The Dissertation Committee for HAMOOD-UR REHMAN
certifies that this is the approved version of the following dissertation:

ARTIFACT ASSESSMENT, GENERATION, AND ENHANCEMENT OF VIDEO HALFTONES

Committee:

BRIAN L. EVANS, Supervisor

ALAN C. BOVIK

GUSTAVO DE VECIANA

WILSON S. GEISLER

HARIS VIKALO

ARTIFACT ASSESSMENT, GENERATION, AND ENHANCEMENT OF VIDEO HALFTONES

by

HAMOOD-UR REHMAN, B.S.E.E., M.S.E.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2010

To Dr. Muhammad Jaffar-Ur Rehman and my parents.

Acknowledgments

First and foremost, I thank the Almighty for giving me the opportunity to write this dissertation. After that, I would like to express my highest gratitude for the members of my family, without whose unrelenting support and encouragement, this dissertation would not have been possible. They have been my superb teachers. For all their contributions, I am indebted to them forever.

The profoundness of the positive impact that my supervisor, Prof. Brian L. Evans, has had on my life is indescribable. I attempt to say a few words nevertheless. Prof. Evans has encouraged me tremendously in my research endeavors leading to the writing of this dissertation. His dedication to his profession and his tireless efforts in helping students are truly remarkable. I have had the honor to have been advised by him at The University of Texas at Austin, at both undergraduate and graduate levels. I would like to thank him for his support, direction, advice, and long mentorship.

The members of my dissertation committee, Prof. Alan C. Bovik, Prof. Gustavo de Veciana, Prof. Wilson S. Geisler, and Prof. Haris Vikalo, have helped me improve my work through generously giving their time and valuable comments. I have learned a great deal from them through their excellent instruction in class and through the insightful discussions outside class. I would like to thank them for their keen guidance and tutelage.

I would like to acknowledge the role of all my teachers in my pursuit of knowledge. I am grateful to Prof. Edward J. Powers, Prof. John R. Cogdell, and Prof. Kathleen A. McKeon for their advice and inspiration. I am appreciative of Prof. Michael F. Becker, Prof. Robert B. McCann, and Prof. Baxter F. Womack for their support.

I appreciate the contribution of the volunteers who provided their subjective opinions on video quality. I would like to thank Marc Mignard, Jennifer Gille, and Clarence Chui for their eager support during my internship at Qualcomm MEMS Technologies in 2008. I am glad to have been in the company of my current and former colleagues in the Embedded Signal Processing Laboratory. I would like to express my appreciation for my supportive colleagues in the Wireless Networking and Communications Group. I am thankful of Melanie Gulick for her kind assistance throughout my graduate studies at The University of Texas at Austin.

ARTIFACT ASSESSMENT, GENERATION, AND ENHANCEMENT OF VIDEO HALFTONES

Publication No. _____

HAMOOD-UR REHMAN, Ph.D.
The University of Texas at Austin, 2010

Supervisor: BRIAN L. EVANS

With the advancement of display technology, consumers expect high quality display of image and video data. Many viewers are used to watching video content on high definition television and large screens. However, certain display technologies, such as several of those used in portable electronic books, are limited on resources such as the availability of number of bits per pixel (i.e. the bit-depth). Display of good or even acceptable perceptual quality video on these devices is a hard technical problem that a display designer must solve.

Video halftoning reduces the number of represented colors or gray levels for display on devices that are unable to render the video at full bit-depth. Bit-depth reduction results in visible spatial and temporal artifacts. The designer would want to choose the halftoning algorithm that reduces these artifacts while meeting the target platform constraints. These constraints include available bit-depth, spatial resolution, computational power, and desired frame rate. Perceptual quality assess-

ment techniques are useful in comparing different video halftoning algorithms that satisfy the constraints.

This dissertation develops a framework for the evaluation of two key temporal artifacts, flicker and dirty-window-effect, in medium frame rate binary video halftones generated from grayscale continuous-tone videos. The possible causes underlying these temporal artifacts are discussed. The framework is based on perceptual criteria and incorporates properties of the human visual system. The framework allows for independent assessment of each of the temporal artifacts.

This dissertation presents design of algorithms that generate medium frame rate binary halftone videos. The design of the presented video halftone generation algorithms benefits from the proposed temporal artifact evaluation framework and is geared towards reducing the visibility of temporal artifacts in the generated medium frame rate binary halftone videos.

This dissertation compares the relative power consumption associated with several medium frame rate binary halftone videos generated using different video halftone generation algorithms. The presented power performance analysis is generally applicable to bistable display devices.

This dissertation develops algorithms to enhance medium frame rate binary halftone videos by reducing flicker. The designed enhancement algorithms reduce flicker while attempting to constrain any resulting increase in perceptual degradation of the spatial quality of the halftone frames.

This dissertation develops algorithms to enhance medium frame rate binary

halftone videos by reducing dirty-window-effect. The enhancement algorithms reduce dirty-window-effect while attempting to constrain any resulting increase in perceptual degradation of the spatial quality of the halftone frames.

Finally, this dissertation proposes design of medium frame rate binary halftone video enhancement algorithms that attempt to reduce a temporal artifact, flicker or dirty-window-effect, under both spatial and temporal quality constraints. Temporal quality control is incorporated by using the temporal artifact assessment framework developed in this dissertation. The incorporation of temporal quality control, in the process of reducing flicker or dirty-window-effect, helps establish a balance between the two temporal artifacts in the enhanced video. At the same time, the spatial quality control attempts to constrain any increase in perceptual degradation of the spatial quality of the enhanced halftone frames.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The increasing affordability of high definition (HD) display technologies means that more consumers will be using the HD display devices. The main advantage of HD display technology is enhanced viewer experience. The HD display devices typically provide this advantage at the expense of increased demand on resources such as increased number of bits per picture element (pixel), higher spatial resolution, and limited restrictions on the use of power and computational resources. What happens when the display devices is, in fact, *low* on some or all of these resources? This is a scenario that is not too uncommon in the case of portable handheld display devices such as cell phone displays or portable electronic book readers. Display of media on low-resource devices is a challenge that becomes even more difficult due to the fact that the users of these low-resource display devices are many times used to watching image and video content on HD display devices with larger screen sizes.

The designer of limited resource display devices must ensure that the perceptual quality of media rendered on such devices is of a quality that is acceptable for all users. To meet this goal, the designer needs to identify the perceptual artifacts that result from limited resources, and figure out a way to minimize the visibility of these artifacts. Digital halftoning is a technology that deals with reducing the visibility

of artifacts that result from reduction in the number of bits available for each pixel (i.e. the bit-depth). Halftoning of a grayscale video to the minimum number of bits per pixel (i.e. 1 bit per pixel) can result in temporal artifacts. Two key temporal artifacts present in binary video halftones are: (1) flicker, and (2) dirty-window-effect. Some low-resource display devices can be more power efficient if the video displayed on these devices does not require high frame rate playback. An example of this kind of display device is a bistable display device that consumes relatively more power in switching the state of its pixels [1]. For such a device, higher frame rate could potentially mean more number of pixels switching state per second, thus, increasing the power usage.

This dissertation attempts to explore the problem of bit-depth reduction for video rendering at medium frame rates of 15 to 30 frames per second (fps). As will be explained in more detail later, the generation of good perceptual quality halftone videos at medium frame rates is even more challenging because the human visual system is very sensitive to temporal frequencies that correspond to such frame rates [2, 3].

There are several reasons for choosing medium frame rate (as opposed to low or high frame rate) videos for my research. Since handheld portable devices that require halftoning prior to video display are generally limited on resources, higher frame rates (i.e. more than 30 fps) may not be supported at all on these handheld devices. Even if a frame rate higher than 30 fps is supported, it may not be feasible to display the halftone video at the higher frame rate. For example, consider a conventional Liquid crystal display (LCD) device that has a periodic screen refresh

rate of say 60 Hz. On this device, 30 fps frame rate would mean that frame buffer data for each frame is refreshed twice on the LCD screen during every 1/30 second interval. In this particular case, synchronization of screen refresh time with frame buffer data refresh time is relatively not that critical. Now consider displaying the video at 60 fps on the same LCD device. In this particular case, the frame buffer needs to have the new frame data before each LCD screen refresh or update. A screen refresh rate higher than 60 Hz is needed to alleviate this problem. Synchronization of screen refresh with frame buffer data refresh is extremely critical in this case. On a typical conventional LCD screen embedded system, as will be discussed in Chapter 4, higher screen refresh rate will mean higher power consumption.

Another example of a system that might benefit from avoiding higher frame rate video display would be a bistable display device. Bistable display systems are explained in detail in Chapter 4. On a bistable display system, the display consumes power each time a pixel changes state. Retaining a pixel's state requires no power consumption, on the other hand. A higher frame rate on these systems could potentially mean more pixels changing their state per second. This, in turn, means higher power consumption for a bistable display device. On handheld multimedia devices, power is a scarce resource that must be conserved to increase battery life. Frame rates lower than 15 fps, on the other hand, might make the video appear not smooth enough. Lower frame rate videos might not capture quick motion very well. Medium frame rates (15 to 30 fps) offer an important trade-off on capturing motion in video sequences and reducing power consumption by the display device.

This chapter briefly introduces the reader to digital image and video ren-

dering. Some common display technologies are briefly touched upon. The problem of halftoning is introduced with general descriptions of image and video halftoning algorithms. Contributions of this dissertation are discussed followed by an outline of this dissertation.

1.1 Digital Image and Video Rendering

Allebach defines image rendering as a transformation operation that converts an image's device-independent representation to a device-dependent representation [4]. As will be discussed later in the current chapter, halftoning is an integral part of this transformation for many display devices. Device characteristics, the human viewer, the content of the image, as well as the system performance requirements are all important factors that need to be considered when rendering an image [4]. This section provides an introduction to the sampling and quantization operations that are involved in the display of image or video data on a digital display device. To give the reader a brief introduction to common display technologies, some image rendering technologies along with their advantages and disadvantages are also discussed at a very high level.

1.1.1 Sampling and Quantization

An image is a spatial distribution of irradiance [5], which can be described as a continuous function of two spatial variables (or coordinates). Similarly, video is a spatiotemporal distribution of irradiance. A video can be thought of as a sequence of images. The format of data (image or video) to be displayed is determined by the

image rendering system. For example, a digital display device can display only digital data. This requires that any analog data, acquired, for example, from an analog image acquisition device, needs to be converted to a digital representation that is suitable for the digital device. An analog representation of image has a (continuous) amplitude value associated with each (continuous) spatial coordinate. Conversion to digital form requires [6] : (1) Sampling the image in spatial coordinates to have a discrete picture element (pixel) addressing system, and (2) Sampling the amplitude values to have a discrete number of amplitude levels. Sampling the amplitude values is also known as quantization. Sampling in spatial coordinates determines the spatial resolution of the resulting digital image. Sampling strategy also determines the form of the pixel grid formed in the resulting digital image. Typically, pixels are arranged in a two-dimensional rectangular grid. Other forms of grids such as hexagonal [5, 7, 8] are also possible. Quantization determines the number of bits that can be represented by each pixel. For example, a grayscale image quantized to 8-bits per pixel can represent a total of 256 gray levels at each pixel location. Number of bits per pixel is also referred to as bit-depth. A bit-depth of 1 bit-per-pixel (bpp) results in a binary image. In the halftoning literature, a digital image or video having a higher bit-depth (for example, 8 bpp) is called a continuous-tone image or video. If the original or the continuous-tone digital image or video is quantized *further*, the resulting reduced bit-depth image or video is known as the halftone image or video.

1.1.2 Display Systems

The display system is an important component of the visual communication that is necessary for a user to perceive and understand the displayed information correctly. A display system needs to be easy and comfortable to use. In this section, typical requirements of a display system are introduced. Examples of popular display technologies are also presented. The section concludes with a general discussion on handheld devices.

1.1.2.1 Typical Requirements

There are several factors that can determine the suitability of a particular display system. Some of the most common factors include [9]:

- *Viewing Conditions:* User environment can impact the performance of a display system. Environmental conditions can include lighting conditions, typical viewing distance, and typical viewing angles etc.

- *Data and Application:* The data and application for which the display system will be used are important factors. For example, some viewing systems are better suited for video and graphics, while others are better for textual data only. Similarly, some viewing systems cannot display color data.

- *Operational Requirements:* These include technical characteristics such as reliability, affordability, portability, power requirements, weight and volume etc.

1.1.2.2 Some Examples of Display Technologies

In this section, I briefly discuss some of the digital image and video display systems that are commonly used. The list of display systems is not exhaustive and the discussion of each system is only at an introductory level. A more detailed treatment of the subject is beyond the scope of this report. Appropriate sources such as [9–11] can be consulted for an in-depth treatment of the subject.

Cathode ray tube (CRT) displays have been popular display systems for different types of data including image, video and alphanumeric data. They are relatively cheap, and produce high quality display of imagery [9]. The cathode ray tube generates an image-forming beam of electrons that is absorbed by phosphor on the display screen. The image-forming beam is modulated using the input image data to form the (output) image on the screen. A major disadvantage of a CRT display system is its relatively large size. At the time of writing of this report, due to the disadvantages of the CRT display systems, they have lost their popularity, especially, as other display system technologies have become more affordable.

Liquid crystal display (LCD) systems control the transmission or reflection of a light source by altering the optical path of light [9]. This altering of optical path is achieved by using electric field across the liquid crystal material. LCD systems consume less power and have become quite affordable in the recent years. Limited viewing angle is a problem with many LCD systems. With the use of thin-film transistors (TFT), some LCD systems can produce high-resolution displays [9]. An obvious advantage of LCD over CRT is the size. LCD systems are lighter, more portable, and consume less space.

Plasma display systems offer a durable, high memory, and high luminance and efficiency alternative. They are, however, expensive and consume relatively more power than most LCD systems [9].

Light-emitting diode (LED) display systems utilize single-crystal phosphor materials [9]. These displays are generally reliable since an LED element in the LED array can fail without affecting the rest of the display [9]. These displays are also not as temperature sensitive as some other display technologies are. LED systems also offer better viewing angles compared to LCD systems. They are, however, not as power efficient or cost effective.

Printing processes that rely on ink or toners represent an example of a binary display technology [11] and, therefore, require halftoning for rendering multiple gray levels. The three basic types of printing include letterpress, lithography, and gravure. The technologies differ in the way the printing plates are prepared and used. Unlike commercial printing processes that press ink onto the paper, ink-jet printers spray the ink [11]. Laser printers utilize electrophotograhpic process for printing. Dye-sublimation printers do not use halftoning [12] as they can control the variation of density of colorant [11].

