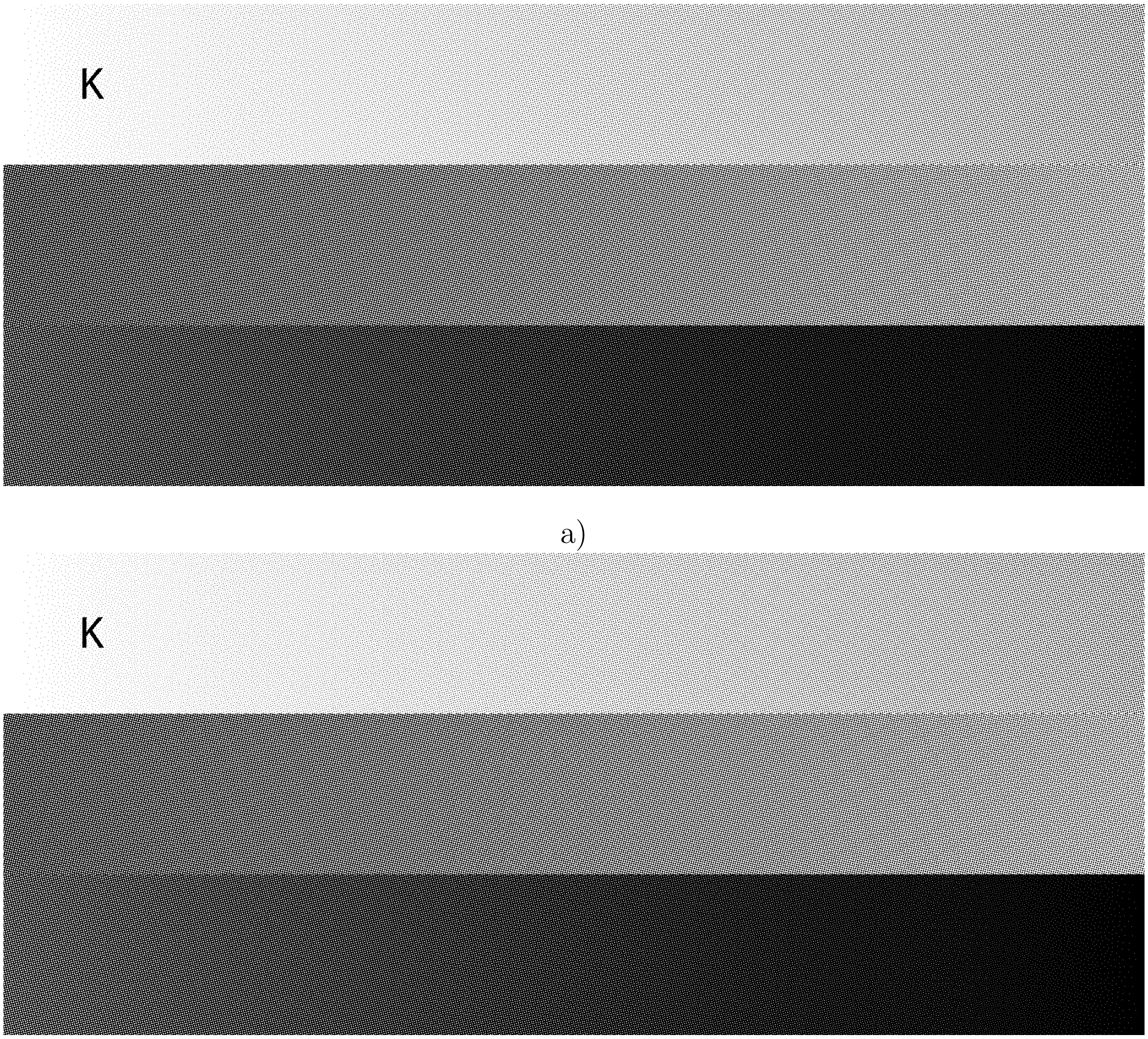
5.2.2 Experimental results

In order to demonstrate our results, we continue with an example geometry presented earlier. The geometry has the screen frequency of 172.48 lpi, screen angle of 14.62°, and printer resolution of 812.8 dpi. As is shown in Fig. 5.2, simply using the base levels creates undesirable contouring artifacts. Hence, first we'll show an example of applying the supercell approach, where the macrocell growing sequence is also obtained using DBS, but there are no cores. The resulting ramp image is shown in Fig. 5.4. It can be seen that with the supercell approach, the contouring artifacts are removed, however the maze like artifacts are present, which can be observed by zooming in to the highlights or shadows of the ramp image. Finally, we applied our updated hybrid screen design procedure for the irregular halftones and the resulting ramp image is shown in Fig. 5.4. As a result, the maze like artifacts are eliminated.



	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
29	19	17	4	**5**	2	9	10	3	**19**	6	19	21	15	10	19	23	22	14	6	15	17	9	1	0
30	6	10	5	1	3	15	19	16	8	16	22	23	17	9	10	18	17	4	0	2	9	10	**2**	2
31	4	18	19	7	8	20	23	21	14	9	18	20	14	3	**13**	8	10	5	**9**	3	16	19	15	8
32	16	22	23	17	10	17	22	18	6	1	5	13	7	1	0	6	19	17	7	8	21	23	20	10
33	14	20	21	13	3	5	9	10	2	0	**0**	15	16	5	4	18	23	22	16	10	18	22	17	5
34	4	9	15	6	0	**16**	7	15	8	4	7	20	22	19	13	15	21	20	9	3	5	14	9	2
35	**8**	3	11	8	2	4	16	21	20	13	14	21	23	18	6	4	13	14	6	0	1	7	13	7
36	1	6	19	21	16	11	19	23	22	14	6	8	17	9	**7**	0	5	14	8	**15**	4	17	22	20
37	8	15	22	23	18	9	10	18	17	4	**18**	2	9	10	3	2	7	20	21	15	11	19	23	21
38	14	9	17	20	14	**14**	1	6	13	5	1	3	15	19	16	8	16	22	23	17	8	13	18	16
39	6	1	5	13	8	2	0	4	19	18	7	10	20	23	21	14	9	18	19	10	2	**11**	7	11
40	3	**3**	2	15	17	7	5	16	23	22	17	10	17	22	18	5	1	6	13	5	0	1	9	19
41	8	4	7	21	23	19	11	15	20	21	9	2	4	13	10	**1**	0	3	15	17	6	4	18	23
42	20	14	13	20	22	18	6	**6**	8	14	6	**12**	1	7	15	7	4	8	21	23	19	13	16	21
43	22	13	5	7	16	14	1	0	4	14	8	3	5	17	22	20	14	14	20	22	18	6	4	13
44	17	4	0	2	8	10	5	2	7	20	21	16	11	19	23	21	13	5	7	16	8	1	0	5
45	10	6	**10**	3	15	20	17	9	15	22	23	18	7	9	18	16	4	**17**	2	9	10	**4**	2	7

Fig. 5.3.: An example of a part of an *index image* for the geometry with the screen frequency of 172.48 lpi, screen angle of 14.62°, and printer resolution of 812.8 dpi. For this geometry, the *tile size* $T = 112$, therefore, the full size of an *index image* is 112×112. Each value in the image corresponds to the microcell growing sequence of the corresponding pixel. The highlight cores are highlighted in green, and the shadow cores are highlighted in blue. An example of the macrocell growing sequence for the highlight cores is shown in red.



a)

b)

Fig. 5.4.: Example of applying the supercell and the hybrid screen design for standard resolution, periodic, irregular, clustered-dot halftones using the geometry with the screen frequency of 172.48 lpi, screen angle of 14.62°, and printer resolution of 812.8: a) Ramp image after the supercell approach; b) Ramp image after the hybrid approach.

5.3 Design of high resolution, periodic, irregular, clustered-dot halftones with the hybrid screen design

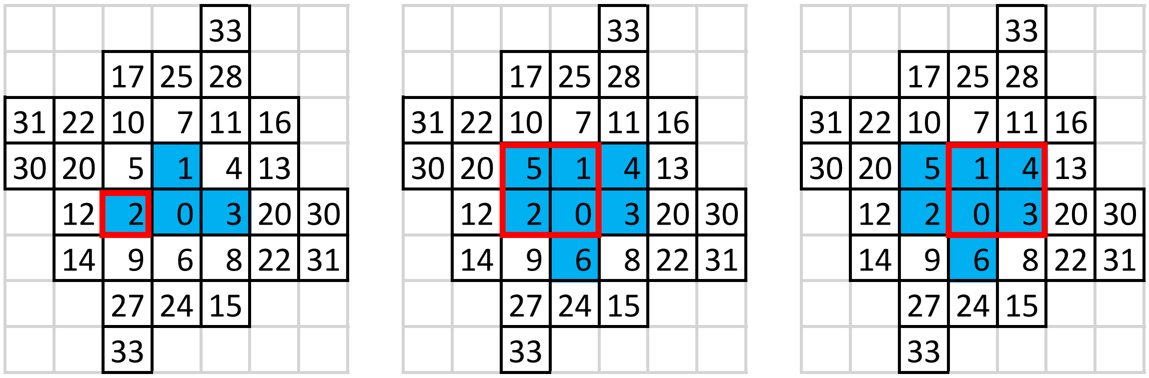
5.3.1 Update to the hybrid screen design used for standard resolution, periodic, clustered-dot halftoning

With high resolution printing, the main difference is that instead of having 1 pixel as the center of a microcell, we want to have a cluster of 2×2 pixels. Clusters of 2×2 pixels are used in electrophotographic printers due to its better print stability in comparison to 1 pixel. That's why the hybrid screen design needs to be updated.

The main update involves increasing the size of the core in order to incorporate the shifting of a 2×2 cluster inside the core. Instead of having a core of 4 pixels in the hybrid screen design, we need to have a core of at least 7 pixels so that the 2×2 block can be shifted inside the region with 7 pixels. An example demonstrating this update is shown in Fig. 5.5.

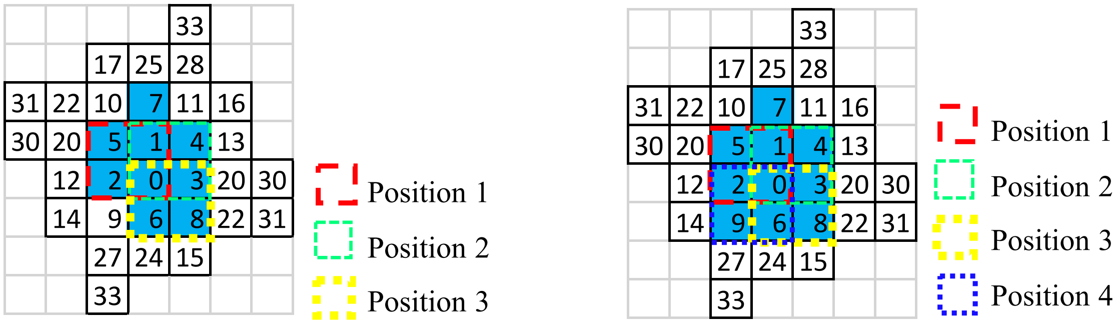
By increasing the core size, the number of possible shifting positions may increase. Figure 5.6 has an example of a microcell whose cores consist of 9 and 10 pixels. In the case of 9 pixels, there are 3 possible shifting positions, and in the case of 10 pixels, there are 4 possible shifting positions. By increasing the core size, the quality of the halftones maybe improved, however it will require more computations.

Similarly to the hybrid screen design for standard resolution periodic clustered-dot halftoning, in order to determine the optimal position of a 2×2 cluster of pixels, DBS with swap is used. After determining the optimal position for a 2×2 cluster of pixels, the remaining indices in the core are chosen using DBS with toggle. Due to the fact that we are swapping clusters of 2×2 pixels and not single pixels, DBS equations were updated accordingly. Chapter 6 shows derivations of the equations for swapping and toggling clusters of 2×2 pixels.



a) b) c)

Fig. 5.5.: An example of a microcell with cores for standard resolution hybrid screen design a) and high resolution hybrid screen design b) and c). A core is highlighted in blue. For standard resolution hybrid screen design, a core consists of 4 pixels with indices $0, ..., 3$, as it is shown in a). Since, a shifting unit is a single pixel, there are 4 possible positions for a pixel to move inside the core. An example of one shifting position is shown in a red box. For the high resolution hybrid screen design, a core should consist of at least 7 pixels (indices from 0 to 6) so that there are at least two possible shifting positions for a cluster of 2×2 pixels: b) and c) show the two possible shifting positions in red boxes.



a)

b)

Fig. 5.6.: An example of a microcell whose cores consist of 9 and 10 pixels: a) The core (highlighted in blue) consists of 9 pixels with indices from 0 to 8 so that there are 3 possible shifting positions for a cluster of 2 × 2 pixels; b) The core consists of 10 pixels with indices from 0 to 9 so that there are 4 possible shifting positions for a cluster of 2 × 2 pixels.

5.3.2 Experimental results

In order to demonstrate our results, we used a geometry with the screen frequency of 288.17 lpi, screen angle of 29.58°, and printer resolution of 1625.6 dpi. For this example, we used a core size of 8 pixels. The halftoned ramp image is shown in Fig. 5.7. By zooming into the highlights and the shadows of the ramp it can be seen that the dot and hole clusters consist of 2×2 pixels instead of single pixels, which will improve the quality of the prints for high resolution printing.

5.4 Conclusion

In this chapter, we presented the design of standard and high resolution, periodic, clustered-dot regular or irregular color screen sets with focus on impact of hybrid screening method on highlights. We started by designing base levels given geometry parameters: screen frequency, screen angle and printer resolution. Since the primary focus of our research was periodic irregular clustered-dot halftones, we extended the hybrid screen design presented by Lee and Allebach [47] to irregular halftones. We then compared the use of the supercell approach with the use of the hybrid screen design, and showed that the hybrid screen design removes the undesirable maze-like artifacts. Finally, we extended the hybrid screen design for high resolution printing, where the center of the microcell consists of a cluster of 2×2 pixels instead of a single pixel. It can be seen that the hybrid screen design for high resolution, periodic, clustered-dot halftones produces smooth halftone prints, while preserving clusters of 2×2 pixels in highlights and shadows.

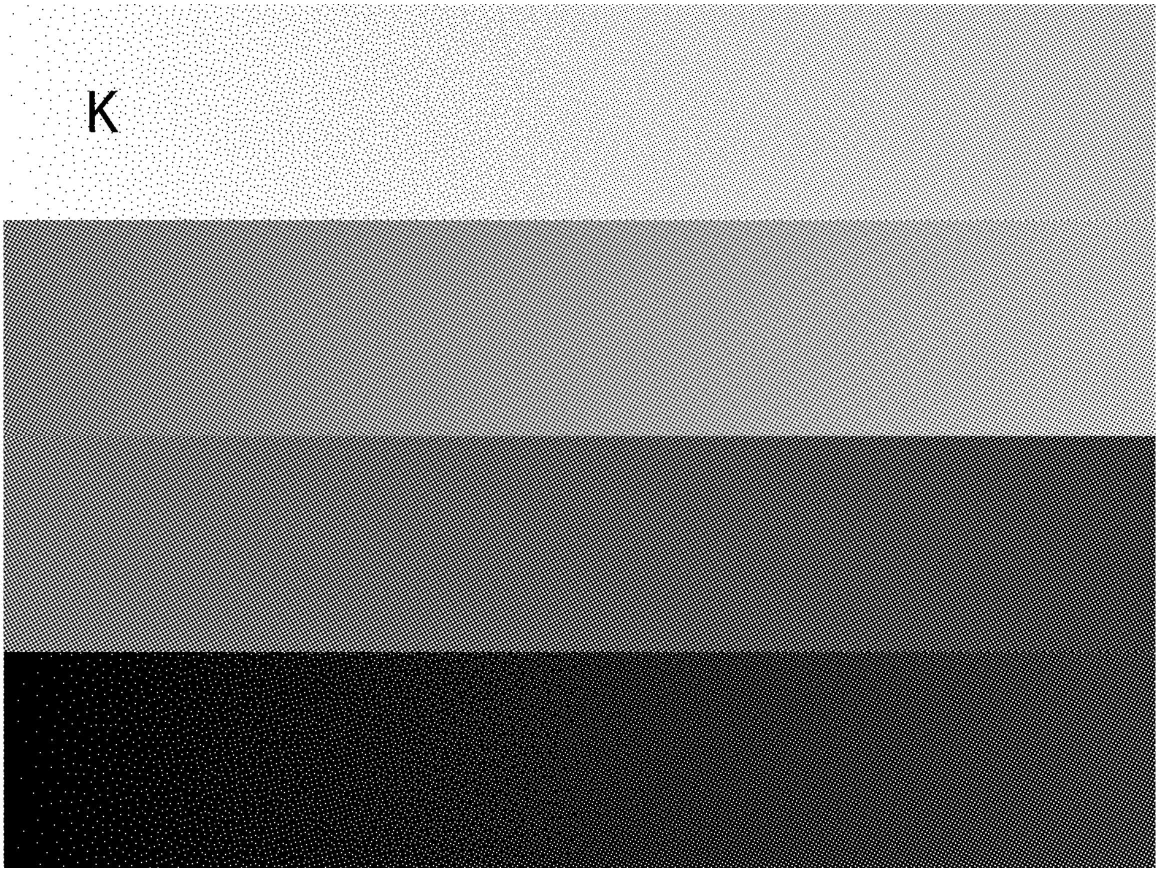


Fig. 5.7.: Example of applying the hybrid screen design for high resolution, periodic, irregular, clustered-dot halftones using the geometry with the screen frequency of 288.17 lpi, screen angle of 29.58°, and printer resolution of 1625.6 dpi. In this case, the core size is 8 pixels.