1.1.2.3 Handheld Display Devices

In the recent years, the handheld display device industry has made quite a progress. The increased use of handheld devices by end-users to access multimedia will make power efficient devices more popular in the market. Some of the key requirements of a multimedia handheld device end-user might include:

1. *Portability*: Lighter handheld display devices have better potability. A user would prefer a mobile phone, that might also offer multimedia capability, to be easy to carry with him or her for at least most of the time.

2. *Battery Life*: Portable multimedia devices need to operate on batteries most of the time. The customer would prefer to not have to charge his or her portable handheld multimedia device too frequently.

3. *Display Quality*: A device used to access multimedia content needs to provide acceptable perceptual quality of image and video data displayed to the user. Watching a video with poor perceptual quality will not only be less entertaining to the end-user, it could also potentially tire the user sooner. For example, a video display system with unacceptable level of flicker could potentially cause eye fatigue.

The requirements discussed above make portable multimedia device design very challenging. A display technology that is portable might be constrained on resources such as available bit-depth, data processing and storage capabilities, and battery size. For devices whose frame rate is tied to power consumption [1], higher frame rates may not be possible. Power consumption of electronic paper (e-paper) technology based display systems is a lot less than the power consumption of Liquid Crystal Display (LCD) systems [1]. Bistable display devices [13–16] are examples of display systems that are very power efficient. This is so because these systems generally require power only to switch the state of a pixel, for example from "on" to "off" or vice versa. Retaining the state of a pixel typically requires little or no

power. Due to their high power efficiency, bistable display devices offer an attractive alternative to other display technologies used for handheld devices.

As discussed in the paragraph above, reduced bit-depth is a shortcoming that some resource constrained handheld devices must have to deal with. Quantization is an inevitable reality for such systems. Display systems that cannot support higher bit-depth must rely on good quantization techniques to minimize the perception of quantization artifacts. Digital halftoning is the process of converting a continuous-tone image or video with picture elements (pixels) at a higher bit-depth to one with lower bit-depth pixels. In the next section, I discuss the general concepts essential to understanding halftoning.

1.2 Halftoning

Halftoning is needed whenever the bit-depth capabilities of a display device, such as a printer or a monitor, are insufficient to display the continuous-tone data (i.e. image or video) at full bit-depth. An everyday example of the use of halftoning is a printer with only black ink. Although the printer can only produce black (and white) dots on the paper, halftoning enables it to print images that give the illusion of the presence of various gray levels. It is important to realize that although halftoning involves quantization, it is not merely scalar quantization with a fixed threshold. Instead, halftoning aims to utilize the properties of the human visual system (HVS) in the process of quantization to distribute quantization noise such that its visibility to a human observer is minimized. Figure 1.1 shows the Peppers continuous-tone image. Figure 1.2 shows the quantized Peppers image. Figure 1.10

Figure 1.1: Continuous-tone Peppers image.

a halftone of the Peppers image.

Digital image halftoning is sometimes also referred to as spatial dithering [7]. Analogously, digital video halftoning can also be referred to as spatiotemporal dithering [17]. The typical goal of a halftoning algorithm is to produce halftone data (i.e. image or video), in a computationally efficient manner, that can be rendered on the given display device such that the visibility of artifacts is minimized [18]. The

Figure 1.2: Peppers image quantized pixel-by-pixel using a fixed threshold at mid-gray.

perceptual quality of a halftone image or video depends on how well the quantization noise is hidden from the viewer. Halftone artifacts refer to the visible perceptual distortion in a halftone image or video. In the following section, I discuss the role of human visual system in the success of halftoning.

1.2.1 Human Visual System Models and Halftoning

What is it that causes a halftone image to give the illusion of a continuous-tone image? It is because of how the human visual system works. A human eye acts as a spatial low pass filter and this very characteristic makes halftoning work [19]. This low pass filtering operation causes blurring of the fine patterns created by halftoned pixels. The blurring causes perception of continuous-tone in the image. This is confirmed by the fact that an appropriate low pass filter used to model the human visual system and explicitly incorporated in the halftoning process can yield visually pleasing halftone patterns. All halftoning algorithms rely on the properties of the HVS. However, there is a certain class of halftoning algorithms that make explicit use of an HVS model. There algorithms are commonly known as model-based halftoning algorithms [20]. In the next two sections, the use of spatial and spatiotemporal models in halftoning applications is discussed.

1.2.1.1 Use of Spatial HVS Models in Halftoning

Although sophisticated models for HVS have been proposed [21], simple filter-based models have demonstrated good results for halftoning. Multichannel models [22–27] could also be used, if computational complexity is not an issue.

Simple models have, however, yielded a favorable trade-off between computational complexity and perceptual quality [28]. This is especially important in iterative halftoning techniques.

In halftoning, the HVS models are typically based on human contrast sensitivity function (CSF) [29]. The CSF describes visual sensitivity as a function of spatial frequency. If the HVS is modeled as a linear shift-invariant system, then the CSF can be used as an approximation of its frequency response. The HVS model can then be represented by a linear shift-invariant filter. Several HVS models have been considered for halftoning [19, 30–35]. The performance of different HVS models [36–39] has been evaluated for direct binary search (DBS) based halftoning in [40]. The models proposed by [36, 37] are bandpass. For the purpose of illustration, I briefly discuss the model proposed by Mannos and Sakrison [37]. This model proposes a frequency response $H_r(f_r)$ given by

$$H_r(f_r) = 2.6(0.0192 + 0.114 f_r) e^{-(0.114 f_r)^{1.1}}, \tag{1.1}$$

where the radial spatial frequency $f_r = \sqrt{f_x^2 + f_y^2}$ is calculated from the horizontal and vertical spatial frequencies f_x and f_y respectively. The unit of spatial frequency is cycles per degree (cpd). Note that degrees is the unit to specify the size of the viewed object formed at the retina of a human viewer. This angle is dependent on the size of the object and the viewing distance. To illustrate this concept, Figure 1.3 shows a viewer standing at a distance of D units of length from an object of vertical height L units of length. Then the vertical angle A, in units of radians, formed at

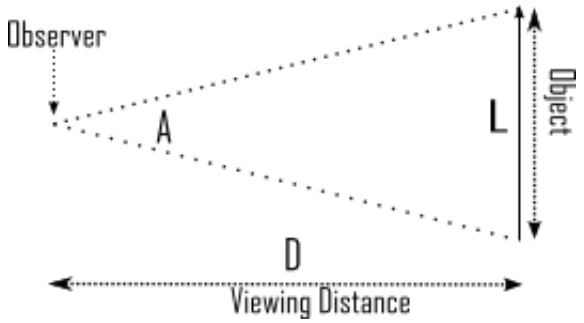


Figure 1.3: Degrees as a unit of viewed object's size. Observer is standing D units of length from an object of height L units of length. A is the angle subtended at the observer's eye.

the observer's eye is given by

$$A = 2\arctan(L/(2D)). \tag{1.2}$$

The angle A with small angle approximation can be expressed in degrees by [41]

$$A \approx 57.3(L/D). \tag{1.3}$$

Figure 1.4 shows a plot of (1.1). The frequency response is considered to represent the sensitivity of eye at various spatial frequencies [20]. Note that this model has bandpass characteristics and proposes a radially symmetric frequency response. It has been argued that HVS is relatively more sensitive to horizontal and vertical orientations [20, 42]. The angular dependence on contrast sensitivity is sometimes incorporated into the design of HVS filter [20, 39]. However, Allebach reports that with their direct binary search halftoning algorithm, pushing halftone energy in the frequency domain towards odd multiples of 45 degrees resulted in diagonal texture structure that was visually undesirable [4]. It has been argued in [43] that this

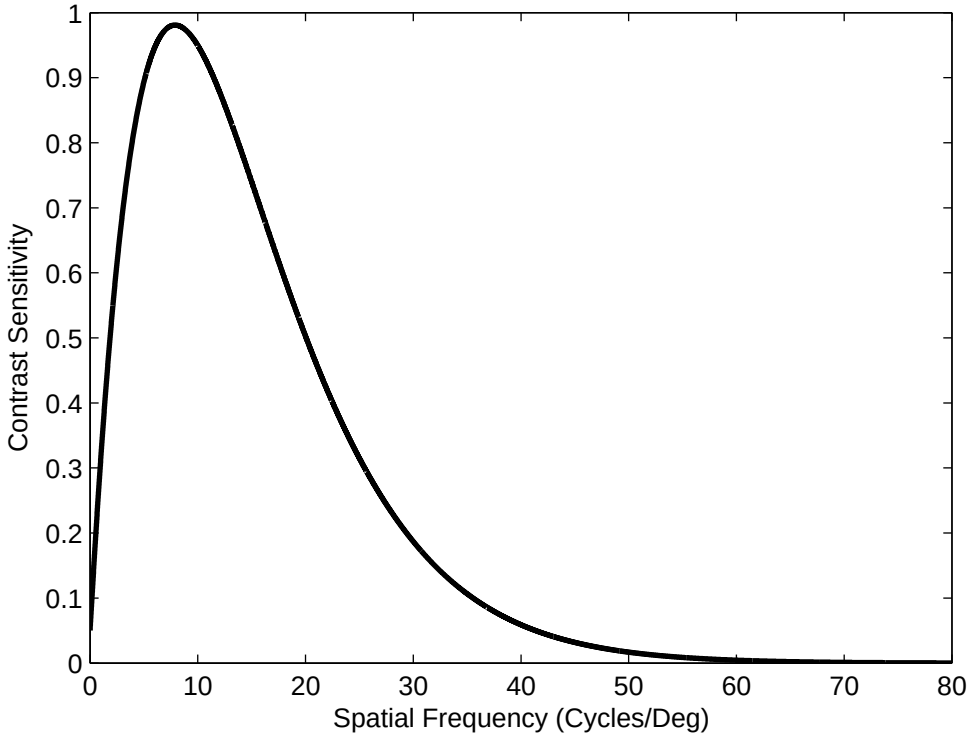


Figure 1.4: Contrast sensitivity function (CSF) for the human visual system (HVS) model proposed by Mannos and Sakrison.

was so because the experiments to determine the sensitivity of HVS to different orientations did not consider all possible stimuli.

Now I will consider an example of a low-pass frequency response. The frequency response, $H_r(f_r)$, for Nasanen's model is given by

$$H_r(f_r) = aL^b e^{-(f_r/[c\ln(L)+d])}, \tag{1.4}$$

where f_r is the radial spatial frequency, L is average luminance, and a, b, c, and d are constants. The unit of spatial frequency is cycles per degree (cpd). With

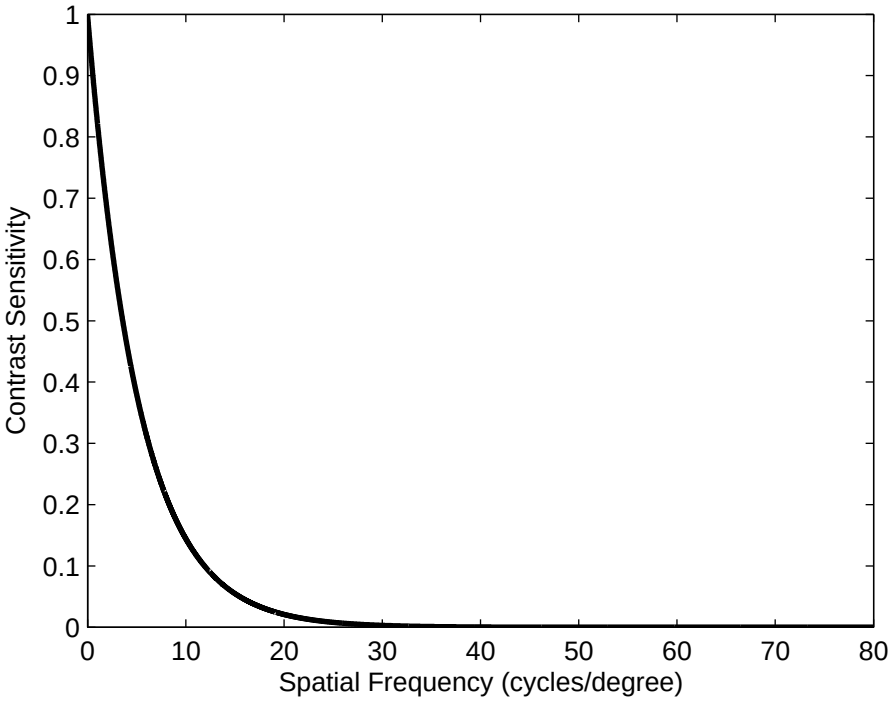


Figure 1.5: Normalized contrast sensitivity function (CSF) for the human visual system (HVS) model proposed by Nasanen.

$a = 131.6$, $b = 0.3188$, $c = 0.525$, $d = 3.91$, and $L = 11\ cd/m^2$, a plot of the normalized frequency response based on Nasanen's model is shown in Figure 1.5. Note that this model suggests a low-pass frequency response. It was shown by Mitsa and Varkur [34] that for quantitative evaluation of halftoning applications, low-pass CSF performs better than a bandpass CSF. In comparison to three other models [36, 37, 39], it was found in [40] that with DBS, Nasanen's model produced halftones with the best subjective quality. The spatial HVS model used in this dissertation will be based on Nasanen's model.

1.2.1.2 Use of Spatiotemporal HVS Models in Halftoning

I have discussed the use of spatial HVS models in image halftoning. I now discuss the use of spatiotemporal HVS models in video halftoning. For the more general case, Watson gives a thorough treatment of temporal aspects of human vision [3].

For video halftoning, spatiotemporal model suggested by Kelly [44] has been used in [45, 46]. This model exhibits non-separability of spatial and temporal characteristics of the HVS. The use of spatiotemporal model in video halftoning algorithms has proven to be effective at higher frame rates such as 60 frames per second (fps) only [45, 46]. With the use of spatiotemporal model, Atkins *et al.* [45] compared their results against the results of applying standard error diffusion algorithm [47] on each frame. They report that with the use of spatiotemporal model, the improvement in the halftone sequence was only observable at a frame rate of 60 frames per second (fps). The incorporation of spatiotemporal model showed no relative improvement at 30 fps. It was concluded that temporal averaging by HVS was not achieved at 30 fps [45]. Similar finding is reported by Hilgenberg *et al.* in [46]. At 30 fps their algorithm that used spatiotemporal model did not perform better than an algorithm that used only spatial model.

The published work discussed in the preceding paragraph suggests that for video halftoning applications, use of spatiotemporal model is beneficial for higher frame rates. However, at medium frame rates such as 30 fps, spatiotemporal model is not useful. Use of spatiotemporal model increases computational demands and introduces processing delay [48]. A better option is to use spatial model for HVS

in model-based video halftoning applications that produce videos to be rendered at medium frame rates (such as 15 to 30 fps). Use of spatial model has produced good quality halftone videos [49]. Consequently, in this dissertation, a spatial model for HVS will be utilized for medium frame rate (15 to 30 fps) video halftones.

1.2.2 Image Halftoning

This section provides a brief introduction to image halftoning. Image halftoning algorithms can be broadly divided into three categories [20]: (1) point algorithms, (2) neighborhood algorithms, and (3) iterative algorithms. I briefly discuss each category in the following three subsections.

1.2.2.1 Point Algorithms

To produce output, a point process needs only the current pixel at its input [50]. Point algorithms are also known as ordered dither or screen algorithms. An array of thresholds determines what binary pixels get turned "on" or "off" in the halftone image. To form a screen, the array of thresholds is typically tiled over the continuous-tone image that is to be halftoned. Each pixel in the continuous-tone image is compared with the corresponding threshold in the screen to make the binary decision of turning the halftone pixel on (with pixel value 1) or off (with pixel value 0). Figure 1.6 illustrates the generation of a halftone using ordered-dither.

Computational requirements of point algorithms are minimal, since to process a pixel, only the corresponding threshold in the dither screen is needed. These algorithms provide fast halftoning solutions as parallel processing of pixels is pos-

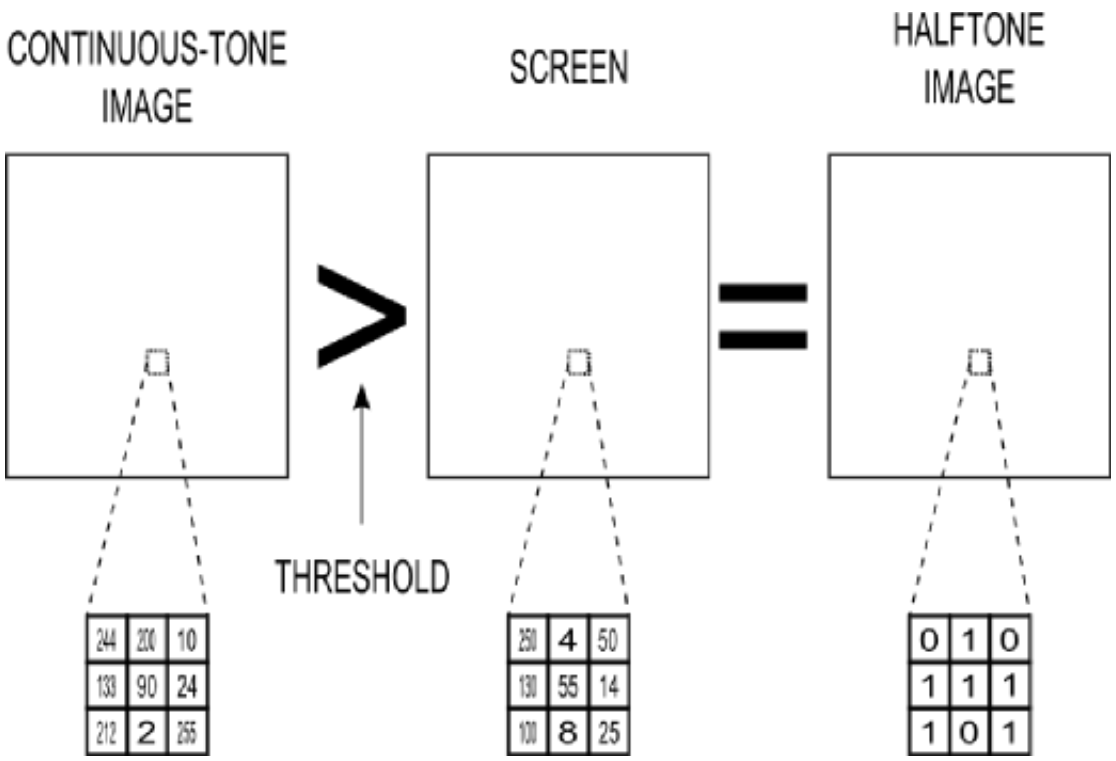


Figure 1.6: Ordered-dither: A Point Process

sible. The quality of halftones is not the best. However, since the design of blue-noise [51] dither arrays [52–54], it has been possible to achieve visual quality that is close to that achieved by using neighborhood algorithms. An example halftone image using a 16x16 void-and-cluster dither array [54] is shown in Figure 1.7.

1.2.2.2 Neighborhood Algorithms

To produce the output pixel, a neighborhood algorithm requires to process current pixel as well as surrounding pixels [50]. Error diffusion, originally introduced by Floyd and Steinberg [47], is a neighborhood based halftoning algorithm.

Figure 1.7: Peppers image halftoned using void-and-cluster dither array

Error diffusion attempts to preserve the average gray level in the halftone image by distributing the quantization error of a pixel among its causal neighbors. Figure 1.8 depicts the error diffusion process. The distribution of quantization error is controlled by the error filter used in the feedback loop. Figure 1.9 shows the raster scan path and the error filter coefficients. In raster scan, pixels are traversed from top to down and from left to right in the image. An error diffusion system is considered "lossless" if the weights in the error filter sum to 1 [55]. Fan analyzed the stability of error diffusion systems in [56]. The error diffusion system is guaranteed to be stable if the filter weights are positive and sum to 1 [56, 57]. This is indeed the case with the error filter shown in Figure 1.9. It was shown in [58] that the quantizer could be modeled as linear gain plus additive noise. Due to the dependence of the quantizer's decision on previous quantization errors, it is not possible to achieve complete

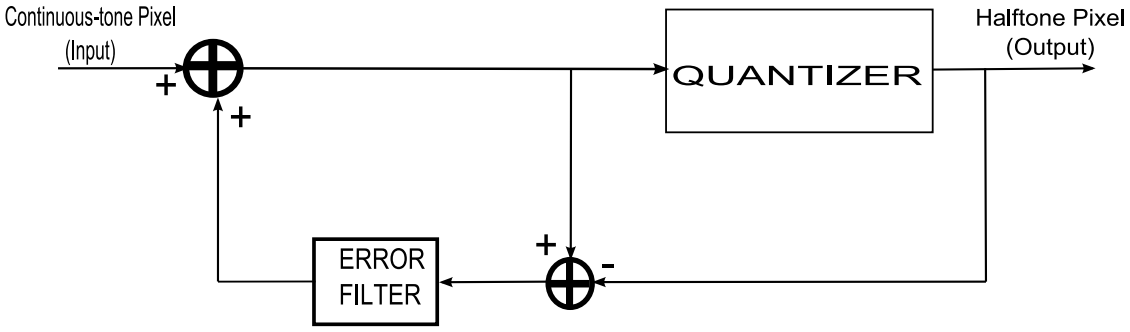
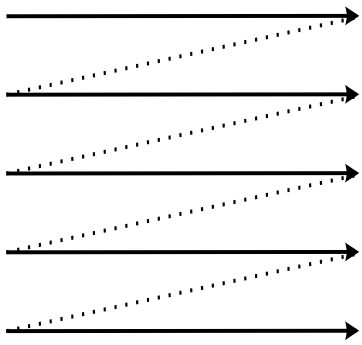


Figure 1.8: Error Diffusion

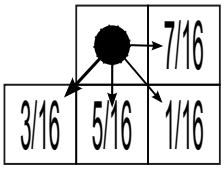
parallel processing of pixels. The process is computationally more intensive than screening. However, since it attempts to generate blue-noise patterns [50], which are visually pleasing, the quality of generated halftones is very good. Figure 1.10 shows the pepper image halftoned using Floyd-Steinberg error diffusion. Several modifications to the error diffusion algorithm have been suggested to improve upon the results produced by the original algorithm. Some of these proposed modifications have been discussed in [59–62].

1.2.2.3 Iterative Algorithms

Generally speaking, iterative algorithms produce the final halftone image by iteratively refining an initial halftone. Due to the multiple passes that the algorithm makes over the entire image, the computational requirements are higher than other classes of halftoning algorithms. As a consequence, the algorithms belonging to the iterative category are slowest in producing output image. An example of iterative algorithms is direct binary search (DBS) [63]. DBS algorithm is also an example of what is known as model-based halftoning [20], as it explicitly incorporates a model



EXAMPLE SCAN: RASTER



FLOYD-STEINBERG ERROR FILTER

Figure 1.9: Error Diffusion: Raster scan and Floyd-Steinberg error filter

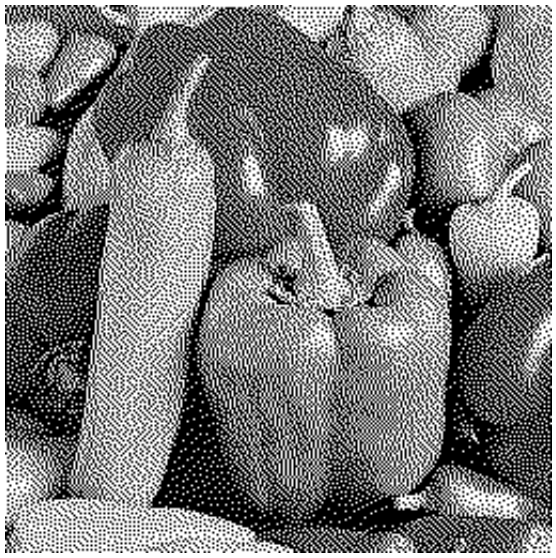


Figure 1.10: Peppers image halftoned using Floyd-Steinberg error diffusion

for the HVS. In DBS, the HVS is typically modeled as a linear shift invariant system. The frequency response obtained from the HVS model can be used to determine the point spread function of the HVS [64]. Perceptual error image is computed by convolving the difference of the halftone and the continuous-tone images with the point spread function representing the cascade of the printer and the HVS model. Figure 1.11 depicts the evaluation of the perceptual error image. Error metric in [64] is the sum of the squared values in the perceptual error image. The algorithm begins by toggling or swapping pixel values in the initial binary halftone. A change in pixel value(s) is accepted if it minimizes the error metric. This process continues until a convergence criteria is met. Figure 1.12 shows the peppers image halftoned using DBS with the HVS model derived from that proposed in [38] and optimized for viewing from a distance of 18 inches.

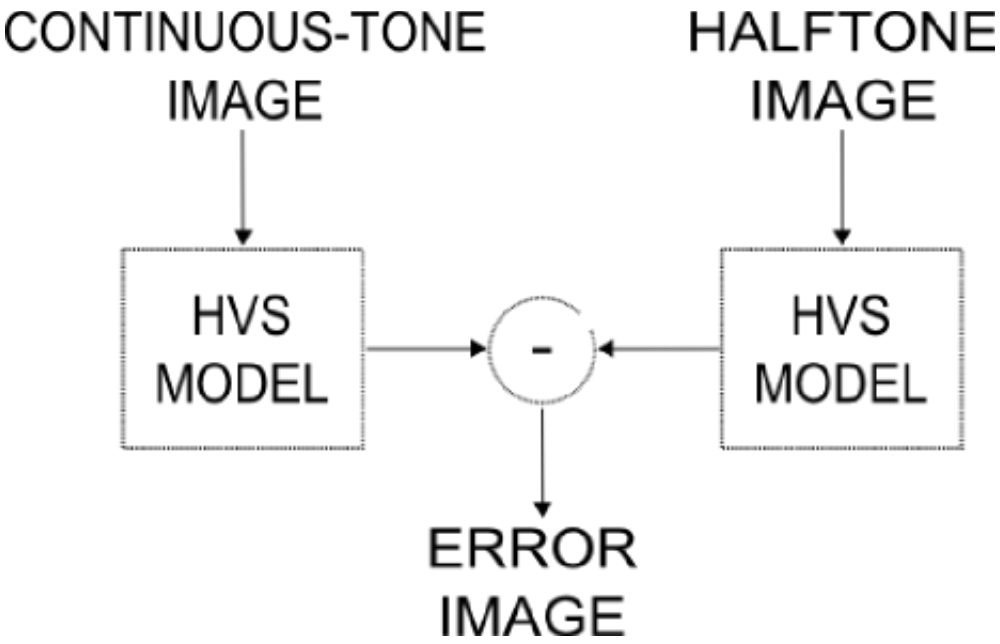


Figure 1.11: Direct Binary Search: Evaluation of perceptual error image

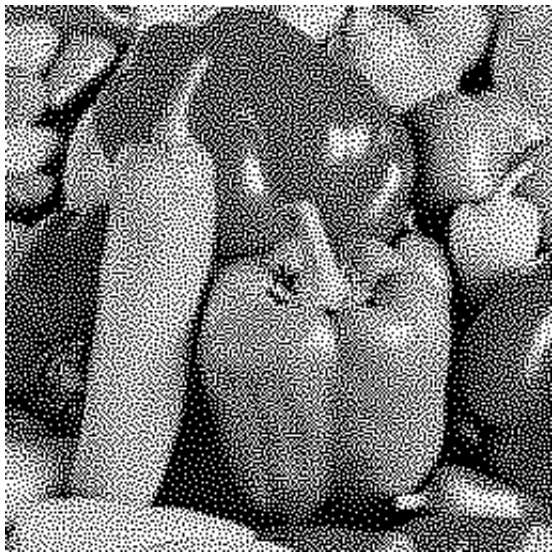


Figure 1.12: Peppers image halftoned using DBS

1.2.3 Video Halftoning

At the time of writing of this document, the main contributions in the area of video halftoning include [17, 45, 46, 48, 49, 65, 66]. Based on the type of computation involved, video halftoning algorithms can also be broadly classified in the same three categories as those of image halftoning algorithms. Point algorithm based video halftones, just like their image counterparts, can be generated by individually thresholding pixels of the continuous-tone video. This is the fastest and least computationally intensive technique to generate the halftone video. Techniques presented in [45, 48, 65, 66] are examples of neighborhood algorithms, while [46, 49] belong to the iterative algorithm category.

1.2.4 Need for Video Halftoning

It is apparent from the discussion in Section 1.2.3 that there are not many existing contributions in the area of video halftoning algorithm design. This is still considered a largely unexplored field [66]. The emerging low-power display technologies that rely on halftoning for rendering display data represent a market that could benefit from the design of new halftoning algorithms. As discussed in Section 1.1.2.3, bistable devices can benefit from video frame rate reduction, if power consumption is dependent on frame rate. The focus of this dissertation is on medium frame rate video halftones. In Chapter 3, this dissertation proposes design of medium frame rate video halftoning algorithms.