6. MONOCHROME DBS-BASED SCREEN DESIGN WITH OVERLAPPING CLUSTERS OF 2×2 PIXELS OR 3×3 PIXELS

6.1 Introduction

Digital Halftoning is the process of rendering a continuous tone image with a limited number of tone levels. The goal of digital halftoning is to generate an image with correct tone and detail of original image without introducing any visible artifacts [47]. There are three most widely used algorithms: point processes (screening or dithering), neighborhood processes (error diffusion), and iterative algorithms [47], [45], [46]. Screening is computationally most efficient among the three methods because it requires only one comparison per pixel with the threshold matrix. The design of the threshold matrix can be complex and computationally expensive, however, it is a one-time and an offline process.

Conventional halftoning is accomplished by either changing the relative density of printed dots or changing the size of printed dots on the page [45], [46]. These two approaches are dispersed-dot halftoning and clustered-dot halftoning. In this chapter, we focus on dispersed-dot halftoning. Dispersed-dot screens are widely used in printers that have no trouble in rendering an isolated dot. However, for electrophotographic printers, this halftoning method doesn't have a good print quality because the isolated dots are not stable. Therefore, our goal was to generate dispersed-dot screens for electrophotographic printers by using clusters of 2×2 or 3×3 pixels as the smallest printing unit.

In the next section, we introduce the procedure for the monochrome DBS-based screen design with overlapping clusters of 2×2 pixels. Next, we extend our screen design to clusters of 3×3 pixels. Finally, we show experimental results by halftoning a continuous tone ramp image. For comparison, we halftone the ramp with the monochrome DBS-based screen design with non-overlapping and overlapping clusters of 2 × 2 pixels.

6.2 Preliminaries

In this chapter, we use $(\mathbf{x}) = (x, y)$ and $(\mathbf{m}) = [m, n]$ to represent continuous and discrete spatial coordinates, respectively. The units of $(\mathbf{x})$ and $(\mathbf{m})$ are inches and pixels, respectively. We use $f[m, n]$ as a continuous-tone image with discrete spatial coordinates and $g[m, n]$ as a halftone image.

6.3 Review of the Direct Binary Search (DBS)

Direct binary search is an iterative halftoning algorithm. It minimizes the error between the perceived continuous-tone image and the perceived halftone image [84], [87]. In DBS, every pixel is visited once per iteration. For every pixel, if the pixel value is toggled or swapped with one of the neighbors, the change in the perceived error is computed. If any of these operations decrease the perceived error, then we apply the operation that has the maximum decrease of the perceived error. Image is scanned iteratively until no operations or a few operations are accepted per iteration. This would be the final halftone image. DBS is not guaranteed to achieve the global minimum, but it produces halftone images with very good quality.

The perceived error $\tilde{e}(\mathbf{x})$ between the continuous-tone image and the halftone image is given by

$$\begin{aligned} \tilde{e}(\mathbf{x}) &= \tilde{g}(\mathbf{x}) - \tilde{f}(\mathbf{x}) \\ &= \sum_{\mathbf{m}} e[\mathbf{m}]\tilde{p}(\mathbf{x} - \mathbf{m}X), \end{aligned} \tag{6.1}$$

where

$$e[\mathbf{m}] = g[\mathbf{m}] - f[\mathbf{m}], \tag{6.2}$$

and

$$\tilde{p}(\mathbf{x}) = p(\mathbf{x}) ** h(\mathbf{x}) \tag{6.3}$$

is the perceived spot profile function of the ideal printer [86]. Here $p(\mathbf{x}) = rect\left(\frac{\mathbf{x}}{X}\right)$ is the spot profile function of the ideal printer, and X is the distance between printer-addressable pixels. Thus, $X = \frac{1}{R}$, where R is the printer resolution. For the HVS model, we use the Nasanen model, which was shown to produce the best halftone quality when incorporated in DBS [69], [85]. The frequency response of the Nasanen HVS model is shown in Fig. 6.1. Finally, the mean squared error is given as

$$E = \int |\tilde{e}(\mathbf{x})| \, d\mathbf{x} = \sum_{\mathbf{m}} e[\mathbf{m}] \, c_{\tilde{p}\tilde{e}}[\mathbf{m}], \tag{6.4}$$

where $c_{\tilde{p}\tilde{e}}[\mathbf{m}] = e[\mathbf{m}] ** c_{\tilde{p}\tilde{p}}[\mathbf{m}]$.

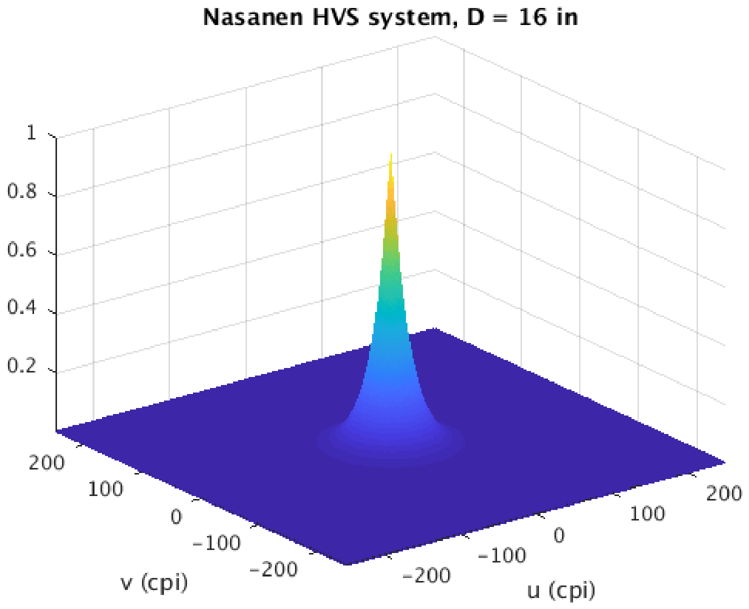


Fig. 6.1.: The frequency response of the Nasanen HVS model with the viewing distance of 16 in.

In our work, we use DBS with swap or DBS with toggle. For example, if we want to swap pixels located at $\mathbf{m}_0$ and $\mathbf{m}_1$, the swap equations are given as follows

$$\begin{aligned} g'[\mathbf{m}] &= g[\mathbf{m}] + a_0\delta[\mathbf{m} - \mathbf{m}_0] + a_1\delta[\mathbf{m} - \mathbf{m}_1] \quad (6.5) \\ \Delta E &= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_0] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_1] \\ &\quad + 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_0 - \mathbf{m}_1] \quad (6.6) \\ c'_{\tilde{p}\tilde{e}}[\mathbf{m}] &= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_0] + a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_1], \quad (6.7) \end{aligned}$$

where $a_0 = 1$ if $g[\mathbf{m}_0]$ is changed from 0 to 1, and $a_0 = -1$ if $g[\mathbf{m}_0]$ is changed from 1 to 0. Hence, $a_1 = -a_0$, since the swapping of $g[\mathbf{m}_0]$ and $g[\mathbf{m}_1]$ is performed [86]. Whereas, if we want to toggle a pixel located at $\mathbf{m}_0$, then the toggle equations are given as follows

$$\begin{aligned} g'[\mathbf{m}] &= g[\mathbf{m}] + a_0\delta[\mathbf{m} - \mathbf{m}_0] \quad (6.8) \\ \Delta E &= a_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_0] \quad (6.9) \\ c'_{\tilde{p}\tilde{e}}[\mathbf{m}] &= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_0]. \quad (6.10) \end{aligned}$$

In our work, we use the equations above and update them for toggling and swapping clusters of k pixels.

6.4 DBS equations for swapping and toggling clusters of k pixels.

Let the first cluster consist of k pixels with coordinates $(\mathbf{m}_0, \mathbf{m}_1, ..., \mathbf{m}_{k-1})$, and the second cluster consist of pixels with coordinates $(\mathbf{n}_0, \mathbf{n}_1, ..., \mathbf{n}_{k-1})$. In order to compute the ΔE and $c_{\tilde{p}\tilde{e}}[\mathbf{m}]$ update equations, we assume the swapping of k pixels step by step, and then add the updates to determine the final result.

To swap $\mathbf{m}_0$ and $\mathbf{n}_0$:

$$\begin{aligned} \Delta E_0 &= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_0] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_0] \quad (6.11) \\ &\quad + 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_0 - \mathbf{n}_0] \\ c'_{\tilde{p}\tilde{e}}[\mathbf{m}] &= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_0] + a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_0] \quad (6.12) \end{aligned}$$

To swap $\mathbf{m}_1$ and $\mathbf{n}_1$:

$$\begin{aligned}
\Delta E_1 &= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] \\
&+ 2a_0(c_{\tilde{p}\tilde{e}}[\mathbf{m}_1] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{m}_0] + a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{n}_0]) \\
&+ 2a_1(c_{\tilde{p}\tilde{e}}[\mathbf{n}_1] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_1 - \mathbf{m}_0] + a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_1 - \mathbf{n}_0]) \\
&+ 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{n}_1] \\
&= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] \\
&+ 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_1] + 2a_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{m}_0] + 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{n}_0] \\
&+ 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_1] + 2a_1a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_1 - \mathbf{m}_0] + 2a_1^2\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_1 - \mathbf{n}_0] \\
&+ 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{n}_1]
\end{aligned} \tag{6.13}$$

$$\begin{aligned}
c''_{\tilde{p}\tilde{e}}[\mathbf{m}] &= c'_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_1] + a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_1] \\
&= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0(c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_1]) \\
&+ a_1(c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_1])
\end{aligned} \tag{6.14}$$

To swap $\mathbf{m}_2$ and $\mathbf{n}_2$:

$$\begin{aligned}
\Delta E_2 &= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] \qquad (6.15)\\
&+ 2a_0(c_{\tilde{p}\tilde{e}}[\mathbf{m}_2] + a_0(c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{m}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{m}_1])\\
&+ a_1(c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_1]))\\
&+ 2a_1(c_{\tilde{p}\tilde{e}}[\mathbf{n}_2] + a_0(c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{m}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{m}_1])\\
&+ a_1(c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{n}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{n}_1]))\\
&+ 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_2]\\
&= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] \qquad (6.16)\\
&+ 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_2] + 2a_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{m}_0] + 2a_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{m}_1]\\
&+ 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_0] + 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_1]\\
&+ 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_2] + 2a_1a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{m}_0] + 2a_1a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{m}_1]\\
&+ 2a_1^2\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{n}_0] + 2a_1^2\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{n}_1]\\
&+ 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_2]\\
&= (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_2] + 2a_0^2 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{m}_i] \qquad (6.17)\\
&+ 2a_1a_0 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_i] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_2] + 2a_1a_0 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{m}_i]\\
&+ 2a_1^2 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{n}_i] + 2a_0a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_2]\\
c'''_{\tilde{p}\tilde{e}}[\mathbf{m}] &= c''_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_2] + a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_2] \qquad (6.18)\\
&= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0(c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_1] + c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_2])\\
&+ a_1(c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_0] + c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_1] + c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_2])
\end{aligned}$$

To swap $\mathbf{m}_{k-1}$ and $\mathbf{n}_{k-1}$:

$$\begin{aligned}\Delta E_{k-1} = {} & (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_{k-1}] + 2a_0^2 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{m}_{k-1} - \mathbf{m}_i] \\ & + 2a_1 a_0 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{m}_{k-1} - \mathbf{n}_i] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_{k-1}] + 2a_1 a_0 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{n}_{k-1} - \mathbf{m}_i] \\ & + 2a_1^2 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{n}_{k-1} - \mathbf{n}_i] + 2a_0 a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_{k-1} - \mathbf{n}_{k-1}]\end{aligned} \tag{6.19}$$

$$c_{\tilde{p}\tilde{e}}{}'^{k}[\mathbf{m}] = c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_i] + a_1 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_i] \tag{6.20}$$

Finally,

$$\Delta E_{update} = \Delta E_0 + \Delta E_1 + \Delta E_2 + ... + \Delta E_{k-1} \tag{6.21}$$

$$\begin{aligned} & = (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_0] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_0] \\ & + 2a_0 a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_0 - \mathbf{n}_0] + (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] \\ & + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_1] + 2a_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{m}_0] + 2a_0 a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{n}_0] \\ & + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_1] + 2a_1 a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_1 - \mathbf{m}_0] + 2a_1^2\, c_{\tilde{p}\tilde{p}}[\mathbf{n}_1 - \mathbf{n}_0] \\ & + 2a_0 a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_1 - \mathbf{n}_1] + (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_2] + 2a_0^2 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{m}_i] \\ & + 2a_1 a_0 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_i] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_2] + 2a_1 a_0 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{m}_i] \\ & + 2a_1^2 \sum_{i=0}^{1} c_{\tilde{p}\tilde{p}}[\mathbf{n}_2 - \mathbf{n}_i] + 2a_0 a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_2 - \mathbf{n}_2] \\ & + ... + (a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0\, c_{\tilde{p}\tilde{e}}[\mathbf{m}_{k-1}] + 2a_0^2 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{m}_{k-1} - \mathbf{m}_i] \\ & + 2a_1 a_0 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{m}_{k-1} - \mathbf{n}_i] + 2a_1\, c_{\tilde{p}\tilde{e}}[\mathbf{n}_{k-1}] + 2a_1 a_0 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{n}_{k-1} - \mathbf{m}_i] \\ & + 2a_1^2 \sum_{i=0}^{k-2} c_{\tilde{p}\tilde{p}}[\mathbf{n}_{k-1} - \mathbf{n}_i] + 2a_0 a_1\, c_{\tilde{p}\tilde{p}}[\mathbf{m}_{k-1} - \mathbf{n}_{k-1}]\end{aligned} \tag{6.22}$$

After adding up all the terms, the final update equation has the following form

$$\begin{aligned}\Delta E_{update} = & \; k(a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{e}}[\mathbf{m}_i] + 2a_1 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{e}}[\mathbf{n}_i] \\ & + 2a_0 a_1 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_i - \mathbf{n}_i] \\ & + 2a_0^2 \sum_{j=1}^{k-1} \sum_{i=0}^{j-1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_j - \mathbf{m}_i] + 2a_0 a_1 \sum_{j=1}^{k-1} \sum_{i=0}^{j-1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_j - \mathbf{n}_i] \\ & + 2a_1 a_0 \sum_{j=1}^{k-1} \sum_{i=0}^{j-1} c_{\tilde{p}\tilde{p}}[\mathbf{n}_j - \mathbf{m}_i] + 2a_1^2 \sum_{j=1}^{k-1} \sum_{i=0}^{j-1} c_{\tilde{p}\tilde{p}}[\mathbf{n}_j - \mathbf{n}_i]\end{aligned} \tag{6.23}$$

Similarly, in order to toggle a cluster of k pixels, the following ΔE_{update} equation should be used:

$$\Delta E_{update} = ka_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{e}}[\mathbf{m}_i] + 2a_0^2 \sum_{j=1}^{k-1} \sum_{i=0}^{j-1} \mathbf{m}_j - \mathbf{m}_i \tag{6.24}$$

6.5 Procedure for the monochrome DBS-based screen design with overlapping clusters of 2×2 pixels

In this section, we present a procedure for monochrome DBS-based screen design with overlapping clusters of 2×2 pixels. The goal of our screen design is to generate a thresholding matrix that will produce smooth halftone textures given a desired matrix size M and take into account overlapping clusters of 2×2 pixels. The block diagram of the procedure is shown in Fig. 6.2.

We start with a desired matrix size M. In our case, matrix size should be an integer multiple of 8 due to hardware limitations. First, a random halftone consisting of clusters of 2×2 pixels with 50% absorptance is generated. We start with 50% absorptance so that we can grow dots and holes in a similar manner. Next, we apply monochrome DBS algorithm for the non-overlapping clusters of 2×2 pixels and generate an initial level. In order to apply the monochrome DBS algorithm for the non-overlapping clusters of 2×2 pixels, we modified the swapping and toggling DBS equations for the case of clusters of 2 × 2 pixels. The derivation of the equations is provided in Section 6.5.1. After designing an initial level consisting of non-overlapping clusters of 2 × 2 pixels, we wanted to introduce the overlapping clusters. Hence, it was decided to grow some clusters of 2 × 2 pixels by swapping single dot pixels with single hole pixels, and creating dot and hole clusters of at least 2 × 2 pixels. The new halftone was called an optimized initial level. Next, in order to generate levels below and levels above the optimized initial level, we apply DBS with toggle and allow toggling of either single pixels, or clusters of 2 × 2 pixels, 1 × 2 or 2 × 1 pixels. Finally, all levels are combined and a thresholding matrix is generated. In the next subsections, further details of each step are provided.

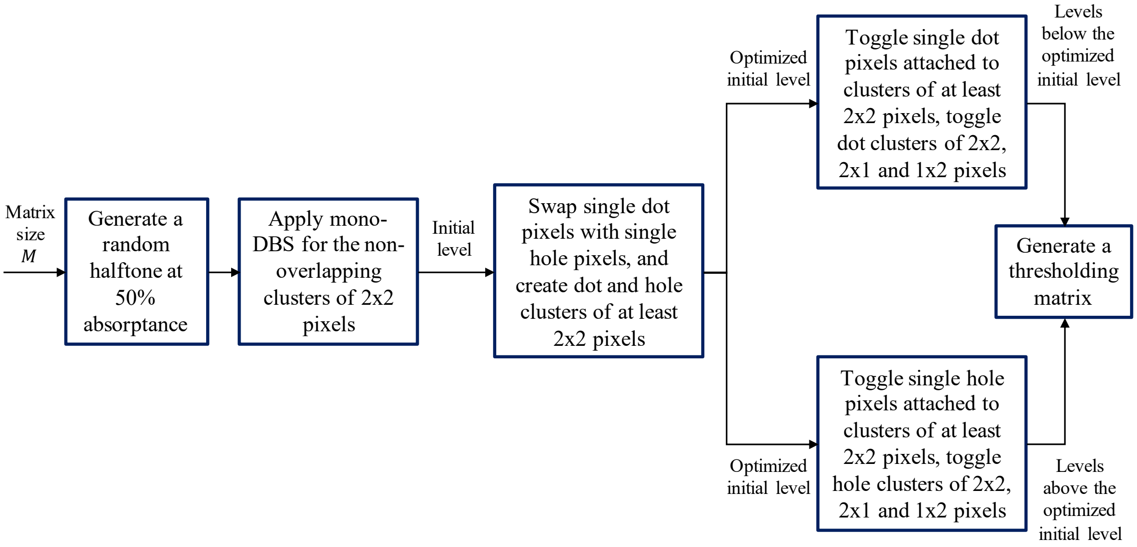

Matrix size *M*
Generate a random halftone at 50% absorptance
Apply mono-DBS for the non-overlapping clusters of 2x2 pixels
Initial level
Swap single dot pixels with single hole pixels, and create dot and hole clusters of at least 2x2 pixels
Optimized initial level
Toggle single dot pixels attached to clusters of at least 2x2 pixels, toggle dot clusters of 2x2, 2x1 and 1x2 pixels
Levels below the optimized initial level
Generate a thresholding matrix
Optimized initial level
Toggle single hole pixels attached to clusters of at least 2x2 pixels, toggle hole clusters of 2x2, 2x1 and 1x2 pixels
Levels above the optimized initial level

Fig. 6.2.: Block diagram of the monochrome DBS-based screen design with overlapping clusters of 2 × 2 pixels.

6.5.1 Generating an initial level

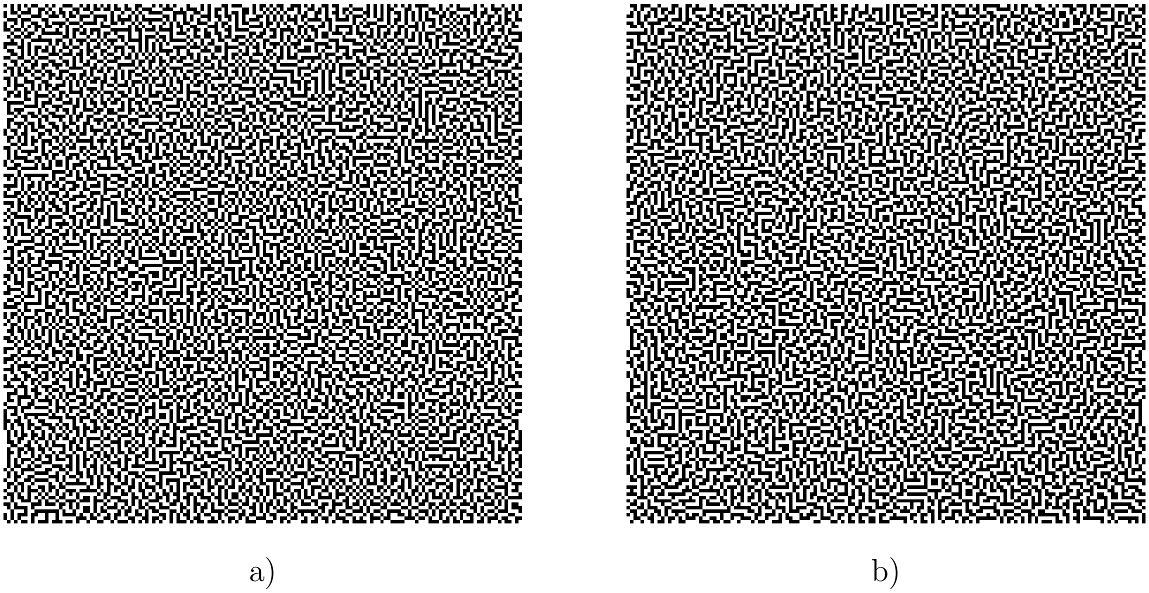
We start with a random halftone consisting of clusters of 2 × 2 pixels with 50% absorptance. We then apply monochrome DBS algorithm for the non-overlapping clusters of 2×2 pixels and generate an initial level. This step of the procedure has the constraint of not allowing the creation of checkerboard patterns, which led to localized artifacts in levels below and above an initial level. One example of eliminating the checkerboard patterns was discussed in [18], [17]. Their method involves an upper and lower tone-dependent threshold and uses a special procedure for the values in between those thresholds. In our work, we assume that random halftone doesn't have any checkerboard artifacts and we use the following approach in order to not create the checkerboard artifacts. For each candidate cluster swap, we check if the removal of the candidate dot cluster in the corresponding region will create a checkerboard. Below is the algorithm that we used to perform the check:

1. Temporarily turn off the candidate dot cluster for swapping.
2. Check whether the left neighbor of the candidate dot cluster has a dot cluster in a 4 - cluster neighborhood. If no, then the candidate dot removal is not allowed, else go to step 3.
3. Repeat step 2 for top, right and bottom neighbors of the candidate dot cluster.
4. Allow the candidate dot removal, i.e. the checkerboard will not be created.

An example of initial level with and without the checkerboard constraint is shown in Fig. 6.3. In order to apply monochrome DBS algorithm for the non-overlapping clusters of 2×2 pixels, DBS equations needed to be updated. Below is the review of the updates.

6.5.2 Generating an optimized initial level

The initial level halftone introduced in the previous subsection consists only of non-overlapping clusters of 2×2 pixels. The goal of our screen design was to work with overlapping clusters of 2×2 pixels. Therefore, it was decided to grow some clusters of 2×2 pixels by swapping single dot pixels with single hole pixels, and



a) b)

Fig. 6.3.: Initial level for $M = 300$ with and without the checkerboard constraint: a) Without the checkerboard constraint; b) With the checkerboard constraint.

creating dot and hole clusters of at least 2×2 pixels. The procedure for optimizing the initial level is the following:

1. Take the first candidate dot pixel and check if the remaining cluster without that pixel has a size of at least 2×2 pixels. Temporarily assign its value to 0 in order to search for candidate hole pixels.

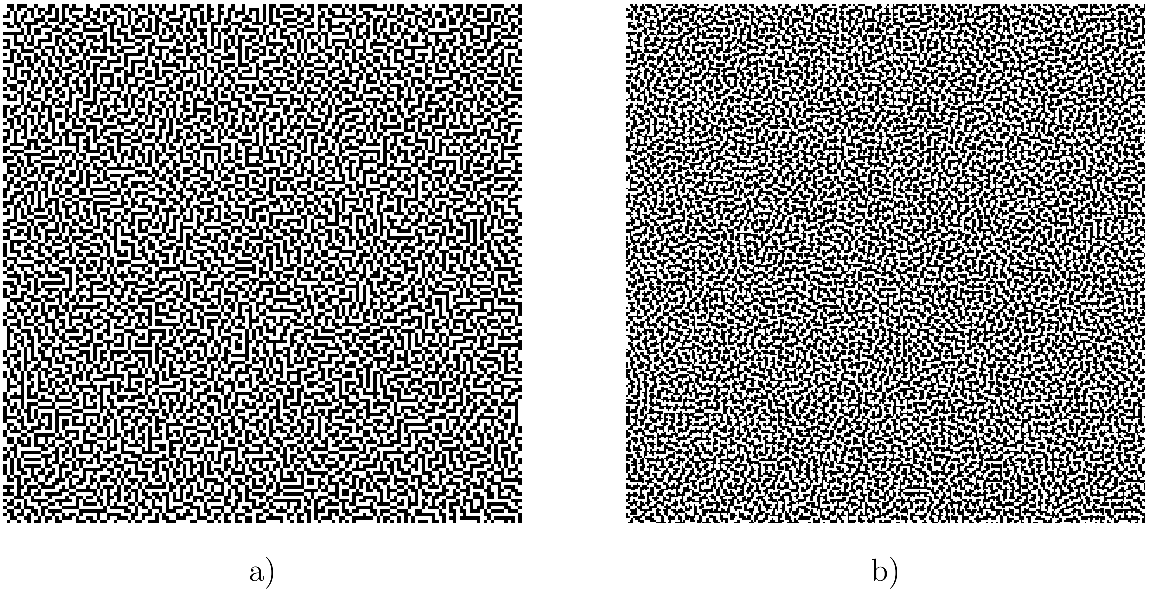
2. Find the candidate hole pixels, which are connected to the cluster of pixels that has a size of at least 2×2 pixels.

3. Compute the ΔE of swapping the candidate dot pixel with all candidate hole pixels, and find an optimum position for the candidate dot pixel if such a position exists.

4. Repeat steps 1-3 for all candidate dot pixels.

5. Iterate through the entire image until no more swaps are accepted.

An example of an initial level with an optimized initial level is shown in Fig. 6.4. Optimized initial level looks much smoother than an initial level. It can be seen from Fig. 6.4 b) that each dot pixel is either a part of a 2×2 dot cluster or is adjacent to the 2×2 dot cluster. Similarly, each hole pixel is either a part of a 2×2 hole cluster or is adjacent to the 2×2 hole cluster. In order to control the allowable length of the path of single pixels attached to a cluster of 2×2, *pathDepthMax* parameters is used.



a)

b)

Fig. 6.4.: Initial level and optimized initial level for $M = 300$: a) Initial level; b) Optimized initial level.

6.5.3 Generating remaining levels

In order to generate the levels below and levels above the optimized initial level, DBS with toggle is used and the stacking constraints are preserved. Due to the fact that the optimized initial level consists of dot (or hole) clusters, where each dot (or hole) pixel is either a part of a 2×2 cluster or is adjacent to the 2×2 cluster, it was decided that the following toggling operations should be considered: toggling a single pixel, toggling clusters of 2 × 2, 1 × 2 or 2 × 1. When considering toggling a block of pixels, we check that the following condition is preserved: each remaining dot (or hole) pixel should be either a part of a 2 × 2 dot (or hole) cluster or should be adjacent to the 2 × 2 dot (or hole) cluster. The number of levels is dependent on whether a single pixel, or a cluster of 2 × 2, 1 × 2 or 2 × 1 pixels are toggled at each step.

DBS update equations when going from level to level.