1.2.5 Halftone Quality Assessment

Objective digital video quality assessment is a very well studied area. At the time of writing of this dissertation, major recent published work on objective digital video quality assessment includes [67, 68]. For a detailed treatment of the subject, please refer to [69]. At the time of writing of this report, objective quality assessment of digital halftone videos has, on the other hand, not been as well studied. In particular, a uniformly accepted objective criteria has not been established. Fortunately, image halftone algorithm assessment has received good attention [7, 18, 70–80]. The artifacts discussed in these publications are spatial artifacts, and hence pertain to halftone images. These artifacts may also be present in halftone videos, in addition to temporal artifacts.

The assessment of commonly reported halftone video temporal artifacts has been mostly subjective [46, 48, 49], with one objective measure proposed in [66]. In making the choice of using a halftoning algorithm for a particular application, one of the criteria is the perceptual quality of the halftone. This assessment can be subjective or objective. A well established objective quality measure that correlates well with subjective assessment is preferred because it does not require labor-intensive subjective testing to evaluate a new algorithm or new algorithm settings and hence makes the decision of the designer a lot easier and reliable.

1.2.6 Need for Artifact Specific Quality Assessment in Video Halftoning

Quantization artifacts are an inevitable reality of halftoning. Often an attempt to reduce one artifact can result in an increase of another. Overall objective

quality assessment of halftone data (i.e. images and videos) is useful in predicting the user experience with viewing the halftone data. The overall quality assessment measure, however, cannot predict the impact of individual artifacts unless the individual artifact assessment constitutes the over all quality assessment framework. Techniques geared towards measuring a certain type of visual artifact are useful in isolating the specific artifact. Isolation of a visual artifact is even necessary if the visual artifact also has impact on system resources. Isolating the artifact can be helpful in deriving a relationship between the "amount" of artifact present and its impact on system resources. It can also be useful in estimating various trade-off relationships such as trade-off between the visibility of one artifact versus another or trade-off between artifact reduction and associated usage of system resources.

This dissertation presents a generalized framework for the evaluation of two key temporal artifacts in medium frame rate video halftones. The two artifacts are: (1) flicker, and (2) dirty-window-effect. As will be discussed in Chapter 2, the two key temporal artifacts are related to each other. Each of the temporal artifacts can be evaluated individually, however. This approach can serve several purposes:

1. The proposed framework enables a close study of the artifacts as well as the underlying reasons. This basic level examination of the artifacts has been utilized to develop quantitative perceptual evaluation of the two temporal artifacts. This understanding can be useful in the design of video halftoning algorithms for generating medium frame rate halftone videos.

2. The display system designer can have the flexibility to compare the perfor-

mance of competing video halftoning algorithms at a more detailed level (i.e. at artifact level).

3. The designer can prioritize his or her preferences with regards to what artifact is more important based on the system and/or consumer needs. This gives the designer flexibility to custom create his or her quality assessment criteria.

4. The proposed framework can be used to study the trade-off relationships between the artifacts.

5. The evaluation of the temporal artifacts is a step towards achieving an overall quality assessment criteria. The presented techniques to evaluate the temporal artifacts can be combined with any existing spatial artifact technique to form an overall quality assessment framework for medium frame rate video halftones.

1.3 Contributions

The research presented in this dissertation deals with digital grayscale videos, typically at a bit-depth of 8 bits-per-pixel (bpp), halftoned to the lowest possible bit-depth (i.e. 1 bpp). Unless specified otherwise, halftone data (image or video) refers to data having a bit-depth of 1 bpp.

The following is an overview of the contributions presented in this dissertation:

1. *Assessment of Key Temporal Artifacts in Medium Frame Rate Binary Video Halftones*: Artifact assessment is needed to compare the performance of two or

more video halftoning algorithms. A halftone video generated from different algorithms can be assessed subjectively and/or objectively to determine which algorithm performs best. This assessment can help decide which algorithm to choose, for a particular application or device, from the available options.

This dissertation proposes a general framework for the objective assessment of two key temporal artifacts of medium frame rate binary video halftones generated from grayscale continuous-tone videos. These artifacts are flicker and dirty-window-effect (DWE). Of these two temporal artifacts, halftone flicker has been discussed in [48, 49, 65, 66, 81]. DWE has been briefly described by Hilgenberg *et al.* in [46]. In [46], however, they have not used the term dirty-window-effect to refer to this particular artifact. The perception of temporal artifacts is, among other factors, dependent on the frame rate at which the halftone video is played back. Since the sensitivity of human visual system at lower temporal frequencies is relatively high [2], the perception of flicker in halftone videos rendered at lower to medium frame rates (15 to 30 frames per second) is correspondingly higher. This makes assessment of temporal artifacts even more critical for such frame rate halftone videos. Chapter 2 presents the first contribution of this dissertation. In Chapter 2:

- I present the results of a small scale subjective study that I conducted to evaluate two key temporal artifacts, flicker and DWE, in medium frame rate (15, 25, and 30 frames per second) binary halftone videos produced from grayscale continuous-tone videos.

- I develop a framework for objective assessment of temporal artifacts in medium frame rate (15 to 30 frames per second) binary halftone videos produced from grayscale continuous-tone videos. I present results of the performance of the framework in evaluating halftone videos generated using different halftoning algorithms. The performance of the objective artifact assessment framework is evaluated by comparing its predictions with the results of the small scale subjective study. The presented framework is intended to supplement existing spatial artifact assessment techniques. Objective assessment of temporal artifacts, developed in this dissertation, can therefore be combined with a suitable spatial artifact assessment criteria to form an overall generalized quality assessment framework.

2. *Generation of Medium Frame Rate Binary Video Halftones*: Maintaining good perceptual quality at medium frame rates is hard because at such frame rates, temporal noise shaping, in the process of halftone video generation, to higher temporal frequencies may not be possible. Medium frame rate videos may be desired to match the capabilities of devices that do not have high frame rate capability.

   The second contribution of my dissertation involves generation of medium frame rate binary video halftones from grayscale continuous-tone videos. I propose the design of two algorithms. The goal of designing these algorithms is to not only provide an alternative to the existing techniques, but also to determine the validity of the artifact assessment criteria developed in the

first contribution. The second contribution of this dissertation is presented in Chapter 3. In Chapter 3:

- I design an algorithm to generate binary video halftones with reduced DWE. The new algorithm is designed by modifying an existing algorithm. The modification is based on the DWE assessment framework developed in this dissertation. I test the algorithm on several grayscale continuous-tone videos at frame rates of 15, 25, and 30 fps. I evaluate the results for DWE performance using the objective DWE assessment framework as well as the results of the small scale subjective study. The performance is evaluated by comparing the generated halftones against the results of the existing algorithm on the basis of which the new algorithm has been designed.
- I design an algorithm to generate binary video halftones with reduced flicker. The new algorithm is designed by modifying an existing algorithm. The modification is based on the flicker assessment framework developed in this dissertation. I test the algorithm on several grayscale continuous-tone videos at frame rates of 15, 25, and 30 fps. I evaluate the resulting halftones' flicker performance using the objective flicker assessment criteria as well as the results of the small scale subjective study. The performance is evaluated by comparing the generated halftones against the results of the existing algorithm on which the design of the new algorithm has been based.

3. *Power Analysis of Video Halftoning Algorithms*: On certain types of display devices, higher flicker can result in higher power consumption [1]. This may be true for bistable display devices that consume relatively more power when switching the state of a pixel.

   As the third contribution of this dissertation, I present a comparison of power performance of five different video halftoning algorithms. The third contribution is presented in Chapter 4. The power performance comparison in Chapter 4 is applicable to bistable display devices. The power consumption calculation is based on a simplistic model of the display component of a bistable display device. The performance comparison includes the flicker, DWE, and power performance of several videos with varying content. The comparison is carried out on halftones generated for display at frame rates of 15, 25, and 30 fps. Correlation statistics are calculated using the data generated for comparison.

4. *Enhancement of Medium Frame Rate Binary Video Halftones by Reducing Flicker under a Spatial Quality Constraint*: Given a binary halftone video, it might be desired to improve its perceptual quality. To meet this goal, post-processing of halftone videos needs to be done to reduce perceptual distortions, such as flicker. Post-processing to reduce one particular artifact can, however, potentially result in introduction of or worsening of other artifacts. Introduction of any additional artifacts due to post-processing needs to be minimized.

   The fourth contribution of my dissertation involves enhancement of binary halftone videos through flicker reduction. I propose two algorithms that at-

tempt to reduce flicker under the constraint that, in the process of flicker reduction, the degradation in spatial quality of the halftone frames is controlled. To enhance a halftone video, these algorithms do not utilize the flicker assessment criteria developed in the first contribution of this dissertation. The fourth contribution of this dissertation is presented in Chapter 5. In Chapter 5:

- I design an algorithm that attempts to reduce flicker in a medium frame rate binary halftone video and, in the process, also attempts to control any resulting degradation in (spatial) perceptual quality of each frame of the halftone video. I use a model of the HVS to quantify the (spatial) perceptual distortion in the halftone frames. The algorithm assumes no knowledge of how the original (distorted) halftone video was generated. The algorithm requires the original continuous-tone video to perform enhancement of the halftone video. Using a threshold during the enhancement process, the algorithm attempts to constrain any increase in the spatial perceptual error of the frames of the halftone video. I discuss the relative computational inefficiency of the proposed enhancement algorithm.
- The first algorithm designed as part of my fourth contribution is computationally inefficient. I modify it to design the second enhancement algorithm which, in a relative sense, is computationally more efficient. The second algorithm attempts to reduce flicker in medium frame rate

binary halftone videos and utilizes threshold modulation to establish constraints on the (spatial) perceptual error of the halftone frames. I derive a theoretical error bound on the increase in perceptual error of each frame of the enhanced halftone video. I establish that the spatial perceptual error bounds (of any introduced perceptual error) are theoretically the same for both the algorithms proposed in this contribution. Using the temporal assessment framework of Chapter 2, I evaluate the halftones enhanced using an implementation of the video halftone enhancement framework proposed in this contribution to show the improvements in flicker. Several halftone videos at frame rates of 15, 25, and 30 fps are evaluated. I also compare the average spatial quality of the enhanced halftone frames with that of the original halftone frames.

5. *Enhancement of Medium Frame Rate Binary Video Halftones by Reducing DWE under a Spatial Quality Constraint*: The post-processing methods (or algorithms) proposed in the fourth contribution of this dissertation are designed to reduce flicker in binary video halftones that suffer from excessive flicker. Some halftoning methods can produce medium frame rate video halftones with excessive DWE. For these video halftones, post-processing is needed to reduce DWE, which is a temporal artifact, such that the spatial quality of the frames is not compromised.

   The fifth contribution of my dissertation comprises development of halftone post-processing algorithms for reducing DWE in a binary halftone video. In

the process of DWE reduction, these algorithms attempt to preserve the spatial perceptual quality of the video's frames. To enhance a halftone video by reducing DWE, the proposed algorithms do not utilize the DWE assessment criteria developed in the first contribution of this dissertation. The fifth contribution of this dissertation is presented in Chapter 6. In Chapter 6:

- I propose an algorithm to enhance a medium frame rate binary halftone video that suffers from excessive DWE. The proposed algorithm attempts to reduce DWE in a binary halftone video, and in this process, also attempts to maintain the spatial quality of the constituent frames of the video. The proposed algorithm assumes no knowledge of how the (distorted input) halftone video was generated. The algorithm uses the continuous-tone video, from which the (distorted) halftone video was generated, to reduce DWE while attempting to preserve the spatial quality of the frames of the enhanced video. The algorithm uses a threshold in an attempt to constrain the additional spatial artifacts that might get introduced in the halftone video as a result of its enhancement. I discuss the relative computational inefficiency of the proposed algorithm.
- I modify the first post-processing algorithm of this contribution to design an algorithm that achieves the halftone video enhancement in a computationally more efficient manner. Unlike the efficient flicker reduction algorithm proposed in the fourth contribution of this dissertation, the new, computationally efficient, DWE reduction algorithm does not

employ threshold modulation. I compare the relative computational efficiency of this DWE reduction algorithm with the threshold modulation approach used to design the efficient flicker reduction algorithm. I establish that the second, modified, algorithm to reduce DWE is computationally superior to the first algorithm of this contribution. Using the temporal assessment framework of Chapter 2, I evaluate the halftones enhanced using an implementation of the video halftone enhancement framework proposed in this contribution to show the improvements in DWE. Several halftone videos at frame rates of 15, 25, and 30 fps are evaluated. I also compare the average spatial quality of the enhanced halftone frames with that of the original halftone frames.

6. *Enhancement of Medium Frame Rate Binary Video Halftones by Reducing Flicker or DWE under Spatial and Temporal Quality Constraints*: In medium frame rate binary halftone videos, reduction of one temporal artifact is typically achieved at the expense of some increase of the other artifact. Reduction of flicker can potentially introduce dirty-window-effect in the enhanced halftone video. Similarly, reduction of DWE can result in an increase of flicker. Reduction of either of these temporal artifacts such that any introduction or increase of the other artifact is explicitly controlled is the primary goal of my sixth, and final, contribution.

   The sixth contribution of this dissertation enables additional control on how much a temporal artifact gets reduced during enhancement. This additional control is gained by the proposed enhancement algorithms by incorporating

the temporal artifact assessment criteria proposed in this dissertation. Spatial quality constraint is still in place to ensure that the perceptual quality of each frame of the enhanced halftone video does not unacceptably deteriorate. The sixth contribution of this dissertation is detailed in Chapter 7. In Chapter 7:

- I propose algorithms that enhance a medium frame rate binary halftone video by carrying out flicker reduction while attempting to control the increase of DWE and any increase in the degradation of the spatial perceptual quality of the halftone frames. The proposed algorithms are modified versions of the algorithms proposed in the fourth contribution of this dissertation. The modification is based on the flicker assessment framework, proposed in the first contribution, and introduces an additional parameter that enables a controlled balance between flicker and any DWE. Using the temporal assessment framework of Chapter 2, I evaluate the halftones enhanced using an implementation of the video halftone enhancement framework proposed in this contribution to show that the algorithm provides control to achieve a balance between flicker and DWE. Halftone videos at frame rates of 15 and 30 fps are evaluated. I also compare the average spatial quality of the enhanced halftone frames with that of the original halftone frames.
- I propose algorithms that enhance a medium frame rate binary halftone video by carrying out DWE reduction while attempting to control the increase of flicker and any increase in the degradation of the spatial perceptual quality of the halftone frames. Utilizing the DWE assessment

framework proposed in this dissertation, I propose a modification that introduces an additional parameter in each of the two algorithms proposed in the fifth contribution. This parameter enables a controlled balance between DWE and flicker. I evaluate the halftones enhanced using an implementation of the video halftone enhancement framework proposed in this contribution to show that, while reducing DWE, the proposed modification provides explicit control to achieve a balance between DWE and flicker. Halftone videos at frame rates of 15 and 30 fps are evaluated. I also compare the average spatial quality of the enhanced halftone frames with that of the original halftone frames.