When going from one level to the next, the continuous tone image needs to be modified in order to compute the error:

$$f'[\mathbf{m}] = f[\mathbf{m}] + \Delta l \cdot 1[\mathbf{m}], \tag{6.25}$$

where Δl is the graylevel update. Similarly as before, the halftone image is updated after toggle at $\mathbf{m}_0$

$$g'[\mathbf{m}] = g[\mathbf{m}] + a_0\delta[\mathbf{m} - \mathbf{m}_0]. \tag{6.26}$$

Therefore, the error is

$$\begin{aligned} e'[\mathbf{m}] &= g'[\mathbf{m}] - f'[\mathbf{m}] \qquad (6.27)\\ &= (g[\mathbf{m}] + a_0\delta[\mathbf{m} - \mathbf{m}_0]) - (f[\mathbf{m}] + \Delta l \cdot 1[\mathbf{m}])\\ &= (g[\mathbf{m}] - f[\mathbf{m}]) + a_0\delta[\mathbf{m} - \mathbf{m}_0] - \Delta l \cdot 1[\mathbf{m}]\\ &= e[\mathbf{m}] + a_0\delta[\mathbf{m} - \mathbf{m}_0] - \Delta l \cdot 1[\mathbf{m}]. \end{aligned}$$

As a result, the $c_{\tilde{p}\tilde{e}}{}'[\mathbf{m}]$ can be obtained as follows

$$\begin{aligned} c_{\tilde{p}\tilde{e}}{}'[\mathbf{m}] &= e'[\mathbf{m}] ** c_{\tilde{p}\tilde{p}}[\mathbf{m}] \qquad (6.28)\\ &= (e[\mathbf{m}] + a_0\delta[\mathbf{m} - \mathbf{m}_0] - \Delta l \cdot 1[\mathbf{m}]) ** c_{\tilde{p}\tilde{p}}[\mathbf{m}]\\ &= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0\, c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_0] - \Delta l \cdot 1[\mathbf{m}]. \end{aligned}$$

6.5.4 Extension to clusters of 3×3 pixels

In the case of clusters of 3×3 pixels, the design procedure is the same, but DBS equations need to be updated. We use $\mathbf{m} = [m, n]$ to represent the discrete spatial coordinates as before. Let the 1st cluster consist of pixels with coordinates $(\mathbf{m}_0, \mathbf{m}_1, ..., \mathbf{m}_8)$, and the second cluster consist of pixels with coordinates $(\mathbf{n}_0, \mathbf{n}_1, ..., \mathbf{n}_8)$. Next, in order to compute the ΔE and $c_{\tilde{p}\tilde{e}}[\mathbf{m}]$ update equations, we swap 9 pairs of pixels step by step, and then add updates to determine the final result. Hence, final update equations for swapping are:

$$\begin{aligned}
\Delta E_{update} &= \sum_{i=0}^{8} \Delta E_i \qquad (6.29)\\
&= 9(a_0^2 + a_1^2)\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0 \sum_{i=0}^{8} c_{\tilde{p}\tilde{e}}[\mathbf{m}_i] + 2a_1 \sum_{i=0}^{8} c_{\tilde{p}\tilde{e}}[\mathbf{n}_i] + 2a_0 a_1 \sum_{i=0}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{m}_i - \mathbf{n}_i]\\
&+ 2a_0^2 \sum_{j=0}^{8} \sum_{i=j+1}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{m}_i - \mathbf{m}_j] + 2a_0 a_1 \sum_{j=0}^{8} \sum_{i=j+1}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{m}_i - \mathbf{n}_j]\\
&+ 2a_1 a_0 \sum_{j=0}^{8} \sum_{i=j+1}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{n}_i - \mathbf{m}_j] + 2a_1^2 \sum_{j=0}^{8} \sum_{i=j+1}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{n}_i - \mathbf{n}_j]\\
c_{\tilde{p}\tilde{e}}^{'''''''''}[\mathbf{m}] &= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0 \sum_{i=0}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_i] + a_1 \sum_{i=0}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{n}_i]
\end{aligned}$$

Similarly, for toggling the update equations are

$$\begin{aligned}
\Delta E_{update} &= \sum_{i=0}^{8} \Delta E_i \qquad (6.30)\\
&= 9a_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0 \sum_{i=0}^{8} c_{\tilde{p}\tilde{e}}[\mathbf{m}_i] + 2a_0^2 \sum_{j=1}^{8} \sum_{i=0}^{j-1} c_{\tilde{p}\tilde{p}}[\mathbf{m}_j - \mathbf{m}_i]\\
c_{\tilde{p}\tilde{e}}^{'''''''''}[\mathbf{m}] &= c_{\tilde{p}\tilde{e}}[\mathbf{m}] + a_0 \sum_{i=0}^{8} c_{\tilde{p}\tilde{p}}[\mathbf{m} - \mathbf{m}_i]
\end{aligned}$$

Another update was made when generating levels below and above the optimized initial level. Previously, we considered toggling either a single pixel or clusters of 2×2, 1×2, or 2×1. In the case of clusters of 3×3, the variety of possible clusters for toggling is increased. We can now consider toggling clusters of 3 × 3, 1 × 3, 3 × 1, 2 × 2, 1 × 2, or 2 × 1. Therefore, the general form for toggling k pixels is:

$$\Delta E_{update} = ka_0^2\, c_{\tilde{p}\tilde{p}}[\mathbf{0}] + 2a_0 \sum_{i=0}^{k-1} c_{\tilde{p}\tilde{e}}[\mathbf{m}_i] + 2a_0^2 \sum_{j=1}^{k-1} \sum_{i=0}^{j-1} \mathbf{m}_j - \mathbf{m}_i \tag{6.31}$$

6.6 Experimental results

In order to demonstrate our results, we chose to work with the matrix size $M = 800$. For comparison, we first generated a thresholding matrix using monochrome DBS-based screen design with non-overlapping clusters of 2×2 pixels. In this case, the design procedure is analogous to the overlapping case described previously, however, in this case, all clusters consist of 2 × 2 pixels and there are no overlaps between the clusters. After obtaining the thresholding matrix, we halftoned the K ramp, and the resulting ramp is shown in Fig. 6.5. Similarly, we generated a thresholding matrix using monochrome DBS-based screen design with overlapping clusters of 2 × 2 pixels and the resulting ramp is shown in Fig. 6.6. By comparing Figs. 6.5 and 6.6, it can be concluded that the ramp image halftoned with the matrix that was generated with monochrome DBS-based screen design with overlapping clusters of 2 × 2 pixels looks much smoother than the ramp image halftoned with the matrix in the non-overlapping case. There are, however, some localized artifacts at the quartertones of the ramp. These artifacts are due to the greedy algorithm that tries to minimize the cost at every step.

6.7 Conclusion

In this chapter, we presented a new monochrome DBS-based screen design with overlapping clusters of 2×2 or 3×3 pixels. The clusters of 2×2 or 3×3 pixels are used in electrophotographic printers due to their print stability. We started by designing screens using monochrome DBS-based screen design with non-overlapping clusters of

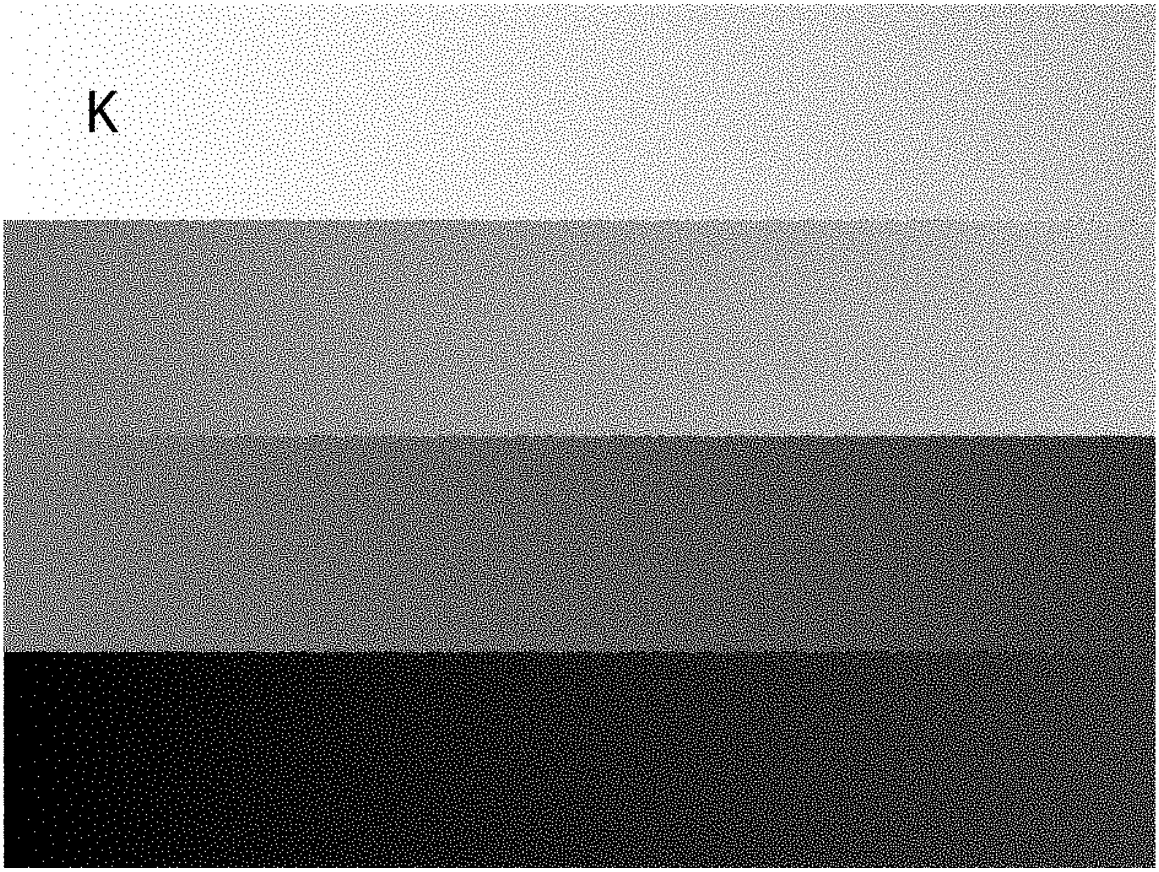


Fig. 6.5.: Ramp image halftoned with matrix of size 800 × 800 that was generated with monochrome DBS-based screen design with non-overlapping clusters of 2 × 2 pixels.

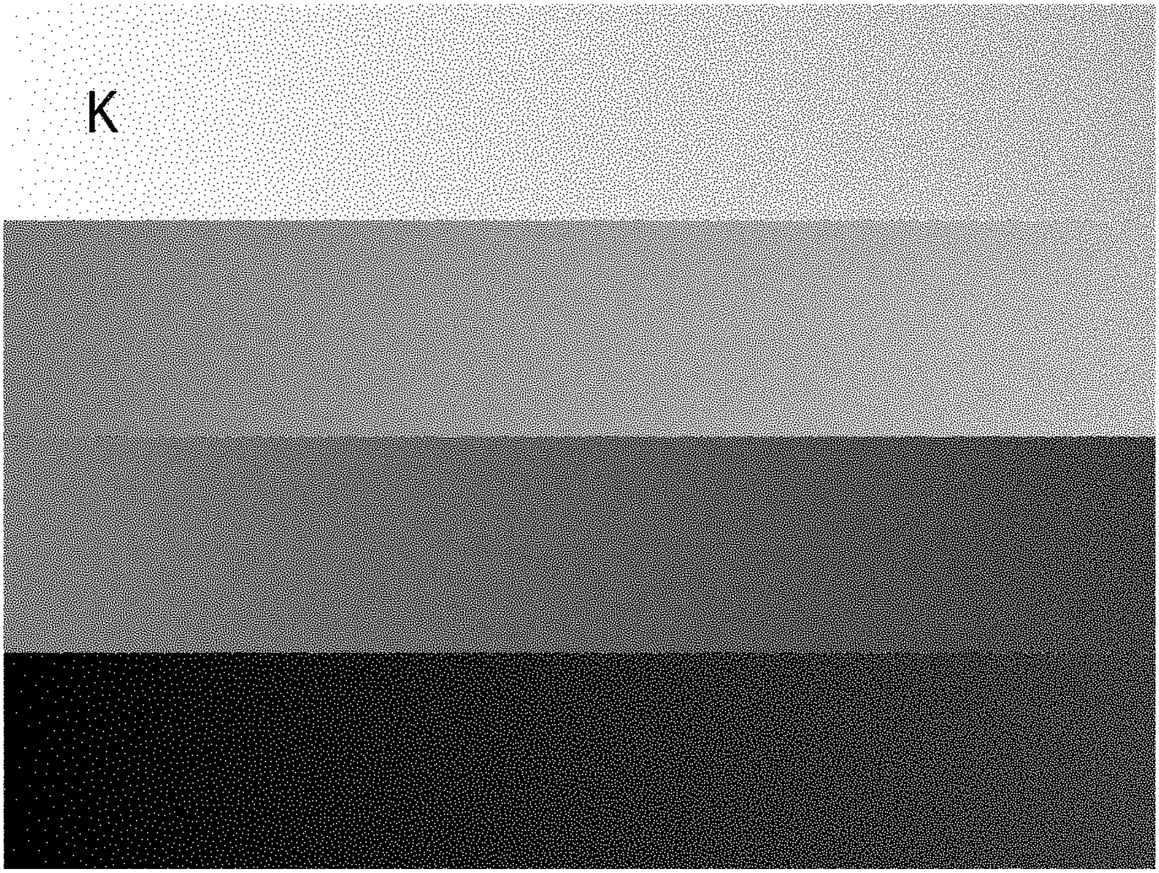


Fig. 6.6.: Ramp image halftoned with matrix of size 800 × 800 that was generated with monochrome DBS-based screen design with overlapping clusters of 2 × 2 pixels.

2×2 pixels. We then came up with the approach to create dot and hole clusters that contain the region of 2×2 pixels inside of them, which led us to the monochrome DBS-based screen design with overlapping clusters of 2×2 pixels. We updated the DBS equations for swapping and toggling clusters of pixels instead of swapping and toggling single pixels. After working with 2×2 clusters, we also considered using 3×3 clusters, which are even more stable than the 2×2 clusters. Finally, we presented experimental results to illustrate the efficacy of our new approach. Our results show that the screen design with overlapping clusters produces much smoother halftones than the screen design with non-overlapping clusters.

7. CONCLUSION

7.1 Summary and major contributions

In this dissertation, we investigated model-based analysis and design of color screen sets for clustered-dot periodic halftoning and design of monochrome screens based on direct binary search for aperiodic dispersed-dot halftoning. In Chapter 2, we described the single separation analysis for clustered-dot halftones. We started by introducing regular and irregular clustered-dot halftones. We then presented an algorithm for obtaining candidate screen tile vectors given target screen parameters. After that we developed a framework for determining the minimum set of frequency components that must be included in the evaluation of halftone fluctuation. We proposed a set of four ratios to quantify this fluctuation. These ratios are fully based on an analytic expression for the Continuous-Space Fourier Transform (CSFT) of the halftone pattern. As a result, we developed a tool that can be used to eliminate periodicity matrices that will result in poor quality halftones, based only on an analysis of the periodicity matrix itself, without considering the specific threshold matrix that will control the growth of dot clusters in highlights and hole clusters in shadows, as well as the midtone halftone textures.

Major contributions of Chapter 2:

1. Presented a procedure for choosing an optimal irregular screen with a given degree of irregularity. These screens can achieve a closer approximation to the target screen frequency and angle.

2. Provided a measure for computing image fluctuation for a single separation based only on the periodicity matrix.

3. Conducted an in-depth Fourier Analysis in order to establish the behavior of the frequency components in regular and irregular clustered dot halftones.

4. Found out that with irregular clustered-dot halftones, there are additional frequency components in the digital rendering spectra that are not present in the analog rendering spectra.

5. Concluded that there's a trade-off between getting closer to the target screen parameters and the graininess levels.

In Chapter 3, we presented digital simulation-based analysis of halftoning with an arbitrary number of separations (regular or irregular) using the spatiochromatic HVS model. Our goal was to design a predictor that will help us assign colorants to different screens that will minimize the perceived error. We started by integrating the HVS based model that takes into account the spectral representation of color and the difference in the responses of the human viewer to luminance and chrominance information in order to obtain the perceived error metric. To start, we investigated superimposing of two clustered-dot color halftones only. The perceived error metric demonstrated that the color assignment plays an important role in the visual appearance of the resulting superposition image. We then expanded our model to the superposition of three and four clustered dot color halftones as well as looking at different absorptance combinations. In the second part of Chapter 3, we presented results of the analysis of impact of screen configuration on image fluctuation and colorshift across all absorptance combinations. Due to the fact that we expanded our model for different absorptance combinations, we needed to propose and implement the screen design algorithm for irregular clustered-dot halftones. The main challenge with the screen design of irregular clustered-dot halftones was that, unlike with regular halftones, the microcells within the tile were not identical. Hence, we had to generate all microcell growing sequences for all microcells within a tile, and then tile them. After that, we looked into the effect of misregistration on image fluctuation and color shift, which are very common in electrophotographic printers.

Major contributions of Chapter 3:

1. Presented a procedure for making the best color assignments to the two, three or four regular or irregular halftones in order to minimize the perceived error.

2. Developed tools for digitally analyzing the image fluctuation due to halftone structure and image fluctuation and color shift due to misregistration for arbitrary screen sets.

3. Performed the analysis of the interdependence of the screen configuration and the color that's being halftoned.

In Chapter 4, we proposed to perform content-color-dependent screening (CCDS) instead of using a single screen set for one image. We first presented the procedure for CCDS, which involves obtaining the cluster-map by clustering the color content using K-means algorithm, obtaining the segmented edge-map, merging the cluster-map and the edge-map into a final map, determining the best color assignments for different clusters using the HVS-based model presented in Chapter 3, and finally halftoning with the optimal color assignments for their corresponding clusters. Since the HVS-based model determines the color assignments that will minimize the perceived error, and the entire image will be halftoned with the best color assignments based on the color content, we believe that the CCDS approach can move the quality of color prints generated by limited-resolution digital presses closer to that of the much higher resolution analog offset printing presses with which the digital presses are competing.

Major contributions of Chapter 4:

1. Proposed and implemented content-color-dependent screening method using clustered-dot color halftones, which helps us produce prints with better quality.

2. The main novelty of our approach is the application of different color assignments within the image depending on the local color and content of the image, without changing the overall set of screens that are used.

In Chapter 5, we presented design of standard and high resolution, periodic, clustered-dot color screen sets with focus on impact of hybrid screening method on highlights. We started with the screen design approach for generating base levels for irregular clustered-dot halftones presented in Chapter 3. Next, we proposed the design of standard resolution periodic irregular clustered-dot halftones with the hybrid screen design. We used the hybrid screen design proposed by Lee and Allebach in [47] as the foundation for our approach. The novelty of our approach lies in applying the hybrid screen design to the irregular clustered-dot halftones, where the microcell growing sequences are unique for each microcell within the tile. Therefore, some of the steps in the procedure had to be modified. In addition, we extended the hybrid screen design to high-resolution periodic clustered-dot halftones, where instead of using a single pixel at the center of a microcell we used a cluster of 2×2 pixels.

Major contributions of Chapter 5:

1. The novelty of our approach lies in applying the hybrid screen design to the irregular clustered-dot halftones, where microcell growing sequences are unique for each microcell within the tile.

2. Implemented the hybrid screen design for high-resolution periodic clustered-dot halftones.

In Chapter 6, we proposed monochrome DBS-based screen design with overlapping clusters of 2×2 or 3×3 pixels that can be used in electrophotographic printers. We started by reviewing monochrome DBS-based screen design which produces dispersed-dot textures and is usually used for inkjet printers, which have no trouble in rendering isolated dots. Due to the fact that with electrophotographic printers, isolated dots are not stable, we proposed to use a cluster of 2×2 or 3×3 pixels as a single unit instead of a single pixel. We provided the updated DBS equations for performing swapping and toggling of clusters. Next, it was decided to update monochrome DBS-based screen design by considering the clusters to overlap. Our results showed that the screen design with overlapping clusters produces much smoother halftones than the screen design with non-overlapping clusters.

Major contributions of Chapter 6:

1. Implemented the monochrome DBS-based screen design with non-overlapping clusters of 2 × 2 pixels.

2. Presented and implemented a new monochrome DBS-based screen design with overlapping clusters of 2 × 2 or 3 × 3 pixels.

7.2 My publications

1. A. Jumabayeva, Y.-T. Chen, T. Frank, R. Ulichney, and J. P. Allebach, "Design of irregular screen sets that generate maximally smooth halftone patterns," in Proc. IS&T/SPIE Electron. Imaging, 20th Color Imaging: Displaying, Process., Hardcopy, and Appl., vol. 9395, San Francisco, CA, Feb. 2015.

2. A. Jumabayeva, T. Frank, Y. Ben-Shoshan, R. Ulichney, and J. P. Allebach, "HVS-based model for superposition of two color halftones," in Proc. IS&T Electron. Imaging, 21st Color Imaging: Displaying, Process., Hardcopy, and Appl., vol. 9395, San Francisco, CA, Feb. 2016.

3. A. Jumabayeva, Y.-T. Chen, T. Frank, R. Ulichney, and J. P. Allebach, "Single separation analysis for clustered dot halftones," in Proc. 2016 IEEE Int. Conf. on Image Process., Phoenix, AZ, Sep. 2016, pp. 4383 – 4387

4. A. Jumabayeva, T. Frank, Y. Ben-Shoshan, R. Ulichney, and J. P. Allebach, "Content-color-dependent screening (CCDS) using regular or irregular clustered-dot halftones," in Proc. 2018 7th Eur. Workshop Visual Inform. Process., Tampere, Finland, Nov. 2018.

5. A. Jumabayeva and J. P. Allebach, "Assigning halftone screens based on color and content", U.S. Patent Application 2019-ALLE-68610, Mar. 2019

7.3 Future work

Below is the list of items that could be investigated further:

1. Print patches using regular or irregular halftones for CMYK separations and perform a psychophysical experiment in order to check whether our HVS-based model for superposition of multiple halftones predicts the image fluctuation correctly.

2. Investigate machine learning methods to locate smooth areas in the image for color-content-dependent screening (CCDS).

3. Improve the computation efficiency of the monochrome DBS-based screen design with overlapping clusters of 2×2 or 3×3 clusters by generating fewer number of levels, instead of using a greedy approach and generating all possible number of levels.

4. Update the monochrome DBS-based screen design with overlapping clusters for multiple color separations.

5. Update the monochrome DBS-based screen design with overlapping clusters for multilevel halftoning instead of bilevel halftoning.

REFERENCES

REFERENCES

[1] I. D. Lau and R. Gonzalo, *Modern Digital Halftoning.* Boca Raton, FL: CRC Press, 2008.

[2] F. Pelcastre-Jimenez, L. Rosales-Roldan, M. Nakano-Miyatake, and H. Perez-Meana, "Inverse halftoning using a multilayer perceptron neural network," in *Proc. Int. Conf. Elect. Commun. Comput.*, Puebla, Mexico, Feb. 2012, pp. 202–206, doi:10.1109/CONIELECOMP.2012.6189909.

[3] Q.-X. Ong and W.-L. Hsue, "Inverse halftoning based on edge detection classification," in *Proc. Asia-Pacific Signal Inform. Process. Assoc. Annu. Summit Conf.*, vol. 4, Kaohsiung, Taiwan, Oct. 2013, doi:10.1109/APSIPA.2013.6694350.

[4] T. Kite, N. Damera-Venkata, B. Evans, and A. Bovik, "A fast, high-quality inverse halftoning algorithm for error diffused halftones," *IEEE Trans. Image Process.*, vol. 9, pp. 1583 – 1592, Sep. 2000, doi:10.1109/83.862639.

[5] Y. Yi-fei, "Space dependent quality assessment for color inverse halftoning images," *J. Comput. Appl.*, vol. 29, pp. 1699–1701, Jun. 2009, doi:10.3724/SP.J.1087.2009.01699.

[6] F. Lujing, Z. Yunfei, and Z. Tao, "B-spline wavelet based color printing scan images inverse halftoning," *Comput. Eng. Appl.*, vol. 49, pp. 178–181, Jul. 2013, doi:10.3778/j.issn.1002-8331.1111-0285.

[7] C.-I. Lin, Y.-P. Sie, T.-H. Lin, and P.-L. Sun, "Slicing and halftoning algorithm for high quality color 3D printing," in *Proc. Int. Display Workshop*, vol. 2, Otsu, Japan, Dec. 2015, pp. 786–789.

[8] O. Bulan, G. Sharma, and V. Monga, "Orientation modulation for data hiding in clustered-dot halftone prints," *IEEE Trans. Image Process.*, vol. 19, pp. 2070 – 2084, Aug. 2010, doi:10.1109/TIP.2010.2046795.

[9] J.-M. Guo and Y.-F. Liu, "Halftone-image security improving using overall minimal-error searching," *IEEE Trans. Image Process.*, vol. 20, pp. 2800 – 2812, Oct. 2011, doi:10.1109/TIP.2011.2131667.

[10] ——, "Joint compression/watermarking scheme using majority-parity guidance and halftoning-based block truncation coding," *IEEE Trans. Image Process.*, vol. 19, pp. 2056 – 2069, Aug. 2011, doi:10.1109/TIP.2010.2045709.

[11] J.-M. Guo, C.-C. Su, Y.-F. Liu, H. Lee, and J.-D. Lee, "Oriented modulation for watermarking in direct binary search halftone images," *IEEE Trans. Image Process.*, vol. 21, pp. 4117 – 4127, Sep. 2012, doi:10.1109/TIP.2012.2198221.

[12] J.-M. Guo, G.-H. Lai, K. Wong, and L.-C. Chang, “Progressive halftone watermarking using multilayer table lookup strategy,” *IEEE Trans. Image Process.*, vol. 24, pp. 2009 – 2024, Jul. 2015, doi:10.1109/TIP.2014.2387417.

[13] “Commercial printing industry profile,” Dun&Bradstreet First Research, May 2019. [Online]. Available: http://www.firstresearch.com/Industry-Research/Commercial-Printing.html

[14] A. Paparozzi, “How the commercial printing industry is being redefined in 2018,” Printing Impressions, Dec. 2017. [Online]. Available: https://www.piworld.com/article/commercial-printing-industry-redefined/

[15] J. A. C. Yule, *Principles of Color Reproduction, Applied to Photomechanical Reproduction, Color Photography and the Ink, Paper and Other Related Industries.* New York, NY: Wiley, 1967.

[16] R. Ulichney, *Digital Halftoning.* Cambridge, MA: MIT Press, 1987.

[17] T. Chang and J. P. Allebach, “Memory efficient error diffusion,” *IEEE Trans. Image Process.*, vol. 12, pp. 1352–1366, Nov. 2003, doi:10.1109/TIP.2003.818214.

[18] P. Li and J. P. Allebach, “Tone-dependent error diffusion,” *IEEE Trans. Image Process.*, vol. 13, pp. 201–215, Feb. 2004, doi:10.1109/TIP.2003.819232.

[19] Y.-H. Fung and Y.-H. Chan, “Green noise digital halftoning with multiscale error diffusion,” *IEEE Trans. Image Process.*, vol. 19, pp. 1808–1823, Jul. 2010, doi:10.1109/TIP.2010.2044961.

[20] ——, “Optimizing the error diffusion filter for blue noise halftoning with multiscale error diffusion,” *IEEE Trans. Image Process.*, vol. 22, pp. 413 – 417, Jan. 2013, doi:10.1109/TIP.2012.2211370.

[21] Z. He, “Hierarchical error diffusion,” *IEEE Trans. Image Process.*, vol. 18, pp. 1524 – 1535, Jul. 2009, doi:10.1109/TIP.2009.2019778.

[22] G. Sarailidis and I. Katsavounidis, “A multiscale error diffusion technique for digital multitoning,” *IEEE Trans. Image Process.*, vol. 21, pp. 2693 – 2705, May 2012, doi:10.1109/TIP.2012.2185936.

[23] D. E. Knuth, “Digital halftones by dot diffusion,” *ACM Trans. Graph.*, vol. 6, p. 245–273, Oct. 1987, doi:10.1145/35039.35040.

[24] Y.-F. Liu and J.-M. Guo, “New class tiling design for dot-diffused halftoning,” *IEEE Trans. Image Process.*, vol. 22, pp. 1199 – 1208, Mar. 2013, doi:10.1109/TIP.2012.2228491.

[25] J. M. Guo and Y. F. Liu, “Improved dot diffusion by diffused matrix and class matrix co-optimization,” *IEEE Trans. Image Process.*, vol. 18, p. 1804–1816, Aug. 2009, doi:10.1109/TIP.2009.2021318.

[26] J. P. Allebach and Q. Lin, “FM screen design using DBS algorithm,” in *Proc. 1996 IEEE Int. Conf. on Image Process.*, Lausanne, Switzerland, Sep. 1996, p. 549–552, doi:10.1109/ICIP.1996.559555.

[27] D. J. Lieberman and J. P. Allebach, “Efficient model based halftoning using direct binary search,” in *Proc. 1997 IEEE Int. Conf. on Image Process.*, Santa Barbara, CA, Oct. 1997, p. 775–778, doi:10.1109/ICIP.1997.648077.

[28] J.-R. Liao, “Theoretical bounds of direct binary search halftoning,” *IEEE Trans. Image Process.*, vol. 24, pp. 3478 – 3487, Nov. 2015, doi:10.1109/TIP.2015.2449077.

[29] J. Havlicek, D. Harding, and A. Bovik, “The multicomponent AM-FM image representation,” *IEEE Trans. Image Process.*, vol. 5, pp. 1094 – 1100, Jun. 1996, doi:10.1109/83.503927.

[30] M. Pattichis, G. Panayi, A. Bovik, and S.-P. Hsu, “Fingerprint classification using an AM–FM model,” *IEEE Trans. Image Process.*, vol. 10, pp. 951 – 954, Jun. 2001, doi:10.1109/83.923291.

[31] S. Acton, D. P. Mukherjee, J. Havlicek, and A. C. Bovik, “Oriented texture completion by AM–FM reaction–diffusion,” *IEEE Trans. Image Process.*, vol. 10, pp. 885 – 896, Jun. 2001, doi:10.1109/83.923285.

[32] R. A. Ulichney, “Dithering with blue noise,” *Proc. IEEE*, vol. 76, pp. 56 – 79, Jan. 1988, doi:10.1109/5.3288.

[33] D. Lau, R. Ulichney, and G. Arce, “Blue and green noise halftoning models,” *IEEE Trans. Signal Process.*, vol. 20, pp. 28 – 38, Jul. 2003, doi:10.1109/MSP.2003.1215229.

[34] Z. He, “Hierarchical colorant-based direct binary search halftoning,” *IEEE Trans. Image Process.*, vol. 19, pp. 1824 – 1836, Jul. 2010, doi:10.1109/TIP.2010.2045690.

[35] K. Ishizaka, “New spatial measure for dispersed-dot halftoning assuring good point distribution in any density,” *IEEE Trans. Image Process.*, vol. 18, pp. 2030 – 2047, Sep. 2009, doi:10.1109/TIP.2009.2022443.

[36] P. Morovic, J. Morovic, J. Gondek, and R. Ulichney, “Direct pattern control halftoning of neugebauer primaries,” *IEEE Trans. Image Process.*, vol. 26, pp. 4404 – 4413, Sep. 2017, doi:10.1109/TIP.2017.2713939.

[37] ——, “Direct pattern control halftoning of neugebauer primaries,” *IEEE Trans. Image Process.*, vol. 26, pp. 4404–4413, Sep. 2017, doi:10.1109/TIP.2017.2713939.

[38] J. Morovic, P. Morovic, and J. Arnabat, “HANS: Controlling ink-jet print attributes via neugebauer primary area coverages,” *IEEE Trans. Image Process.*, vol. 21, pp. 688 – 696, Feb. 2012, doi:10.1109/TIP.2011.2164418.

[39] D. L. Lau, G. R. Arce, and N. C. Gallagher, “Green-noise digital halftoning,” *Proc. IEEE*, vol. 86, pp. 2424 – 2444, Dec. 1998, doi:10.1109/5.735449.

[40] ——, “Digital color halftoning with generalized error diffusion and multichannel green-noise masks,” *IEEE Trans. Image Process.*, vol. 9, p. 923–935, May 2000, doi:10.1109/83.841537.

[41] V. Babaei and R. D. Hersch, “Juxtaposed color halftoning relying on discrete lines,” *IEEE Trans. Image Process.*, vol. 22, pp. 679 – 686, Feb. 2013, doi:10.1109/TIP.2012.2221727.

[42] ——, “N-ink printer characterization with barycentric subdivision,” *IEEE Trans. Image Process.*, vol. 25, pp. 3023 – 3031, Jul. 2016, doi:10.1109/TIP.2016.2560526.

[43] W. Jiang, A. Veis, Y. Ekstein, R. Ulichney, and J. Allebach, “Novel color halftoning algorithm for ink savings,” in *Proc. IS&T Electron. Imaging, 23rd Color Imaging: Displaying, Process., Hardcopy, Appl.*, San-Francisco, CA, Feb. 2018, doi:https://doi.org/10.2352/ISSN.2470-1173.2018.16.COLOR-429.

[44] “Silky screen: neuer raster von HP,” LabelPack, Jun. 2016. [Online]. Available: https://www.labelpack.de/silky-screen-neuer-raster-von-hp/

[45] Y. Chen, T. Kashti, M. Fischer, D. Shaked, R. Ulichney, and J. P. Allebach, “The lattice-based screen set: a square n - color all-orders moiré-free screen set,” *IEEE Trans. Image Process.*, vol. 25, pp. 1873 – 1886, Apr. 2016, doi: 10.1109/TIP.2016.2523428.