1.4 Organization

The remainder of this dissertation is organized as follows. Chapter 2 proposes a general framework for the assessment of two key temporal artifacts in medium frame rate binary video halftones. The chapter also describes the design and results of a small scale subjective quality assessment study that was carried out to evaluate the temporal artifact performance of several video halftones. Chapter 3 discusses several existing video halftoning methods and develops two new video halftone generation methods. Chapter 4 analyzes relative power consumption of binary video halftones generated using five algorithms. Chapter 5 proposes video halftone enhancement algorithms that reduce flicker while attempting to preserve the perceptual quality of each halftone frame of a medium frame rate binary halftone video. Chapter 6 proposes algorithms that reduce DWE in a medium frame rate

binary halftone video while constraining any resulting degradation in the (spatial) perceptual quality of the video frames. Chapter 7 utilizes the temporal artifact assessment framework of Chapter 2 to modify the enhancement algorithms of Chapters 5 and 6. The modified algorithms attempt to reduce a temporal artifact (flicker or DWE) while constraining both the increase in the other temporal artifact (DWE or flicker) and any resulting degradation in the spatial perceptual quality of the halftone frames. Finally, Chapter 8 concludes this dissertation with a brief summary of my contributions.

1.5 List of Acronyms

2AFC	Two-alternative forced choice
2D	Two-dimensional
3D	Three-dimensional
AFR	Average flicker rate
Bpp	Bits per pixel
CCFL	Cold cathode fluorescent lamp
ChLCD	Cholesteric liquid crystal display
Cpd	Cycles per degree
CRT	Cathode Ray Tube
CSF	Contrast sensitivity function
DBS	Direct binary search
DWE	Dirty-window-effect
DWE	The Dirty-window-effect Index
EPD	Electrophoretic display
F	The Flicker Index
FDFSED	Frame-dependent Floyd-Steinberg error diffusion
FIFSED	Frame-independent Floyd-Steinberg error diffusion
FIOD	Frame-independent ordered-dither
Fps	Frames per second
FR	Full reference
GM	Gotsman's method

HD	High definition
HVS	Human visual system
LCD	Liquid crystal display
LED	Light-emitting diode
MEMS	Micro-electro-mechanical systems
MGM	Modified Gotsman's method
MSE	Mean square error
P	The Power Index
Pixel	Picture element
PSF	Point spread function
PSNR	Peak signal-to-noise ratio
SNR	Signal-to-noise ratio
SSIM	Structural similarity index map
TFT	Thin-film transistor
WSNR	Weighted signal-to-noise ratio

Chapter 2

Artifact Assessment of Video Halftones

As discussed in Chapter 1, video halftone artifact assessment is useful in the design and comparison of video halftoning algorithms. This chapter describes the visual artifacts common to image and video halftones, and proposes a generalized framework for the assessment of two key temporal artifacts in medium rate (15 to 30 fps) binary video halftones. Spatial artifacts and temporal artifacts are separately discussed. This discussion is followed by a description of a small scale subjective study carried out to evaluate the temporal artifact performance of several video halftones. The development of temporal artifact assessment framework is presented next. This is followed by a comparison of the developed objective artifact assessment measures with the results of the subjective study. The comparison is carried out using several halftone videos at different frame rates. Parts of this chapter expand upon the research presented in [81]. This chapter expands upon part of the work that has been published in [82].

2.1 Spatial Artifacts in Image Halftones

This section provides a brief introduction to image halftone artifacts along with commonly used quality assessment techniques. Details of the topic have been

discussed in [7, 18, 58, 70–79]. A recent review of halftone artifacts as well as quality assessment methods has been done by Cittadini *et al.* [72].

Some artifacts are specific to the type of the halftone algorithm used to generate the halftone image [78]. For example, patterning is an artifact that typically results from spatial replication of a dither array over the entire image. Thus, this artifact is typical of screening methods [72]. Similarly, directional artifacts or limit cycles typically appear in error diffused halftones as a form of nonlinear distortion [58]. Limit cycles generally appear as periodic binary patterns under constant input [55]. Fan and Eschbach analyzed the limit cycle behavior of error diffusion and suggested modification of error filter weights to control the limit cycles [55]. Classical error diffusion algorithm of [47] also suffers from "worm" artifacts in the shadow and highlight regions of an image [60]. *Moiré* is an artifact that results from superimposing two or more halftone patterns [18]. False contouring is another artifact that is possible in image halftones [50]. If the gray level in the continuous-tone image changes smoothly, the halftone patterns might abruptly change to represent a different gray level. If this happens along a constant gray level contour in the continuous-tone image, a "false" edge might be perceivable in the halftone [57]. Pappas and Neuhoff suggest that such false contouring can be reduced by adding to the continuous-tone image, white, uniform noise with amplitude equal to half the quantization level spacing [57]. Edge blurring is an artifact that can get introduced in the halftone image, if the halftoning algorithm is unable to properly "track" rapid gray level changes in the continuous-tone image. Error diffusion and DBS algorithms track rapid gray level changes better than order-dither based halftoning algorithms.

Therefore, they typically produce images that are relatively sharper [57]. Edge sharpening is, however, another artifact of classical error diffusion. Edge behavior of error diffusion has been analyzed in [58, 83]. Traditional error diffusion [47] can also cause unwanted textures, known as "worms," in the highlight and shadow regions of an image [59].

Since the HVS is involved in the perception of artifacts, mean-square-error (MSE) is not a good criterion for evaluation of halftones [50]. Similarly, distortion metrics of Signal-to-noise ratio (SNR) and peak signal-to-noise ratio (PSNR) are not suitable for halftone image quality assessment as they do not correlate well with human perception [76]. Quality measures such as weighted signal-to-noise ratio (WSNR) that take the HVS into account perform better [76]. In fact, the error metric used in DBS [64] is based on a weighting derived from a model for the HVS.

Some quality assessment techniques assess the overall capability of a halftoning algorithm in generating good quality halftone images. An example of such a technique is the use of grayscale ramp. The grayscale ramp is halftoned using the algorithm that is to be evaluated [7, 18]. The resulting halftone ramp image shows strengths and weaknesses of the algorithm at discrete gray levels. The halftone ramp image has to be viewed by a human to assess the quality. Another example of image independent techniques involves looking at particular spatial and spectral statistics [7, 18, 77] of the halftones generated (at particular gray levels) by the algorithm in question. The obtained statistics are then evaluated to see how close they are to the ideal statistics or the model. An example of such a model is the blue noise model [51, 77]. According to this model, a halftone algorithm that pro-

duces outputs with pattern energy concentrated in high frequencies (also known as blue noise) performs well in producing high perceptual quality halftones. Since the human visual system behaves like a low pass filter [77, 84], the spectral components of the high frequency patterns lie in regions that are not very visible to the human observer. The blue noise model also specifies ideal spatial characteristics of halftone patterns. Error diffusion, originally introduced by Floyd and Steinberg [47], is a popular halftoning technique that attempts to generate halftone patterns with blue-noise spectral characteristics [50]. The blue-noise model benefited the design of some ordered dither based techniques, such as those in [52–54]. Recently a new halftoning algorithm assessment technique has been proposed in [74] that provides an alternative to the widely used assessment technique proposed by Ulichney and Lau *et al.* [77].

2.2 Key Temporal Artifacts Specific to Binary Video Halftones

The key temporal artifacts typical to medium frame rate binary halftone videos are [46, 48, 49, 65, 66]: (1) flicker, and (2) the dirty-window-effect (DWE). In this discussion, it is assumed that each halftone frame is a good representation of the continuous-tone frame. Played back in a sequence, these frames may produce the perception of the key temporal artifacts discussed in this dissertation. If the halftone video is to be displayed at higher frame rates, it would be possible to exploit the temporal averaging properties, in addition to the spatial properties, of the HVS to reduce the perception of these artifacts. However, at medium frame rates (15 to

30 frames per second), the temporal sensitivity of the HVS is relatively higher [2]. The next two sections discuss the key temporal artifacts and their significance.

2.2.1 Flicker in Video Halftones

Gotsman [49] described binary halftone image sequence flicker as a form of high frequency temporal noise. When two successive *perceptually similar* frames of a continuous-tone video differ in pixel distributions when halftoned, flicker is observed in the resulting halftone video [49]. It may be true that when viewed individually a binary frame might be perceptually similar to its continuous-tone version. However, at the same time, it is also possible that the adjacent binary halftone frames differ significantly from each other at individual pixel locations. When this happens, two successive *similar* continuous-tone frames might perceptually appear temporally smoother than their corresponding binary halftones. This temporal smoothness at individual pixel locations of the *similar* successive continuous-tone frames is a consequence of the availability of higher bit-depth. On the other hand, the binary halftone versions might end up having a lot of pixel locations toggle their values between *similar* successive frames due to having a bit-depth of 1 bpp. This is called halftone flicker. Flicker might be considered a significant factor due to reasons that might include:

- *Visible Distortion:* On different display systems, the presence of halftone flicker can produce different forms of visible visual distortion. For example, full-field flicker may be observed on certain Liquid Crystal Display (LCD) systems. The same flicker might appear as scintillations on Cathode Ray Tube

(CRT) based systems. Generally speaking, at medium frame rates the visibility of flicker is high because HVS is very sensitive to temporal distortions at such frame rates [2, 17]. Flicker may not be perceived as annoying visible distortion in some cases where spatiotemporal noise shaping has been done [17]. This is typically possible in higher frame rate videos where temporal averaging properties of the HVS come into play [46, 48].

- *Compression:* Lower flicker in a binary halftone video means fewer *unnecessary* pixel toggles along the temporal dimension. A pleasant consequence of this property would be better compression performance [49, 66, 85]. This might be important since storage capacity on handheld devices is typically limited.

- *Power Conservation:* Binary devices that consume power each time a pixel value is toggled can benefit from flicker reduction [1]. Such devices might also operate on lower frame rates to conserve power.

- *Pixel Life:* Devices with micro-electro-mechanical systems (MEMS) based bistable pixels could possibly experience pixel failure, over extended times, due to pixel fatigue. Reduction of flicker would potentially alleviate this problem for such display devices rendering halftone video. The preferable operational frame rate of these devices might be low as well.

Besides doing overall quality assessment, the above mentioned reasons also make it necessary to explicitly assess the amount of flicker present in the halftone video. This is especially true for devices whose power consumption or pixel life

is affected by flicker. Explicit assessment of flicker in binary halftone videos has been done by taking the absolute pixel-by-pixel difference of two successive halftone frames [48, 49]. The resulting difference image shows "on" pixel locations (having a value of 1) where the pixel values toggled. Figure 2.1 shows the difference image taken from frames 36 and 37 of the halftone video obtained by halftoning the Trevor sequence using frame-by-frame Floyd-Steinberg [47] error diffusion. Figure 2.2 shows the same difference image obtained when the halftoning was done by using a 16-by-16 void-and-cluster mask [54] on independent frames. It can be concluded by comparing the two difference images that void-and-cluster mask based halftone video has relatively lower flicker between frames 36 and 37.

The difference image technique is good for evaluation of flicker. However, there are several problems with this approach. It is not entirely objective, since a human observer is needed to look at the difference images. Moreover, a difference image only gives an evaluation of flicker between the frames whose difference has been taken. For longer videos, it may not be practical to evaluate flicker over the entire video. The difference image would also show high flicker (false positive) between frames whenever there is a scene change, although that should really not be perceived as flicker due to the scene change, as a consequence of temporal masking properties of the HVS [86, 87]. Thus, this method of flicker evaluation is not perceptual.

An alternative method to evaluate flicker over the entire video has been used in [66]. That method builds on the difference image approach by computing the average flicker rate over a frame by adding the number of "on" pixels in the

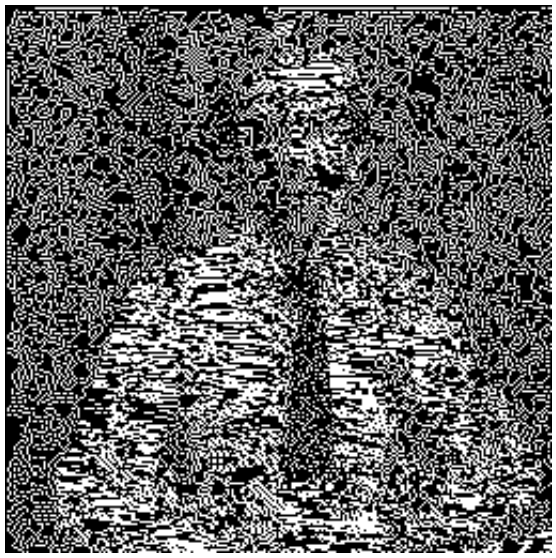


Figure 2.1: Absolute pixel-wise difference of frames 36 and 37 in the Floyd-Steinberg error diffusion halftone of the Trevor sequence.

corresponding difference image and then dividing the resulting sum by the total number of pixels in the difference image. This gives one number, called average flicker rate (AFR), per adjacent pair of frames. This number is then computed for all the adjacent frame pairs in the video and plotted as a function of frame number in [66]. The resulting plot helps visualize flicker performance of the entire video. This technique is better than the difference image approach because the entire video behavior is observable in one plot. However, in this method, masking properties of the HVS are not incorporated. Although it is objective, this method does not give us a perceptual measure for flicker.

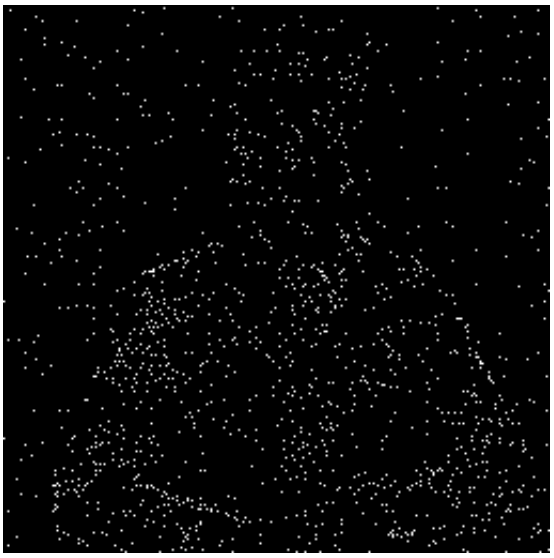


Figure 2.2: Absolute pixel-wise difference of frames 36 and 37 in the Trevor sequence halftoned using a 16x16 void-and-cluster dither array.