[46] Y.-Y. Chen, M. Fischer, T. Kashti, D. Shaked, and J. P. Allebach, “The lattice-based screen set: a square N-color all-orders moiré-free screen set,” in *Proc. IS&T/SPIE Electron. Imaging, 17th Color Imaging: Displaying, Process., Hardcopy, Appl.*, vol. 8292, Burlingame, CA, Jan. 2012, doi:10.1117/12.912297.

[47] C. Lee and J. P. Allebach, “The hybrid screen - improving the breed,” *IEEE Trans. Image Process.*, vol. 19, pp. 435 – 450, Feb. 2010, doi: 10.1109/TIP.2009.2032941.

[48] F. A. Baqai and J. P. Allebach, “Computer-aided design of clustered-dot color screens based on a human visual system model,” *Proc. IEEE*, vol. 90, pp. 104–122, Jan. 2002, doi:10.1109/5.982409.

[49] C. Tang, A. Veis, R. Ulichney, and J. P. Allebach, “Irregular clustered-dot periodic halftone screen design,” in *Proc. IS&T/SPIE Electron. Imaging, 19th Color Imaging: Displaying, Process., Hardcopy, Appl.*, vol. 9015, San Francisco, CA, Feb. 2015, doi:10.1117/12.2042428.

[50] J.-Y. Kim, Y.-Y. Chen, M. Fischer, O. Shacham, C. Staelin, K. Bengtson, and J. P. Allebach, “Design of color screen tile vector sets,” in *Proc. IS&T/SPIE Electron. Imaging, 16th Color Imaging: Displaying, Process., Hardcopy, Appl.*, vol. 7866, San Francisco, CA, Jan. 2011, doi:10.1117/12.878658.

[51] I. Amidror and R. D. Hersch, “Analysis of the superposition of periodic layers and their moiré effects through the algebraic structure of their fourier spectrum,” *J. Math. Imaging Vision*, vol. 8, pp. 99 – 130, Mar. 1998, doi:10.1023/A:1008248527004.

[52] ——, “Analysis of the microstructures (”rosettes”) in the superposition of periodic layers,” *J. Electron. Imaging*, vol. 11, Jul. 2002, doi:10.1117/1.1477442.

[53] S.-G. Wang and R. Loce, “Uniform-rosette color halftoning for N-color moiré-free printing,” *J. Electron. Imaging*, vol. 17, Apr. 2008, doi:10.1117/1.2907206.

[54] B. Oztan, G. Sharma, and R. P. Loce, “Conditions for color misregistration sensitivity in clustered-dot halftones,” in *Proc. 2007 IEEE Int. Conf. on Image Process.*, San Antonio, TX, Sep. 2007, doi:10.1109/ICIP.2007.4379994.

[55] ——, "Misregistration sensitivity in clustered-dot color halftones," *J. Electron. Imaging*, vol. 17, Apr. 2008, doi:10.1117/1.2917517.

[56] J.-Y. Kim, Y.-Y. Chen, M. Fischer, O. Shacham, C. Staelin, and J. P. Allebach, "Design of color screen sets for robustness to color plane misregistration," in *Proc. 2011 IEEE Int. Conf. on Image Process.*, Brussels, Belgium, Sep. 2011, pp. 1733–1736, doi:10.1109/ICIP.2011.6115793.

[57] A. U. Agar and J. P. Allebach, "Model-based color halftoning using direct binary search," *IEEE Trans. Image Process.*, vol. 14, pp. 1945 – 1959, Dec. 2005, doi: 10.1109/TIP.2005.859380.

[58] S. J. Park, M. Q. Shaw, G. Kerby, T. Nelson, D.-Y. Tzeng, K. R. Bengston, and J. P. Allebach, "Halftone blending between smooth and detail screens to improve print quality with electrophotographic printers," *IEEE Trans. Image Process.*, vol. 25, pp. 601 – 614, Feb. 2016, doi:10.1109/TIP.2015.2500035.

[59] V. Ostromoukhov and S. Nehab, "Halftoning with gradient-based selection of dither matrices," U.S. Patent 5 701 366, Dec. 23, 1997.

[60] J. Huang and A. Bhattacharjya, "An adaptive halftone algorithm for composite documents," in *Proc. IS&T/SPIE Electron. Imaging, 9th Color Imaging: Process., Hardcopy, Appl.*, vol. 5293, San Jose, CA, Dec. 2003, doi:10.1117/12.532912.

[61] G.-Y. Lin and J. P. Allebach, "Generating stochastic dispersed and periodic clustered textures using a composite hybrid screen," *IEEE Trans. Image Process.*, vol. 15, pp. 3746–3758, Dec. 2006, doi:10.1109/TIP.2006.881968.

[62] T. M. Holladay, "Electronic halftone screening," U.S. Patent 4 185 304, Jan. 22, 1980.

[63] ——, "An optimum algorithm for halftone generation for displays and hard copies," *Proc. Soc. Inform. Display (SID)*, vol. 21, pp. 185 – 192, Jan. 1980.

[64] A. Jumabayeva, Y.-T. Chen, T. Frank, R. Ulichney, and J. P. Allebach, "Single separation analysis for clustered dot halftones," in *Proc. 2016 IEEE Int. Conf. on Image Process.*, Phoenix, AZ, Sep. 2016, pp. 4383 – 4387, doi:10.1109/ICIP.2016.7533188.

[65] ——, "Design of irregular screen sets that generate maximally smooth halftone patterns," in *Proc. IS&T/SPIE Electron. Imaging, 20th Color Imaging: Displaying, Process., Hardcopy, Appl.*, vol. 9395, San Francisco, CA, Feb. 2015, doi:10.1117/12.2083553.

[66] A. Jumabayeva, T. Frank, Y. Ben-Shohan, R. Ulichney, and J. P. Allebach, "HVS-based model for superposition of two color halftones," in *Proc. IS&T Electron. Imaging, 21st Color Imaging: Displaying, Process., Hardcopy, Appl.*, San-Francisco, CA, Feb. 2016, doi:10.2352/ISSN.2470-1173.2016.20.COLOR-311.

[67] Y.-T. Chen, T. Kashti, T. Frank, R. Ulichney, and J. P. Allebach, "Fourier-based analysis of regular and irregular periodic clustered-dot halftone textures," in *Int. Congr. Imaging Sci.*, May 2014.

[68] H. Kang, *Digital Color Halftoning.* Bellingham, WA: SPIE The International Society for Optical Engineering, 1999.

[69] S. H. Kim and J. P. Allebach, “Impact of HVS models on model-based halftoning,” *IEEE Trans. Image Process.*, vol. 11, pp. 258 – 269, Mar. 2002, doi: 10.1109/83.988959.

[70] 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. IS&T/SPIE Electron. Imaging, 4th Human Vision, Visual Process., Digital Display*, vol. 1913, San Jose, CA, Sep. 1993, doi:10.1117/12.152701.

[71] J. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Proc. 5th Berkeley Symp. Math. Statist. Probability*, vol. 1, Oakland, CA, 1967, pp. 281–297.

[72] K. McLaren, “The development of the CIE 1976 (L*a*b*) uniform colour space and colour-difference formula,” *J. Soc. Dyers Colourists*, vol. 92, pp. 338–341, Sep. 1976, doi:10.1111/j.1478-4408.1976.tb03301.x.

[73] S. Hu, Z. Pizlo, and J. P. Allebach, “JPEG ringing artifact visibility evaluation,” in *Proc. IS&T/SPIE Electron. Imaging, 11th Image Quality Syst. Performance*, vol. 9016, San Francisco, CA, Feb. 2014, doi:10.1117/12.2048594.

[74] P.-E. Danielsson and O. Seger, “Generalized and separable sobel operators,” in *Machine Vision for Three-Dimensional Scenes*, H. Freeman, Ed. Orlando, FL: Academic Press, Inc., Jul. 1990, pp. 347 – 381.

[75] X. Feng and J. P. Allebach, “Measurement of ringing artifacts in JPEG images,” in *Proc. IS&T/SPIE Electron. Imaging, Digital Publishing*, vol. 6076, San Jose, CA, Feb. 2006, doi:10.1117/12.645089.

[76] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, pp. 679–698, Nov. 1986, doi:10.1109/TPAMI.1986.4767851.

[77] T. Zhang and C. Y. Suen, “A fast parallel algorithm for thinning digital patterns,” *Commun. ACM*, vol. 27, pp. 236–239, Mar. 1984, doi:10.1145/357994.358023.

[78] L. G. Shapiro and R. M. Haralick, *Computer and Robot Vision.* Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 1992, vol. 1.

[79] R. J. Pellar, “Electronic halftone generator,” U.S. Patent 4 149 183, Apr. 10, 1979.

[80] G. Sharma, *Digital Color Imaging Handbook.* Boca Raton, FL: CRC Press, 2003.

[81] V. Babaei and R. D. Hersch, “Juxtaposed color halftoning relying on discrete lines,” *IEEE Trans. Image Process.*, vol. 22, pp. 679–686, Feb. 2013, doi:10.1109/TIP.2012.2221727.

[82] S. Gooran, D. Nyström, M. Namedanian, and S. Hauck, “Measuring register shift and investigating its effect on color appearance for different halftoning,” in *Proc. Tech. Assoc. Graph. Arts*, Pittsburgh, PA, Mar. 2011, pp. 225–232.

[83] Y.-F. Liu and J.-M. Guo, "Clustered-dot screen design for digital multitoning," *IEEE Trans. Image Process.*, vol. 25, pp. 2971 – 2982, Jul. 2016, doi:10.1109/TIP.2016.2552723.

[84] T. N. Pappas, J. P. Allebach, and D. L. Neuhoff, "Model-based digital halftoning," *IEEE Signal Process. Mag.*, vol. 20, pp. 14–27, Jul. 2003, doi: 10.1109/MSP.2003.1215228.

[85] R. Näsänen, "Visibility of halftone dot textures," *IEEE Trans. Syst., Man, Cybern.*, vol. 14, pp. 920–924, Nov. 1984, doi: 10.1109/TSMC.1984.6313320.

[86] D. J. Lieberman and J. P. Allebach, "A dual interpretation for direct binary search and its implications for tone reproduction and texture quality," *IEEE Trans. Image Process.*, vol. 9, pp. 1950–1963, Nov. 2000, doi: 10.1109/83.877215.

[87] M. Analoui and J. P. Allebach, "Model-based halftoning using direct binary search," in *Human Vision, Visual Process., Digital Display III, (Part of IS&T and SPIE Electron. Imaging 1992)*, vol. 1666, San Jose, CA, Aug. 1992, doi: 10.1117/12.135959.

APPENDICES

A. DERIVATION FOR OBTAINING PARAMETERS FOR OPPONENT CHANNELS VISUALIZATION

Our goal is to display an image defined as $(Y_{mid}, 0, C_z)$ in sRGB space, which we specify as

$$RGB_{FULL} = (R_{Cz}^{FULL}, G_{Cz}^{FULL}, B_{Cz}^{FULL}). \tag{A.1}$$

However, when we convert directly, we have negative values in sRGB space, which correspond to the opponency of channels (Y, C_x, C_z). We use the fact that

$$(Y_{mid}, 0, C_z) = (Y_{mid}, 0, 0) + (0, 0, C_z). \tag{A.2}$$

After converting an image $(Y_{mid}, 0, 0)$ from XYZ space to sRGB space, we obtain the image $(R_{Ymid}, G_{Ymid}, B_{Ymid})$. Similarly, converting an image $(0, 0, C_z)$ from XYZ space to sRGB space, we obtain the image (R_{Cz}, G_{Cz}, B_{Cz}).
Therefore,

$$(R_{Cz}^{FULL}, G_{Cz}^{FULL}, B_{Cz}^{FULL}) = \triangle(R_{Cz}, G_{Cz}, B_{Cz}) + (R_{Ymid}, G_{Ymid}, B_{Ymid}), \tag{A.3}$$

where

$$\begin{aligned} \triangle R_{Cz}(m,n) &= R_{Cz}(m,n) - R_{Ymid} \\ \triangle G_{Cz}(m,n) &= G_{Cz}(m,n) - G_{Ymid} \\ \triangle B_{Cz}(m,n) &= B_{Cz}(m,n) - B_{Ymid}. \end{aligned} \tag{A.4}$$

Since the range of RGB_{FULL} should be from 0 to 1 for each channel, we define the new parameter α such that

$$(\tilde{R}_{Cz}^{FULL}, \tilde{G}_{Cz}^{FULL}, \tilde{B}_{Cz}^{FULL}) = \alpha \triangle (R_{Cz}, G_{Cz}, B_{Cz}) + (R_{Ymid}, G_{Ymid}, B_{Ymid}). \tag{A.5}$$

We scale $\triangle(R_{Cz}, G_{Cz}, B_{Cz})$ by α, such that

$$0 \leq \alpha(\triangle R_{Cz}, \triangle G_{Cz}, \triangle B_{Cz}) + (R_{Ymid}, G_{Ymid}, B_{Ymid}) \leq 1. \tag{A.6}$$

Therefore, for each channel, we have:

$$\begin{aligned} R_{Cz}^{FULL}(m,n) &= \alpha_R \triangle R_{Cz}(m,n) + R_{Ymid} \\ G_{Cz}^{FULL}(m,n) &= \alpha_G \triangle G_{Cz}(m,n) + G_{Ymid} \\ B_{Cz}^{FULL}(m,n) &= \alpha_B \triangle B_{Cz}(m,n) + B_{Ymid}, \end{aligned} \tag{A.7}$$

such that

$$\begin{aligned} 0 &\leq \alpha_R \triangle R_{Cz}(m,n) + R_{Ymid} \leq 1 \\ 0 &\leq \alpha_G \triangle G_{Cz}(m,n) + G_{Ymid} \leq 1 \\ 0 &\leq \alpha_B \triangle B_{Cz}(m,n) + B_{Ymid} \leq 1, \end{aligned} \tag{A.8}$$

and

$$\alpha = \min(\alpha_R, \alpha_G, \alpha_B). \tag{A.9}$$

The following assumptions have been made:

$$\begin{gathered} 0 < R_{Ymid} < 1 \\ 0 < G_{Ymid} < 1 \\ 0 < B_{Ymid} < 1 \\ (\alpha_R, \alpha_G, \alpha_B) > 0. \end{gathered} \tag{A.10}$$

We start by looking at the red channel.

$$0 \leq \alpha_R \triangle R_{Cz}(m,n) + R_{Ymid} \leq 1. \tag{A.11}$$

Part 1. Given that

$$\alpha_R \bigtriangleup R_{Cz}(m) + R_{Ymid} \leq 1, \tag{A.12}$$

we have

$$\begin{cases} \alpha_R \leq \frac{1-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, & \text{if } \bigtriangleup R_{Cz}(m,n) > 0 \\ \alpha_R \geq \frac{1-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, & \text{if } \bigtriangleup R_{Cz}(m,n) < 0. \end{cases} \tag{A.13}$$

The second equation in the system is satisfied by assumptions stated in the beginning that

$$0 < R_{Ymid} < 1$$
$$\alpha_R > 0.$$

Therefore,

$$\alpha_R \geq \frac{1-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, \quad \text{if } \bigtriangleup R_{Cz}(m,n) < 0, \tag{A.14}$$

is always true. As for the first equation, we have

$$\alpha_R^+ \leq \min_{(m,n)}\{\frac{1-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, \quad \text{if } \bigtriangleup R_{Cz}(m,n) > 0\}. \tag{A.15}$$

Part 2. Given that

$$\alpha_R \bigtriangleup R_{Cz}(m) + R_{Ymid} \geq 0, \tag{A.16}$$

$$\tag{A.17}$$

we have

$$\begin{cases} \alpha_R \geq \frac{-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, & \text{if } \bigtriangleup R_{Cz}(m,n) > 0 \\ \alpha_R \leq \frac{-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, & \text{if } \bigtriangleup R_{Cz}(m,n) < 0. \end{cases} \tag{A.18}$$

The first equation in the system is satisfied by assumptions stated before that

$$0 < R_{Ymid} < 1$$
$$\alpha_R > 0.$$

Therefore

$$\alpha_R \geq \frac{-R_{Ymid}}{\bigtriangleup R_{Cz}(m,n)}, \quad \text{if } \bigtriangleup R_{Cz}(m,n) > 0. \tag{A.19}$$

As for the second equation, we have

$$\alpha_R^- \leq \min_{(m,n)}\{\frac{-R_{Ymid}}{\triangle R_{Cz}(m,n)}, \quad \text{if } \triangle R_{Cz}(m,n) < 0\}. \tag{A.20}$$

And lastly,

$$\alpha_R = \min(\alpha_R^+, \alpha_R^-). \tag{A.21}$$

α_G and α_B can be obtained similarly as α_R. And all three will be used to find

$$\alpha = \min(\alpha_R, \alpha_G, \alpha_B). \tag{A.22}$$

B. DETAILED DERIVATION OF THEORETICAL FOURIER ANALYSIS FOR A SINGLE SEPARATION AND A SUPERPOSITION OF TWO HALFTONES

B.1 CSFT of analog and digital rendered images

A continuous space halftone image $g(x)$ is defined as

$$\begin{aligned} g\left(\mathbf{x}\right) &= c\left(\mathbf{x}\right) * l\left(\mathbf{x}\right) \\ &= circ\left(\mathbf{M}\mathbf{x}\right) * \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{x} - R\mathbf{N}\mathbf{k}\right) \qquad \text{(B.1)} \\ &= \sum_{\mathbf{k}\in Z^2} circ\left(\mathbf{M}(\mathbf{x} - R\mathbf{N}\mathbf{k}\right) \end{aligned}$$

where

$$c\left(\mathbf{x}\right) = circ\left(\mathbf{M}\mathbf{x}\right) = \begin{cases} 1, & \text{if } \sqrt{x^2+y^2} \le \frac{1}{2} \\ 0, & \text{otherwise} \end{cases} \qquad \text{(B.2)}$$

with its corresponding CSFT

$$C\left(\mathbf{u}\right) = jinc\left(\mathbf{M}^{-T}\mathbf{u}\right) \qquad \text{(B.3)}$$

and $l\left(\mathbf{x}\right)$ is the impulse function defined as

$$l\left(\mathbf{x}\right) = \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{x} - R\mathbf{N}\mathbf{k}\right) \qquad \text{(B.4)}$$

where the matrix $\mathbf{N}$ is the periodicity matrix, and the parameter R is the horizontal and vertical distance between printer-addressable pixels in units of inches. The matrix $\mathbf{M}$ is a diagonal matrix, and it is defined as

$$\mathbf{M} = \begin{bmatrix} \frac{1}{d} & 0 \\ 0 & \frac{1}{d} \end{bmatrix}, \qquad \text{(B.5)}$$

where a parameter d is a diameter of a circular dot. The image $g\left(\mathbf{x}\right)$ is called **the analog rendering image.** Next, we compute the Continuous Space Fourier Transform (CSFT) of the continuous space halftone image with circular dot pattern $g\left(\mathbf{x}\right)$. The CSFT is obtained the following way:

$$\begin{aligned}
G\left(\mathbf{u}\right) &= CSFT(g\left(\mathbf{x}\right)) \\
&= CSFT(c\left(\mathbf{x}\right) * l\left(\mathbf{x}\right)) \\
&= CSFT(circ\left(\mathbf{M}\mathbf{x}\right) * \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{x} - R\mathbf{N}\mathbf{k}\right)) \\
&= \frac{jinc\left(\mathbf{M}^{-T}\mathbf{u}\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|} \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u} - \frac{1}{R}\mathbf{N}^{-\mathbf{T}}\mathbf{k}\right) \\
&= \frac{C\left(\mathbf{u}\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|} \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u} - \frac{1}{R}\mathbf{N}^{-\mathbf{T}}\mathbf{k}\right)
\end{aligned}$$

where the CSFT of a circ function is a jinc function which is defined as

$$jinc\left(\mathbf{u}\right) = \frac{J_1\left(\pi\sqrt{u^2+v^2}\right)}{2\sqrt{u^2+v^2}}, \tag{B.6}$$

and J_1 is the Bessel function of the first kind with order 1.

Additional derivation: Assume $g\left(x\right)$ is given as:

$$g\left(x\right)=rep_{X}\left[\delta\left(x\right)\right] \tag{B.7}$$

Its corresponding CSFT has a form:

$$G\left(u\right)=\frac{1}{X}comb\left[1\right] \tag{B.8}$$

Now, assume $g\left(x\right)$ is an impulse train given as:

$$g\left(x\right)=\sum_{k\in Z}\delta\left(x-k\bar{X}\right) \tag{B.9}$$

Its corresponding CSFT has a form:

$$G\left(u\right)=\frac{1}{\bar{X}}\sum_{k\in Z}\delta\left(u-\frac{k}{\bar{X}}\right) \tag{B.10}$$

We can derive $g\left(x\right)$ as following:

$$\begin{aligned}
g\left(x\right) &= \int_{-\infty}^{\infty}G\left(u\right)\mathrm{e}^{j2\pi ux}\mathrm{du} && \text{(B.11)}\\
&= \int_{-\infty}^{\infty}\frac{1}{\bar{X}}\sum_{k\in Z}\delta\left(u-\frac{k}{\bar{X}}\right)\mathrm{e}^{j2\pi ux}\mathrm{du}\\
&= \frac{1}{\bar{X}}\sum_{k\in Z}\int_{-\infty}^{\infty}\delta\left(u-\frac{k}{\bar{X}}\right)\mathrm{e}^{j2\pi ux}\mathrm{du} && \text{(B.12)}\\
&= \frac{1}{\bar{X}}\sum_{k\in Z}\mathrm{e}^{j2\pi\frac{k}{\bar{X}}x} && \text{(B.13)}
\end{aligned}$$

Therefore, we have:

$$g\left(x\right)=\sum_{k\in Z}\delta\left(x-k\bar{X}\right)=\frac{1}{\bar{X}}\sum_{k\in Z}\mathrm{e}^{j2\pi\frac{k}{\bar{X}}x} \tag{B.14}$$

and for our derivation, we have:

$$\sum_{k\in Z^2}\mathrm{e}^{-j2\pi u^TR\mathbf{N}k}=\frac{1}{R^2\left|\det\mathbf{N}\right|}\sum_{k\in Z^2}\delta\left(u-\frac{1}{R}\mathbf{N}^{-T}k\right) \tag{B.15}$$

In order to obtain the digital rendering form of the continuous space halftone image $g\left(\mathbf{x}\right)$, we sample $g\left(\mathbf{x}\right)$ with the printer lattice, and obtain $g_s\left(\mathbf{x}\right)$:

$$g_s\left(\mathbf{x}\right) = comb_{RR}\left[g\left(\mathbf{x}\right)\right] \tag{B.16}$$

And, finally we convolve $g_s\left(\mathbf{x}\right)$ with the printer dot profile function defined as $p\left(\mathbf{x}\right)$, where

$$p\left(\mathbf{x}\right) = rect\left(\frac{\mathbf{x}}{R}\right) \tag{B.17}$$

and, therefore

$$P\left(\mathbf{u}\right) = R^2 sinc\left(R\mathbf{u}\right) \tag{B.18}$$

The resulted image is defined as $g_p\left(\mathbf{x}\right)$:

$$\begin{aligned} g_p\left(\mathbf{x}\right) &= g_s\left(\mathbf{x}\right) * p\left(\mathbf{x}\right) \qquad \text{(B.19)} \\ &= comb_{RR}\left[g\left(\mathbf{x}\right)\right] * rect\left(\frac{\mathbf{x}}{R}\right). \end{aligned}$$

The image $g_{\text{printer}}\left(\mathbf{x}\right)$ is **the digital rendering image** of a continuous space halftone with circular dot pattern $g\left(\mathbf{x}\right)$. The CSFT of $g_{\text{printer}}\left(\mathbf{x}\right)$ is computed the following way:

$$\begin{aligned} G_p\left(\mathbf{u}\right) &= \frac{1}{R^2} rep_{\frac{1}{R}\frac{1}{R}}\left[G\left(\mathbf{u}\right)\right] P\left(\mathbf{u}\right) \qquad \text{(B.20)} \\ &= \left(\frac{1}{R^2} rep_{\frac{1}{R}\frac{1}{R}}\left[G\left(\mathbf{u}\right)\right]\right)\left(R^2 sinc\left(R\mathbf{u}\right)\right) \\ &= sinc\left(R\mathbf{u}\right)\sum_{\mathbf{m}\in Z^2} G\left(\mathbf{u}-\mathbf{V}\mathbf{m}\right) \\ &= sinc\left(R\mathbf{u}\right)\sum_{\mathbf{m}\in Z^2}\frac{jinc\left(\mathbf{M}^{-T}\left(\mathbf{u}-\mathbf{V}\mathbf{m}\right)\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\mathbf{V}\mathbf{m}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right), \end{aligned}$$

where the matrix $\mathbf{V}$ is defined as

$$\mathbf{V} = \begin{bmatrix} \frac{1}{R} & 0 \\ 0 & \frac{1}{R} \end{bmatrix} \tag{B.21}$$

and R is the horizontal and vertical distance between printer addressable pixels in inches.

In short, we derived **the two main CSFT functions** for analog rendering image $g\left(\mathbf{x}\right)$ and digital rendering image $g_{\text{printer}}\left(\mathbf{x}\right)$:

$$\begin{aligned} G\left(\mathbf{u}\right) &= \frac{jinc\left(\mathbf{M}^{-T}\mathbf{u}\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\frac{1}{R}\mathbf{N^{-T}k}\right) \qquad \text{(B.22)} \\ &= \frac{C\left(\mathbf{u}\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\frac{1}{R}\mathbf{N^{-T}k}\right) \end{aligned}$$

and

$$G_p\left(\mathbf{u}\right) = sinc\left(R\mathbf{u}\right)\sum_{\mathbf{m}\in Z^2}\frac{jinc\left(\mathbf{M}^{-T}\left(\mathbf{u}-\mathbf{Vm}\right)\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\mathbf{Vm}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right) \qquad \text{(B.23)}$$

In order to obtain the perceived digital rendered continuous space halftone images, we define $\tilde{g}_{\text{printer}}\left(\mathbf{x}\right)$ as

$$\tilde{g}_{\text{printer}}\left(\mathbf{x}\right) = g_{\text{printer}}\left(\mathbf{x}\right) * h_{HVS}\left(\mathbf{x}\right). \qquad \text{(B.24)}$$

Thus, the CSFT of a perceived digital rendered continuous space halftone image with circular dot pattern $\tilde{g}_{\text{printer}}\left(\mathbf{x}\right)$ is

$$\begin{aligned} \tilde{G}_p\left(\mathbf{u}\right) &= G_p\left(\mathbf{u}\right)H_{HVS}\left(\mathbf{u}\right) \qquad \text{(B.25)} \\ &= H_{HVS}\left(\mathbf{u}\right)sinc\left(R\mathbf{u}\right)\sum_{\mathbf{m}\in Z^2}\frac{jinc\left(\mathbf{M}^{-T}\left(\mathbf{u}-\mathbf{Vm}\right)\right)}{R^2\left|\det\left(\mathbf{M}\right)\det\left(\mathbf{N}\right)\right|}\sum_{\mathbf{k}\in Z^2}\delta\left(\mathbf{u}-\mathbf{Vm}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right) \end{aligned}$$

B.2 Multiplication model in reflectance for a superposition of 2 halftones

We start by defining a function $g_i(\mathbf{x})$

$$g_i(\mathbf{x}) = \begin{cases} 1, & \text{colorant is present} \\ 0, & \text{otherwise} \end{cases} \tag{B.26}$$

and for 2 colorants, we define their corresponding reflectance functions as:

$$r_i(\mathbf{x}) = 1 - g_i(\mathbf{x}) \tag{B.27}$$

Therefore, we have:

$$\begin{aligned} r_1(\mathbf{x}) &= 1 - g_1(\mathbf{x}), \\ r_2(\mathbf{x}) &= 1 - g_2(\mathbf{x}). \end{aligned} \tag{B.28}$$

Next, we multiply the two functions together, and lastly convert them back into absorptance:

$$\begin{aligned} r_{total}(\mathbf{x}) &= r_1(\mathbf{x})\, r_2(\mathbf{x}), \\ g_{total}(\mathbf{x}) &= 1 - r_{total}(\mathbf{x}) = 1 - (1 - g_1(\mathbf{x}))(1 - g_2(\mathbf{x})) \\ &= g_1(\mathbf{x}) + g_2(\mathbf{x}) - g_1(\mathbf{x})\, g_2(\mathbf{x}) \end{aligned} \tag{B.29}$$

The value of $g_{total}(\mathbf{x})$ at each pixel is equal to one when there exists any colorant at the pixel location, or zero otherwise.
Therefore,

$$g_{total}(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x}) - g_1(\mathbf{x})\, g_2(\mathbf{x}). \tag{B.30}$$

In frequency domain, we have

$$G_{total}(\mathbf{u}) = G_1(\mathbf{u}) + G_2(\mathbf{u}) - G_1(\mathbf{u}) * *G_2(\mathbf{u}) \tag{B.31}$$

where $**$ denotes the two-dimensional convolution.

In order to understand Eq. B.31, we start by looking at Eq. B.23, which is the CSFT of a digital-rendered image

$$G_p(\mathbf{u}) = sinc(R\mathbf{u}) \sum_{\mathbf{m}\in Z^2} \frac{jinc\left(\mathbf{M}^{-T}(\mathbf{u}-\mathbf{Vm})\right)}{R^2 \left|\det(\mathbf{M})\det(\mathbf{N})\right|} \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right) \tag{B.32}$$

$$= P(\mathbf{u}) \sum_{\mathbf{m}\in Z^2} \frac{C(\mathbf{u}-\mathbf{Vm})}{R^2 \left|\det(\mathbf{M})\det(\mathbf{N})\right|} \sum_{\mathbf{k}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm}-\frac{1}{R}\mathbf{N}^{-T}\mathbf{k}\right)$$

Since now we have 2 different colorants, we have different M and N parameters for each screen. In order to simplify the derivation of the formula we define the following parameters:

$$A_i = \frac{1}{R^2 \left|\det(\mathbf{M_i})\det(\mathbf{N_i})\right|} \tag{B.33}$$

Therefore, the simplified CSFT functions look like:

$$G_1(\mathbf{u}) = A_1 P_1(\mathbf{u}) \sum_{\mathbf{m_1}\in Z^2} C_1(\mathbf{u}-\mathbf{Vm_1}) \sum_{\mathbf{k_1}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm_1}-\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right) \tag{B.34}$$

and

$$G_2(\mathbf{u}) = A_2 P_2(\mathbf{u}) \sum_{\mathbf{m_2}\in Z^2} C_2(\mathbf{u}-\mathbf{Vm_2}) \sum_{\mathbf{k_2}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm_2}-\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \tag{B.35}$$

B.2.1 Derivation of $G_{total}(u)$

According to Eq. B.31, we have

$$G_{total}\left(\mathbf{u}\right) = G_1\left(\mathbf{u}\right) + G_2\left(\mathbf{u}\right) - G_1\left(\mathbf{u}\right) * * G_2\left(\mathbf{u}\right) \tag{B.36}$$