2.2.2 Dirty-window-effect in Video Halftones

In the context of binary halftone videos, the term dirty-window-effect (DWE) refers to an artifact that causes the perception of viewing moving objects through a dirty window. This artifact has not been explicitly discussed much in the published work. Hilgenberg *et al.* discuss this artifact in [46]. This artifact is visually annoying and can become more visible if the halftone video shows moving objects. This artifact is caused by "over" stability of binary pixels in the temporal dimension. This results in binary pattern not "sufficiently" changing in response to a changing scene in the continuous-tone video.

An example to illustrate the point discussed in the paragraph above follows. For this example, each frame of the standard Caltrain sequence [88] was indepen-

dently halftoned using Ulichney's 32-by-32 void-and-cluster mask [54]. The first continuous-tone frame, and the first halftone frame of the Caltrain sequence are shown in Figures 2.3 and 2.4 respectively. The second continuous-tone frame, and the second halftone frame of the Caltrain sequence are shown in Figures 2.5 and 2.6 respectively. The absolute difference of the first two (grayscale) continuous-tone frames is shown in Figure 2.7. In this figure, the brighter regions (i.e. pixels) represent spatial locations where the two successive frames differed in luminance. The absolute difference image of the halftone frames depicted in Figures 2.4 and 2.6 is shown in Figure 2.8. In the image shown in Figure 2.8, the dark pixels represent spatial locations where the pixels in the successive halftone frames have identical values. Observe that the locations of some of the dark pixels in the image shown in Figure 2.8 overlap with the locations that represent change of scene (due to object or camera motion) in Figure 2.7. These are the spatial locations where DWE is likely to be observed in the halftone video. This was observed upon visual inspection of the halftone sequence at frame rates of 15 fps and 30 fps. Now refer to Figure 2.9, which shows absolute difference of the first two frames halftoned using Gotsman's method [49]. Gotsman's method is an iterative halftoning technique [49]. Compare Figures 2.8 and 2.9 with Figure 2.7 to observe that Gotsman's method [49] produces less DWE than the frame independent void-and-cluster method. This was found to be the case upon visual inspection of these videos at frame rates of 15 fps and 30 fps.

It is interesting to observe how these temporal artifacts, flicker and DWE, are related. Stability of pixel values in the temporal dimension would result in lower

Figure 2.3: Frame 1 of the (continuous-tone) Caltrain sequence.

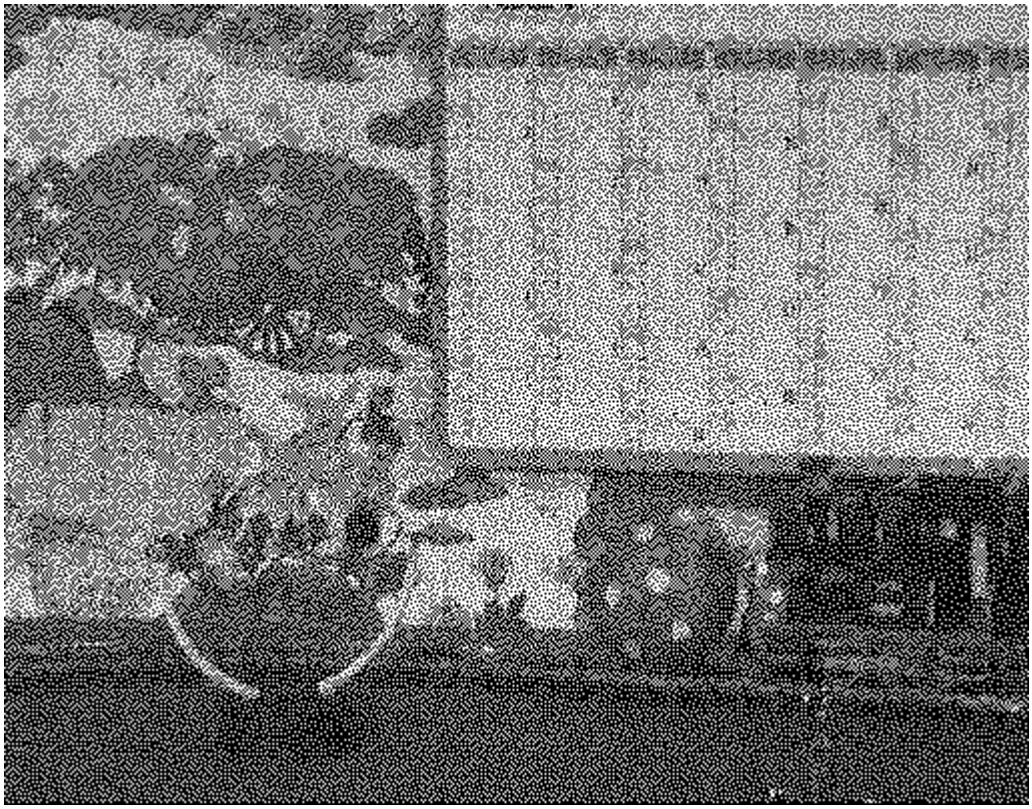


Figure 2.4: Frame 1 of the Caltrain sequence halftone. The halftone was generated using Ulichney's 32x32 void-and-cluster mask.

Figure 2.5: Frame 2 of the (continuous-tone) Caltrain sequence.

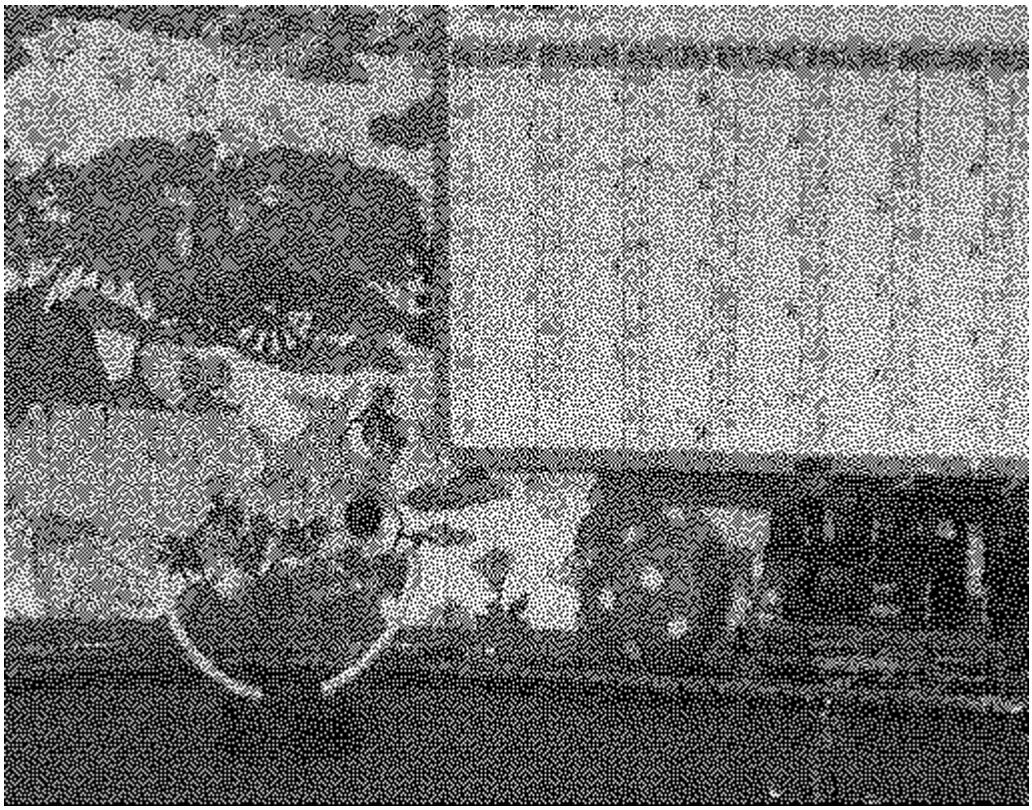


Figure 2.6: Frame 2 of the Caltrain sequence halftone. The halftone was generated using Ulichney's 32x32 void-and-cluster mask.

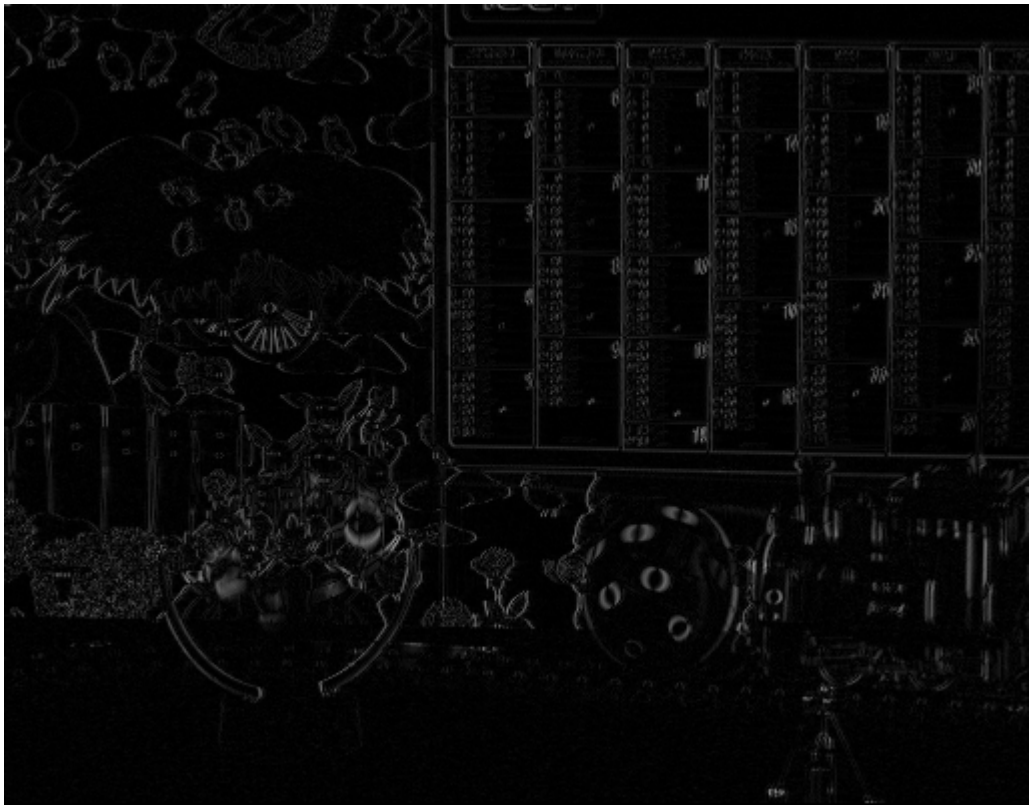


Figure 2.7: Absolute difference of frame 1 (Fig. 2.3) and frame 2 (Fig. 2.5) of the (continuous-tone) Caltrain sequence.

Figure 2.8: Absolute difference of frame 1 (Fig. 2.4) and frame 2 (Fig. 2.6) of the Caltrain sequence halftone. The white pixels indicate a change in halftone value (i.e. a bit flip). The Caltrain halftone frames 1 and 2 were generated using Ulichney's 32x32 void-and-cluster mask.

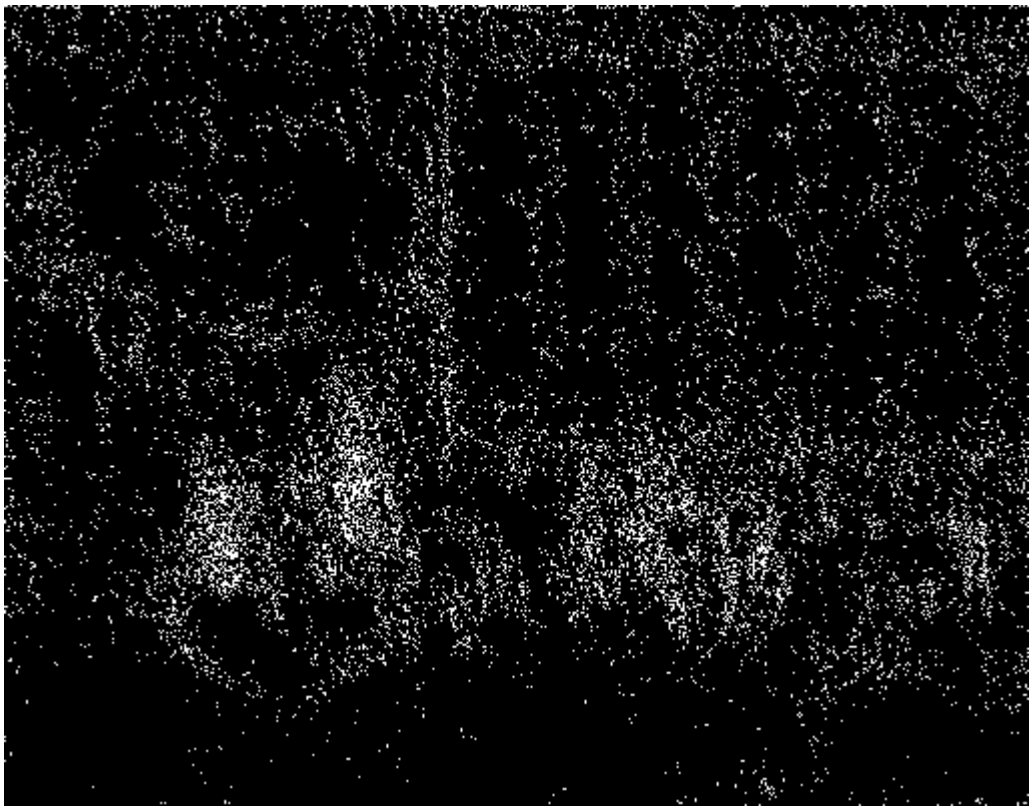


Figure 2.9: Absolute difference of frame 1 and frame 2 of the Caltrain sequence halftone generated using Gotsman's iterative method.

flicker. However, if there is too much of this stability, DWE could be observed. A good halftone video would balance these two artifacts. Scene changes and motion should result in sufficient change, between the adjacent frames, in the corresponding pixel patterns to reduce the DWE. Minor luminance changes and perceptually similar areas in the successive frames should not cause significant change of pixel patterns to reduce flicker.