Based on Eqs. B.34 and B.35,

$$\begin{aligned} G_1\left(\mathbf{u}\right) + G_2\left(\mathbf{u}\right) = & \left[A_1 P_1\left(\mathbf{u}\right) \sum_{\mathbf{m_1} \in Z^2} C_1\left(\mathbf{u} - \mathbf{V}\mathbf{m_1}\right) \sum_{\mathbf{k_1} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m_1} - \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right] \\ & + \left[A_2 P_2\left(\mathbf{u}\right) \sum_{\mathbf{m_2} \in Z^2} C_2\left(\mathbf{u} - \mathbf{V}\mathbf{m_2}\right) \sum_{\mathbf{k_2} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m_2} - \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right] \end{aligned} \tag{B.37}$$

and

$$\begin{aligned} G_1\left(\mathbf{u}\right) * * G_2\left(\mathbf{u}\right) = & \left[A_1 P_1\left(\mathbf{u}\right) \sum_{\mathbf{m_1} \in Z^2} C_1\left(\mathbf{u} - \mathbf{V}\mathbf{m_1}\right) \sum_{\mathbf{k_1} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m_1} - \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right] \\ & * * \left[A_2 P_2\left(\mathbf{u}\right) \sum_{\mathbf{m_2} \in Z^2} C_2\left(\mathbf{u} - \mathbf{V}\mathbf{m_2}\right) \sum_{\mathbf{k_2} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m_2} - \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right] \\ & = \left[\sum_{\mathbf{m_1} \in Z^2} \sum_{\mathbf{k_1} \in Z^2} A_1 P_1\left(\mathbf{u}\right) C_1\left(\mathbf{u} - \mathbf{V}\mathbf{m_1}\right) \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m_1} - \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right] \\ & * * \left[\sum_{\mathbf{m_2} \in Z^2} \sum_{\mathbf{k_2} \in Z^2} A_2 P_2\left(\mathbf{u}\right) C_2\left(\mathbf{u} - \mathbf{V}\mathbf{m_2}\right) \delta\left(\mathbf{u} - \mathbf{V}\mathbf{m_2} - \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right] \end{aligned} \tag{B.38}$$

In order to simplify the calculations, we define:

$$\begin{aligned} F_1(\mathbf{u}) &= P_1(\mathbf{u})\, C_1(\mathbf{u} - \mathbf{V}\mathbf{m_1}) \\ F_2(\mathbf{u}) &= P_2(\mathbf{u})\, C_2(\mathbf{u} - \mathbf{V}\mathbf{m_2}) \\ \mathbf{u_1} &= \mathbf{V}\mathbf{m_1} + \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1} \\ \mathbf{u_2} &= \mathbf{V}\mathbf{m_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2} \end{aligned} \tag{B.39}$$

Therefore, with the use of Sifting Property, we have

$$\begin{aligned} G_1(\mathbf{u}) ** G_2(\mathbf{u}) &= \left[\sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2} A_1 F_1(\mathbf{u})\, \delta(\mathbf{u} - \mathbf{u_1})\right] \\ &** \left[\sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} A_2 F_2(\mathbf{u})\, \delta(\mathbf{u} - \mathbf{u_2})\right] \\ &= \int_{-\infty}^{\infty} \sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2} A_1 F_1(\boldsymbol{\mu})\, \delta(\boldsymbol{\mu} - \mathbf{u_1}) \\ &\times \sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} A_2 F_2(\mathbf{u} - \boldsymbol{\mu})\, \delta(\mathbf{u} - \boldsymbol{\mu} - \mathbf{u_2})\, \mathrm{d}\boldsymbol{\mu} \\ &= \sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2}\sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} \int_{-\infty}^{\infty} A_1 F_1(\boldsymbol{\mu})\, \delta(\boldsymbol{\mu} - \mathbf{u_1}) \\ &\times A_2 F_2(\mathbf{u} - \boldsymbol{\mu})\, \delta(\mathbf{u} - \boldsymbol{\mu} - \mathbf{u_2})\, \mathrm{d}\boldsymbol{\mu} \\ &= \sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2}\sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} A_1 F_1(\mathbf{u_1})\, A_2 F_2(\mathbf{u} - \mathbf{u_1})\, \delta(\mathbf{u} - \mathbf{u_1} - \mathbf{u_2}) \\ &= \sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2}\sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} A_1 A_2 F_1(\mathbf{u_1})\, F_2(\mathbf{u_2})\, \delta(\mathbf{u} - (\mathbf{u_1} + \mathbf{u_2})) \end{aligned} \tag{B.40}$$

where

$$\begin{aligned}
\mathbf{u_1} &= \mathbf{V}\mathbf{m_1} + \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1} \\
\mathbf{u_2} &= \mathbf{V}\mathbf{m_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2} \\
A_1 &= \frac{1}{R^2\left|\det\left(\mathbf{M_1}\right)\det\left(\mathbf{N_1}\right)\right|} \\
A_2 &= \frac{1}{R^2\left|\det\left(\mathbf{M_2}\right)\det\left(\mathbf{N_2}\right)\right|}
\end{aligned} \tag{B.41}$$

and

$$\begin{aligned}
F_1\left(\mathbf{u_1}\right) &= P_1\left(\mathbf{u}\right)C_1\left(\mathbf{u}-\mathbf{V}\mathbf{m_1}\right) \\
F_2\left(\mathbf{u_2}\right) &= P_2\left(\mathbf{u}\right)C_2\left(\mathbf{u}-\mathbf{V}\mathbf{m_2}\right) \\
P_1\left(\mathbf{u}\right) &= sinc\left(R\mathbf{u}\right) \\
P_2\left(\mathbf{u}\right) &= sinc\left(R\mathbf{u}\right)
\end{aligned} \tag{B.42}$$

Due to the fact that

$$\begin{aligned}
A_1A_2 &= \frac{1}{R^2\left|\det\left(\mathbf{M_1}\right)\det\left(\mathbf{N_1}\right)\right|}\frac{1}{R^2\left|\det\left(\mathbf{M_2}\right)\det\left(\mathbf{N_2}\right)\right|} \\
&= \frac{1}{R^4\left|\det\left(\mathbf{M_1}\mathbf{N_1}\mathbf{M_2}\mathbf{N_2}\right)\right|}
\end{aligned} \tag{B.43}$$

and

$$\begin{aligned}
F_1\left(\mathbf{u_1}\right) &= P_1\left(\mathbf{u_1}\right)C_1\left(\mathbf{u_1}-\mathbf{V}\mathbf{m_1}\right) \\
&= sinc\left(R\left(\mathbf{V}\mathbf{m_1}+\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right)C_1\left(\mathbf{V}\mathbf{m_1}+\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}-\mathbf{V}\mathbf{m_1}\right) \\
&= sinc\left(R\left(\mathbf{V}\mathbf{m_1}+\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right)C_1\left(\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right) \\
&= sinc\left(R\left(\mathbf{V}\mathbf{m_1}+\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right)jinc\left(\mathbf{M}^{-T}\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)
\end{aligned} \tag{B.44}$$

and

$$\begin{aligned} F_2\left(\mathbf{u_2}\right) &= P_2\left(\mathbf{u_2}\right) C_2\left(\mathbf{u_2} - \mathbf{Vm_2}\right) \\ &= sinc\left(R\left(\mathbf{Vm_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right) C_1\left(\mathbf{Vm_2} + \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_2} - \mathbf{Vm_2}\right) \\ &= sinc\left(R\left(\mathbf{Vm_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right) C_2\left(\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k}_2\right) \\ &= sinc\left(R\left(\mathbf{Vm_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right) jinc\left(\mathbf{M}^{-T}\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \end{aligned} \tag{B.45}$$

and

$$\begin{aligned} \delta\left(\mathbf{u} - \left(\mathbf{u_1} + \mathbf{u_2}\right)\right) &= \delta\left(\mathbf{u} - \left(\left(\mathbf{Vm_1} + \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right) + \left(\mathbf{Vm_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right)\right) \\ &= \delta\left(\mathbf{u} - \mathbf{Vm_1} - \mathbf{Vm_2} - \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1} - \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \end{aligned} \tag{B.46}$$

The half of the equation $G_1\left(\mathbf{u}\right) * *G_2\left(\mathbf{u}\right)$ has the following form:

$$\begin{aligned} G_1\left(\mathbf{u}\right) * *G_2\left(\mathbf{u}\right) &= \frac{1}{R^4\left|\det\left(\mathbf{M_1N_1M_2N_2}\right)\right|} \\ &\times \sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2} sinc\left(R\left(\mathbf{Vm_1} + \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right) jinc\left(\mathbf{M}^{-T}\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right) \\ &\times \sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} sinc\left(R\left(\mathbf{Vm_2} + \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right) jinc\left(\mathbf{M}^{-T}\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \\ &\times \delta\left(\mathbf{u} - \mathbf{Vm_1} - \mathbf{Vm_2} - \frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1} - \frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \end{aligned} \tag{B.47}$$

and based on Eq. B.37, another half of the equation $G_1(\mathbf{u}) + G_2(\mathbf{u})$ has the following form:

$$G_1(\mathbf{u}) + G_2(\mathbf{u}) = \left[A_1 P_1(\mathbf{u}) \sum_{\mathbf{m_1} \in Z^2} C_1(\mathbf{u} - \mathbf{V m_1}) \sum_{\mathbf{k_1} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V m_1} - \frac{1}{R} \mathbf{N_1}^{-T} \mathbf{k_1} \right) \right] \tag{B.48}$$

$$\begin{aligned}
&+ \left[A_2 P_2(\mathbf{u}) \sum_{\mathbf{m_2} \in Z^2} C_2(\mathbf{u} - \mathbf{V m_2}) \sum_{\mathbf{k_2} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V m_2} - \frac{1}{R} \mathbf{N_2}^{-T} \mathbf{k_2} \right) \right] \\
&= \frac{1}{R^2 \left| \det(\mathbf{M_1}) \det(\mathbf{N_1}) \right|} sinc(R\mathbf{u}) \sum_{\mathbf{m_1} \in Z^2} jinc\left(\mathbf{M_1}^{-T} (\mathbf{u} - \mathbf{V m_1}) \right) \\
&\times \sum_{\mathbf{k_1} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V m_1} - \frac{1}{R} \mathbf{N_1}^{-T} \mathbf{k_1} \right) \\
&+ \frac{1}{R^2 \left| \det(\mathbf{M_2}) \det(\mathbf{N_2}) \right|} sinc(R\mathbf{u}) \sum_{\mathbf{m_2} \in Z^2} jinc\left(\mathbf{M_2}^{-T} (\mathbf{u} - \mathbf{V m_2}) \right) \\
&\times \sum_{\mathbf{k_2} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V m_2} - \frac{1}{R} \mathbf{N_2}^{-T} \mathbf{k_2} \right) \\
&= \frac{1}{R^2 \left| \det(\mathbf{M_1 N_1}) \right|} sinc(R\mathbf{u}) \sum_{\mathbf{m_1} \in Z^2} jinc\left(\mathbf{M_1}^{-T} (\mathbf{u} - \mathbf{V m_1}) \right) \\
&\times \sum_{\mathbf{k_1} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V m_1} - \frac{1}{R} \mathbf{N_1}^{-T} \mathbf{k_1} \right) \\
&+ \frac{1}{R^2 \left| \det(\mathbf{M_2 N_2}) \right|} sinc(R\mathbf{u}) \sum_{\mathbf{m_2} \in Z^2} jinc\left(\mathbf{M_2}^{-T} (\mathbf{u} - \mathbf{V m_2}) \right) \\
&\times \sum_{\mathbf{k_2} \in Z^2} \delta\left(\mathbf{u} - \mathbf{V m_2} - \frac{1}{R} \mathbf{N_2}^{-T} \mathbf{k_2} \right)
\end{aligned}$$

Finally, the entire equation, which was previously given as

$$G_{total}(\mathbf{u}) = G_1(\mathbf{u}) + G_2(\mathbf{u}) - G_1(\mathbf{u}) * * G_2(\mathbf{u}) \tag{B.49}$$

can now be described as

$$
\begin{aligned}
G_{total}\left(\mathbf{u}\right) = {} & \frac{1}{R^2\left|\det\left(\mathbf{M_1N_1}\right)\right|} sinc\left(R\mathbf{u}\right) \sum_{\mathbf{m_1}\in Z^2} jinc\left(\mathbf{M_1}^{-T}\left(\mathbf{u}-\mathbf{Vm_1}\right)\right) \\
& \times \sum_{\mathbf{k_1}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm_1}-\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right) \\
& + \frac{1}{R^2\left|\det\left(\mathbf{M_2N_2}\right)\right|} sinc\left(R\mathbf{u}\right) \sum_{\mathbf{m_2}\in Z^2} jinc\left(\mathbf{M_2}^{-T}\left(\mathbf{u}-\mathbf{Vm_2}\right)\right) \\
& \times \sum_{\mathbf{k_2}\in Z^2} \delta\left(\mathbf{u}-\mathbf{Vm_2}-\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \\
& - \frac{1}{R^4\left|\det\left(\mathbf{M_1N_1M_2N_2}\right)\right|} \\
& \times \sum_{\mathbf{m_1}\in Z^2}\sum_{\mathbf{k_1}\in Z^2} sinc\left(R\left(\mathbf{Vm_1}+\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right)\right) jinc\left(\mathbf{M}^{-T}\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}\right) \\
& \times \sum_{\mathbf{m_2}\in Z^2}\sum_{\mathbf{k_2}\in Z^2} sinc\left(R\left(\mathbf{Vm_2}+\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)\right) jinc\left(\mathbf{M}^{-T}\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right) \\
& \times \delta\left(\mathbf{u}-\mathbf{Vm_1}-\mathbf{Vm_2}-\frac{1}{R}\mathbf{N_1}^{-T}\mathbf{k_1}-\frac{1}{R}\mathbf{N_2}^{-T}\mathbf{k_2}\right)
\end{aligned}
\tag{B.50}
$$

B.3 Multiplication model in reflectance for a superposition of 3 halftones

We start by defining a function $g_i(\mathbf{x})$

$$g_i(\mathbf{x}) = \begin{cases} 1, & \text{colorant is present} \\ 0, & \text{otherwise} \end{cases} \tag{B.51}$$

and for 3 colorants, we define their corresponding reflectance functions as:

$$r_i(\mathbf{x}) = 1 - g_i(\mathbf{x}) \tag{B.52}$$

Therefore, we have:

$$\begin{aligned} r_1(\mathbf{x}) &= 1 - g_1(\mathbf{x}), \\ r_2(\mathbf{x}) &= 1 - g_2(\mathbf{x}), \\ r_3(\mathbf{x}) &= 1 - g_3(\mathbf{x}). \end{aligned} \tag{B.53}$$

Next, we multiply the three functions together, and convert the result back into absorptance:

$$\begin{aligned} r_{total}(\mathbf{x}) &= r_1(\mathbf{x})\, r_2(\mathbf{x})\, r_3(\mathbf{x}), \\ g_{total}(\mathbf{x}) &= 1 - r_{total}(\mathbf{x}) = 1 - (1 - g_1(\mathbf{x}))(1 - g_2(\mathbf{x}))(1 - g_3(\mathbf{x})) \\ &= g_1(\mathbf{x}) + g_2(\mathbf{x}) + g_3(\mathbf{x}) - g_1(\mathbf{x})\, g_2(\mathbf{x}) - g_1(\mathbf{x})\, g_3(\mathbf{x}) - g_2(\mathbf{x})\, g_3(\mathbf{x}) \\ &+ g_1(\mathbf{x})\, g_2(\mathbf{x})\, g_3(\mathbf{x}). \end{aligned} \tag{B.54}$$

The value of $g_{total}(\mathbf{x})$ at each pixel is equal to one when there exists any colorant at the pixel location, or zero otherwise.

Therefore,

$$\begin{aligned} g_{total}(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x}) + g_3(\mathbf{x}) - g_1(\mathbf{x})\, g_2(\mathbf{x}) - g_1(\mathbf{x})\, g_3(\mathbf{x}) - g_2(\mathbf{x})\, g_3(\mathbf{x}) \\ + g_1(\mathbf{x})\, g_2(\mathbf{x})\, g_3(\mathbf{x}). \end{aligned} \tag{B.55}$$

VITA

VITA

Altyngul Jumabayeva received B.S. from Purdue University in 2014. She is working on her Ph.D. degree at Purdue University, IN, USA. Her research interests mainly focus on image processing and halftoning.