2.3 Visual Inspection of Flicker and Dirty-window-effect

Visual inspection is useful in determining the efficacy of any objective video quality assessment metric. The degree of correlation of visual inspection results with the results of the objective quality assessment metric in question determines the accuracy of the objective metric. Once the degree of accuracy of a particular video quality assessment metric is established, it can be used as a measure of confidence in using the objective metric. If the accuracy of an objective quality assessment metric is acceptable for a given display application, it can be substituted for the cumbersome and tedious subjective evaluations.

Visual inspection results are also useful in estimating the shortcomings of an objective metric. Furthermore, they can reflect how well the underlying causes of the evaluated perceptual degradations are understood. In designing an objective metric to evaluate a particular perceptual degradation or artifact, it is important to have a good understanding of the underlying cause(s) of artifact perception.

In this section, I present the design and setup of a visual inspection experiment to evaluate the key temporal artifacts in binary video halftones viewed at medium frame rates (i.e. between 15 and 30 fps). The outcome/data of the experiment are presented in later sections where it is more appropriate to present them.

A visual inspection experiment was designed. Ten human viewers with normal or corrected-to-normal vision participated in visual inspection of binary video halftones. The viewers viewed the videos on an LCD display screen. The viewers

were advised to keep the viewing distance between 18 to 36 inches. Dirty-window-effect and flicker were evaluated separately. Prior to conducting the visual inspection, each viewer was shown video sample(s) to familiarize him or her with the particular artifact (DWE or flicker) that they were to evaluate the videos for.

2.3.1 Design of Visual Inspection Experiment

The visual inspection experiment was a two-alternative forced choice (2AFC) experiment. In a 2AFC experiment, a viewer is asked to make a choice from a pair of stimuli. In my experiment, each displayed video simultaneously showed two different halftones (i.e. a halftone pair) of the same continuous-tone video/sequence. Each viewer was asked to determine which of the two halftone videos had a higher degree of the evaluated artifact (DWE or flicker). There are a couple of major advantages of choosing a 2AFC experiment for evaluating temporal artifacts in binary video halftones. First, since the subjects were non-experts, it was more reliable and easier for them to make a relative choice rather than having to make an absolute judgment. Second, one of the goals of temporal artifact evaluation metrics is to allow a designer to compare two or more halftone videos. A 2AFC experiment seemed to naturally match that goal. There is, however, also a disadvantage of using 2AFC when evaluating two videos that exhibit similar levels of an artifact. Since this was a 2AFC experiment, the viewers had to make a choice even between halftones that appeared to exhibit extremely similar perception of the artifact. There was no third choice to indicate that the halftone videos were almost similar in terms of the evaluated artifact. This is an inherent limitation of any 2AFC experiment.

No time limits were imposed on the viewing of each video/sequence pair. Viewers were allowed to repeatedly watch the same video pair until they could make a decision. Nine viewers viewed the videos in standard indoor lighting conditions, while one viewer viewed the videos in very similar lighting conditions but in an open and covered environment (patio) at night time. The viewers were asked to first evaluate dirty-window-effect and then flicker.

2.3.2 Evaluated Video Halftoning Algorithms

In the visual inspection experiment, five different video halftoning methods were evaluated. In the first video halftoning method which I will call frame-independent ordered-dither (FIOD), each halftone sequence was formed by using ordered-dither technique on each frame independently. The threshold array was formed by using a 32x32 void-and-cluster mask [54]. In the second video halftoning method which I will call Gotsman's method (GM), each halftone sequence was formed by halftoning the sequence using the technique of [49]. In the third video halftoning method which I will call frame-independent Floyd-Steinberg error diffusion (FIFSED), each halftone sequence was formed by halftoning each frame independently using Floyd-Steinberg [47] error diffusion. The implementation of Floyd-Steinberg [47] error diffusion algorithm was obtained from the Halftoning Toolbox [89]. In the fourth video halftoning method which I will call the modified Gotsman's method (MGM), the halftone sequence is formed by the technique described in Section 3.2.1. In the fifth video halftoning method which I will call frame-dependent Floyd-Steinberg error diffusion (FDFSED), the halftone sequence

is formed using the technique described in Section 3.2.3.

2.3.3 Videos for Visual Inspection

In the 2AFC experiment, DWE and flicker were evaluated separately. For evaluating DWE, 75 videos were viewed by each viewer. These 75 videos were divided into three sets of 25 videos each. The first set was created to compare DWE performance of GM versus MGM. The set comprised of nine videos displayed at 30 fps, nine videos displayed at 15 fps, and seven videos at 25 fps. The second set was created to compare DWE performance of GM versus FIOD. The set comprised of nine videos displayed at 30 fps, nine videos displayed at 15 fps, and seven videos at 25 fps. The third set was created to compare DWE performance of GM versus FIFSED. The set comprised of nine videos displayed at 30 fps, nine videos displayed at 15 fps, and seven videos at 25 fps. For evaluating flicker, three sets of videos were used. The first set was created to compare flicker performance of FIFSED versus FDFSED. The set comprised of nine videos displayed at 30 fps, nine videos displayed at 15 fps, and seven videos at 25 fps. The second and third sets were the same (two) sets that were used for comparing (for DWE evaluation) GM against FIOD, and for comparing GM against FIFSED. This time, however, they were used for flicker evaluation.

In each of the video sets used for evaluating flicker or DWE, the nine videos displayed at 30 fps included halftones of the continuous-tone Caltrain, Tempete, Miss America, Susie, Tennis, Trevor, Garden, Salesman, and Football sequences [88]. The nine videos displayed at 15 fps were formed by halftoning a downsampled

Table 2.1: Description of videos displayed at 30 fps.

Sequence	**Number of Frames**	**Spatial Resolution in Pixels**
Caltrain	33	400x512
Tempete	150	240x352
Miss America	150	288x360
Susie	75	240x352
Tennis	150	240x352
Trevor	99	256x256
Garden	61	240x352
Salesman	449	288x360
Football	60	240x352

version of these continuous-tone sequences. The seven videos displayed at 25 fps were formed by halftoning Pedestrian-area, Rush-hour, Sunflower, Shields (downsampled for display at 25 fps), Blue-sky, Station, and Tractor sequences [90–92]. Tables 2.1, 2.2, and 2.3 give the number of frames and spatial resolution of videos used to produce results in this dissertation.

2.3.3.1 Results of Visual Inspection

The visual inspection results for DWE performance of GM versus FIOD methods and GM versus FIFSED methods are presented in Section 2.4.3. The visual inspection results for flicker performance of GM versus FIOD methods and GM versus FIFSED methods are presented in Section 2.4.5. Presentation of the visual inspection results obtained by comparing DWE performance of GM versus MGM is deferred until Section 3.2.1, where it is more appropriate to discuss. Similarly, since

Table 2.2: Description of videos displayed at 15 fps.

Sequence	**Number of Frames**	**Spatial Resolution in Pixels**
Caltrain	17	400x512
Tempete	75	240x352
Miss America	75	288x360
Susie	38	240x352
Tennis	75	240x352
Trevor	50	256x256
Garden	31	240x352
Salesman	225	288x360
Football	30	240x352

Table 2.3: Description of videos displayed at 25 fps.

Sequence	**Number of Frames**	**Spatial Resolution in Pixels**
Pedestrian-area	250	432x768
Rush-hour	250	432x768
Sunflower	250	432x768
Shields	249	432x768
Blue-sky	217	432x768
Station	250	432x768
Tractor	250	432x768

FDFSED halftoning method is not introduced to the reader until Section 3.2.3, the comparison of flicker performance of FIFSED and FDFSED is deferred until Section 3.2.3.

2.4 Framework for the Assessment of Temporal Artifacts in Medium Frame Rate Binary Halftone Videos

In this section, I propose a generalized framework that can be utilized to evaluate the two key temporal artifacts, flicker and DWE, in medium frame rate binary video halftones. I assume that each frame of the halftone video is a good halftone representation of the corresponding continuous-tone frame. The frames viewed as a sequence may have temporal artifacts. This is, for example, the case when each frame of the continuous-tone video is halftoned independently to produce the halftone frames of the corresponding halftone video. The proposed quality assessment framework also depends on the continuous-tone video from which the halftone video has been produced. Thus, the proposed quality assessment measures are full-reference (FR) quality assessment measures. Before proceeding with the presentation of the proposed artifact assessment framework, I describe some observations about binary halftone videos:

1. Flicker and dirty-window-effect in a binary halftone video represent local phenomena. That is, their perception depends on both the temporal and the spatial characteristics of the halftone video. Thus, flicker or DWE may be more observable in certain frames and in certain spatial locations of those frames. The perception of DWE is higher if the moving objects (or regions)

are relatively flat. This means that moving objects with higher spatial frequencies (or with higher degree of contrast) are less likely to cause the perception of DWE. Similarly, the perception of flicker is higher, if the similar corresponding spatial regions of two successive halftone frames have higher low spatial frequency (or low contrast) content. It is interesting to note that for still image halftones, it has been reported that the nature of dither is most important in the flat regions of the image [50]. This phenomenon is due to the spatial masking effects that hide the presence of noise in regions of the image that have high spatial frequencies or are textured. Masking effects are dominant in the vicinity of edges and in textured regions [22].

2. Due to the temporal masking mechanisms of the human visual system (HVS) [86, 93, 94], the perception of both flicker and DWE might be negligible at scene changes.

3. Flicker and DWE are related. Reducing one artifact could result in an increase of the other. If halftone pixels toggle values between halftone frames within a spatial area that does not change much between continuous-tone frames, flicker might be observed at medium frame rates. If they do not toggle in spatial areas that change between successive frames or exhibit motion, DWE might be observed. To minimize both artifacts, a halftoning algorithm should produce halftone frames that have their pixels toggle values only in spatial regions that have a perceptual change (due to motion, for example) between the corresponding successive continuous-tone frames. Certain halftoning algorithms produce videos that have high DWE but low flicker. An example is

a binary halftone video produced by using ordered-dither technique on each grayscale continuous-tone frame independently. Similarly, there are halftoning algorithms that produce videos with high flicker but low DWE. An example is a binary halftone video produced by halftoning each grayscale continuous-tone frame independently using Floyd-Steinberg [47] error diffusion algorithm.

2.4.1 Notation

The observations discussed above are reflected in the design of the framework for evaluation of temporal artifacts, which I introduce now. To facilitate the clarity of presentation, I repeat the notation introduced in [81, 82]:

- C_i: the i^{th} frame of the continuous-tone (original) video, V_c;
- $C_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the continuous-tone frame C_i;
- $C_{s,i,j}(m,n)$: the local similarity measure between the continuous-tone frames C_i and C_j at the pixel location (m,n);
- $C_{s,i,j}$: the similarity map/image between the continuous-tone frames C_i and C_j;
- $C_{d,i,j}(m,n)$: the local dissimilarity measure between the continuous-tone frames C_i and C_j at the pixel location (m,n);
- $C_{d,i,j}$: the dissimilarity map/image between the continuous-tone frames C_i and C_j;

- D_i: the i^{th} frame of the halftone video, V_d;
- $D_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the halftone frame D_i;
- $D_{s,i,j}(m,n)$: the local similarity measure between the halftone frames D_i and D_j at the pixel location (m,n);
- $D_{s,i,j}$: the similarity map/image between the halftone frames D_i and D_j;
- $D_{d,i,j}(m,n)$: the local dissimilarity measure between the halftone frames D_i and D_j at the pixel location (m,n);
- $D_{d,i,j}$: the dissimilarity map/image between the halftone frames D_i and D_j;
- $DWE_i(m,n)$: the local perceived DWE measure at the pixel location (m,n) in the i^{th} halftone frame $(i \geq 2)$;
- DWE_i: the perceived DWE map/image at the i^{th} halftone frame $(i \geq 2)$;
- $\widehat{DWE}_i$: the perceived average DWE observed at the i^{th} halftone frame $(i \geq 2)$;
- $F_i(m,n)$: the local perceived flicker measure at the pixel location (m,n) in the i^{th} halftone frame $(i \geq 2)$;
- F_i: the perceived flicker map/image at the i^{th} halftone frame $(i \geq 2)$;
- $\widehat{F}_i$: the perceived average flicker observed at the i^{th} halftone frame $(i \geq 2)$;
- $W_i(m,n)$: the local contrast measure at the pixel location (m,n) in the i^{th} continuous-tone frame;

- W_i: the contrast map/image of C_i;
- V_c: the continuous-tone (contone) video;
- V_d: the halftone video produced by halftoning V_c.

Let I be the total number of frames in the continuous-tone video, V_c. Let M be the total number of pixel rows in each frame of V_c, and N be the total number of pixel columns in each frame of V_c.

2.4.2 Halftone Dirty-window-effect Evaluation

It has been explained in the previous section that dirty-window-effect may be observed in the halftone video if, in response to a changing scene in the continuous-tone video, the halftone patterns do not change sufficiently between successive frames of a halftone video. Based on my observations on DWE, note that $DWE_i(m,n)$ is a function of $C_{d,i,i-1}(m,n)$, $D_{s,i,i-1}(m,n)$, and $W_i(m,n)$. Therefore,

$$DWE_i(m,n) = f(C_{d,i,i-1}(m,n), D_{s,i,i-1}(m,n), W_i(m,n)). \tag{2.1}$$

For the i^{th} halftone frame, I define the perceived average dirty-window-effect as

$$\widehat{DWE}_i = \frac{\sum_m \sum_n DWE_i(m,n)}{M \cdot N}. \tag{2.2}$$

The Perceptual Dirty-window-effect Index (or more simply, the DWE Index) DWE of a halftone video V_d is defined as

$$DWE = \frac{\sum_i \widehat{DWE}_i}{(I-1)}. \tag{2.3}$$

Dirty-window-effect performance of individual halftone frames can be represented by plotting $\widehat{DWE}_i$ against the frame number. On the other hand, the Perceptual DWE Index, DWE, is a single number that represents the DWE performance of the entire halftone video. The framework introduced thus far is quite general. I have not described the form of the function in (2.1). I have also not described how to calculate the arguments of the function in (2.1). These details are provided next.

I now describe a particular instantiation of the framework introduced above. $DWE_i(m,n)$, $C_{d,i,i-1}(m,n)$, $D_{s,i,i-1}(m,n)$, and $W_i(m,n)$ constitute the maps DWE_i, $C_{d,i,i-1}$, $D_{s,i,i-1}$, and W_i respectively. To evaluate $DWE_i(m,n)$ in (2.1), I need the contrast map of C_i, W_i, dissimilarity map between the successive contone frames C_i and C_{i-1}, $C_{d,i,i-1}$, and the similarity map between the successive halftone frames D_i and D_{i-1}, $D_{s,i,i-1}$. I *derive* $C_{d,i,i-1}$ from the Structural Similarity (SSIM) Index Map [95] evaluated between the continuous-tone frames C_i and C_{i-1}. I will denote it by $SSIM\{C_i, C_{i-1}\}$. I *scale* $SSIM\{C_i, C_{i-1}\}$ to have its pixels take values between 0 and 1 inclusive. For the dissimilarity map, I set

$$C_{d,i,i-1} = 1 - SSIM\{C_i, C_{i-1}\}. \tag{2.4}$$

For the similarity map, I set

$$D_{s,i,i-1} = (1 - |D_i - D_{i-1}|) \circledast \widetilde{p}, \tag{2.5}$$

where $\widetilde{p}$ represents the point spread function (PSF) of the HVS, $\circledast$ represents the two-dimensional convolution, and $|D_i - D_{i-1}|$ represents the absolute difference image

for the successive halftone frames D_i and D_{i-1}. It has been assumed here that the HVS can be represented by a linear shift-invariant system [20] represented by $\widetilde{p}$. For the evaluation of $\widetilde{p}$, Nasanen's model [38] is utilized to form a model for HVS. Note that the pixel values of the map $D_{s,i,i-1}$ are between 0 and 1 inclusive. To account for spatial masking mechanisms, more weight is needed to be given to the spatial regions that are relatively "flat." This is done by calculating W_i. W_i represents an image that has pixels with values that represent local contrast. The calculation of high contrast content is approximated by computing the local standard deviation. To carry out this calculation, each pixel of the image is replaced by the standard deviation of pixels in a 3x3 local window around the pixel. The filtered image thus obtained is then normalized (via pixel-wise division) by the mean image, which is also computed by replacing each pixel by the mean value of pixels in a 3x3 local window around the pixel. This results in W_i. W_i is further normalized to have each of its pixels take a value between 0 and 1 inclusive. With these maps defined, (2.1) is defined as

$$DWE_i(m,n) = (1 - SSIM\{C_i, C_{i-1}\}(m,n)) \cdot D_{s,i,i-1}(m,n) \cdot (1 - W_i(m,n)). \tag{2.6}$$

Note that $DWE_i(m,n) \in [0,1]$. This instantiation of the DWE assessment framework is shown in Figure 2.10. In Figure 2.10, K, P, and R each have a value of -1. L, Q, and S have each a value of 1. The "Artifact Map" is DWE_i. Each of its pixels, $DWE_i(m,n)$, is a product of three terms. At pixel location (m,n), the first term measures the local dissimilarity between the successive continuous-tone

frames. A higher value of the first term, $(1 - SSIM\{C_i, C_{i-1}\}(m,n))$, will mean that the successive frames have a lower structural similarity in a local neighborhood of pixels centered at pixel location (m,n). This will in turn result in assigning a higher weight to any DWE observed. This reflects the fact that the "local" scene change should result in higher perception of DWE, if the halftone pixels do not change "sufficiently" between the successive frames. The second term, $D_{s,i,i-1}(m,n)$, depends on the number of pixels that stayed the same (between D_i and D_{i-1}) in a neighborhood around (and including) pixel location (m,n). It gives a measure of the perceived DWE due to the HVS filtering. Since the HVS is modeled as a low pass filter here, $D_{s,i,i-1}(m,n)$ will have a higher value, if the "constant" pixels form a cluster as opposed to being dispersed. The third term, $(1 - W_i(m,n))$, measures the *low* contrast content in a local neighborhood centered at $C_i(m,n)$. A higher value of this term will result in higher value of the perceived DWE. The effect of scene changes is incorporated by setting DWE_i to zero whenever a scene change between successive frames is detected. This is where scene change detection comes into play. This accounts for temporal masking effects. Note that between successive continuous-tone frames C_{i-1} and C_i, a very low *average* value of $SSIM\{C_i, C_{i-1}\}$ can indicate a change of scene. Any scene change detection algorithm can be utilized, however. For the results reported in this dissertation, I (manually) determined scene changes in the videos through visual inspection and manually set DWE_i to zero at frames where a scene change is determined to have occurred.

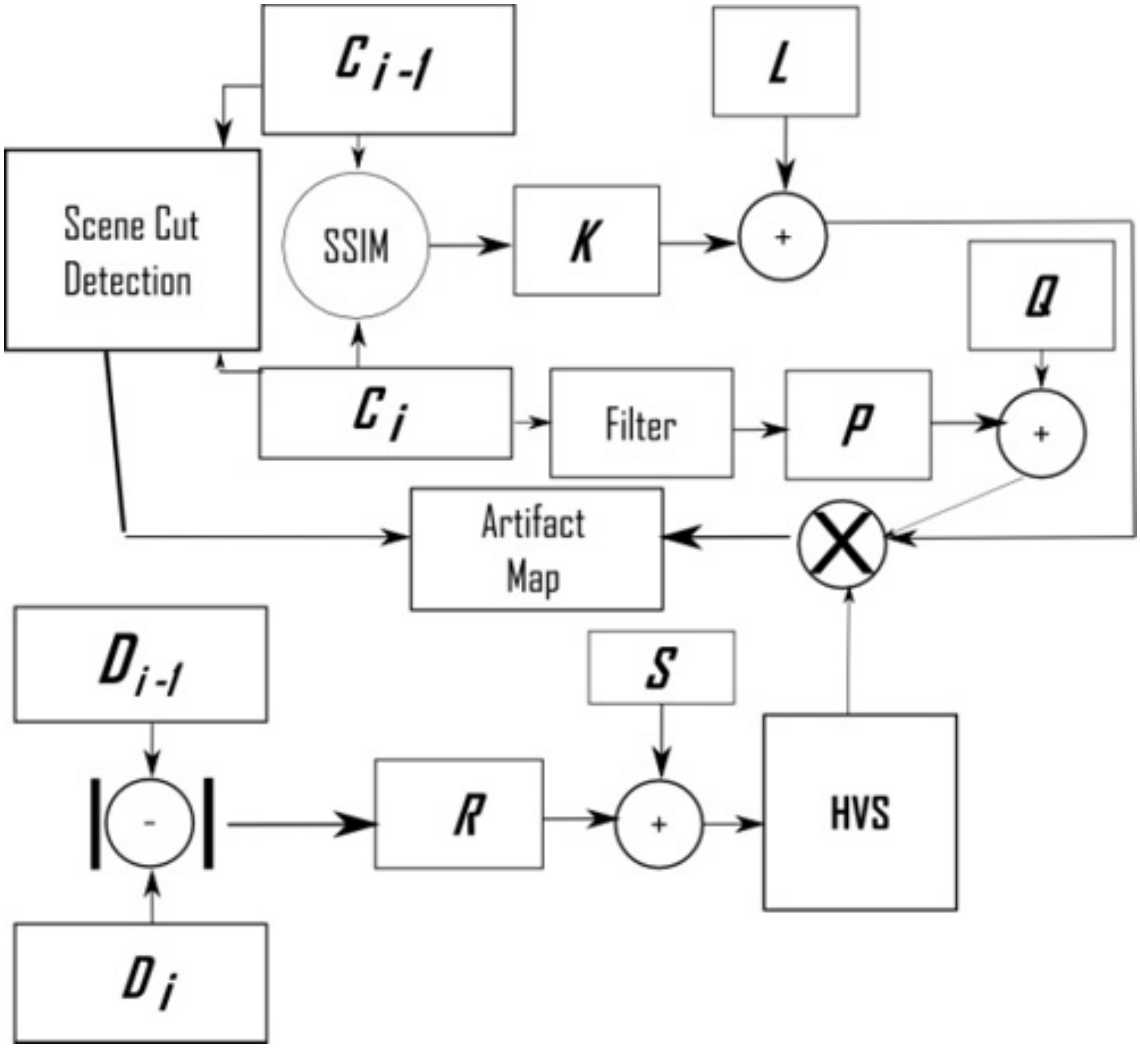


Figure 2.10: Graphical depiction of the video halftone temporal artifact quality assessment framework.

2.4.3 Experimental Results on DWE Assessment

Let us discuss the DWE evaluation results on the standard Caltrain sequence [88]. Figure 2.11 shows the dissimilarity map $C_{d,2,1}$. In this map/image, the brighter regions depict the areas where the first two frames of the Caltrain sequence are

Figure 2.11: Structural dissimilarity map of the first two frames of the continuous-tone Caltrain sequence.

structurally dissimilar. These are the regions where DWE is likely to be observed, if the corresponding halftone pixels do not "sufficiently" change between the successive halftone frames. Figure 2.12 shows W_2. In this map, the luminance of a pixel is proportional to the local normalized standard deviation in the image. Therefore, brighter regions in this image correspond to areas where DWE is less likely to be observed, if the corresponding halftone pixels do not "sufficiently" change between the successive halftone frames.

Figure 2.12: Normalized standard deviation map of the second continuous-tone frame of the Caltrain sequence.

The Caltrain sequence [88] was halftoned using three techniques. The first halftone sequence was formed by using ordered-dither technique on each frame independently. The threshold array was formed by using a 32x32 void-and-cluster mask [54]. The second sequence was formed by halftoning the sequence using Gotsman's technique [49]. The third halftone sequence was formed by halftoning each frame independently using Floyd-Steinberg [47] error diffusion. Figure 2.13 depicts DWE_i plotted as a function of frame number. According to this plot, the ordered-dither halftone sequence has highest DWE. Gotsman's technique has relatively lower DWE, whereas the error diffusion based halftone sequence has the lowest DWE. These results are consistent with the visual inspection of the Caltrain sequence played back at frame rates of 15 fps, and 30 fps.

To evaluate the performance of DWE evaluation framework more thoroughly I compare the results of the DWE evaluation measure developed in this dissertation with the visual inspection results of the experiment described in Section 2.3. Tables 2.4, 2.5, and 2.6 compare the visual inspection results with the dirty-window-effect index, DWE for halftone videos generated using Gotsman's method (GM) and frame-independent Floyd-Steinberg error diffusion (FIFSED) methods. It can be observed by looking at the data in these tables that the objective evaluation measure for DWE, the DWE Index, DWE predicts the dirty-window-effect very well. As far as DWE is concerned, the DWE Index, DWE ranks FIFSED as better than GM in producing halftone videos. This ranking of DWE is consistent with the visual inspections results.

One thing to notice from Tables 2.4, 2.5, 2.6 is that the difference in DWE

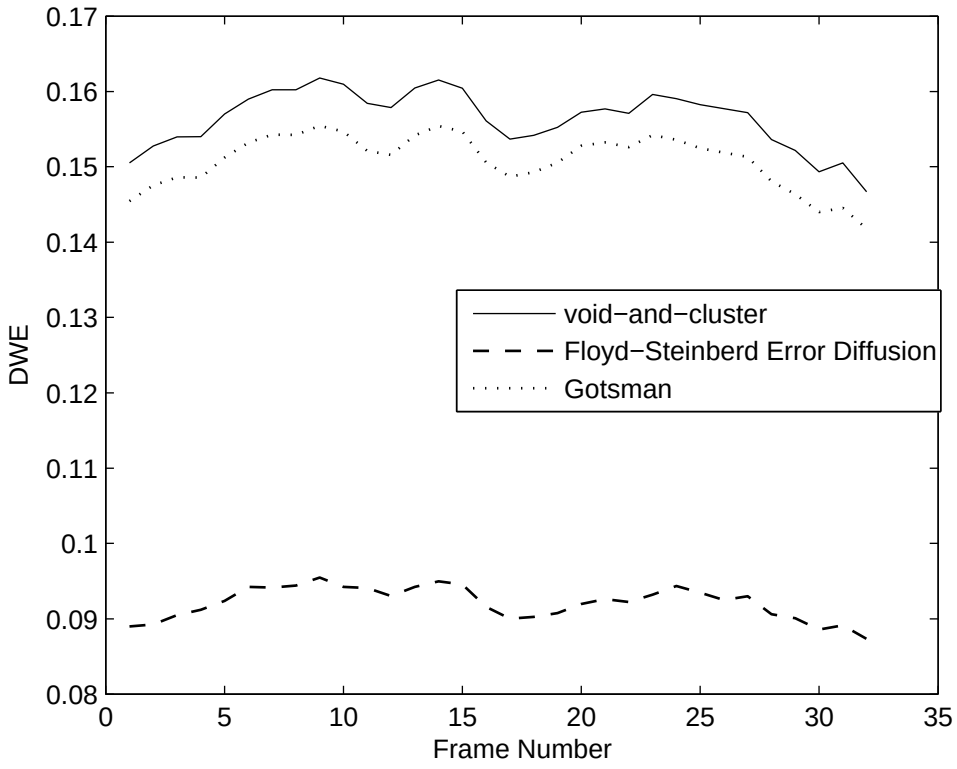


Figure 2.13: The perceived average DWE evaluation in three different halftones of the Caltrain sequence. The top curve is for frame-independent ordered-dither (FIOD) halftone. The middle curve is for halftone sequence produced using (frame-dependent) Gotsman's method (GM). The lowest curve is for frame-independent Floyd-Steinberg error diffusion (FIFSED) halftone.

Table 2.4: The DWE Index, DWE, and visual inspection results of 2AFC between 30 fps halftone videos generated using GM and FIFSED methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for FIFSED**	DWE **for FIFSED**
Caltrain	10	0.151	0	0.092
Tempete	10	0.058	0	0.042
Miss America	10	0.065	0	0.044
Susie	10	0.071	0	0.043
Tennis	10	0.11	0	0.066
Trevor	10	0.042	0	0.027
Garden	10	0.18	0	0.127
Salesman	10	0.04	0	0.026
Football	10	0.113	0	0.087

Table 2.5: The DWE Index, DWE, and visual inspection results of 2AFC between 15 fps halftone videos generated using GM and FIFSED methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for FIFSED**	DWE **for FIFSED**
Caltrain	10	0.202	0	0.134
Tempete	10	0.111	0	0.079
Miss America	10	0.049	0	0.036
Susie	10	0.096	0	0.063
Tennis	10	0.126	0	0.08
Trevor	10	0.063	0	0.042
Garden	10	0.204	0	0.16
Salesman	10	0.016	0	0.011
Football	10	0.138	0	0.109

Table 2.6: The DWE Index, DWE, and visual inspection results of 2AFC between 25 fps halftone videos generated using GM and FIFSED methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for FIFSED**	DWE **for FIFSED**
Pedestrian-area	10	0.061	0	0.051
Rush-hour	10	0.039	0	0.027
Sunflower	10	0.088	0	0.07
Shields	10	0.188	0	0.152
Blue-sky	10	0.127	0	0.112
Station	10	0.084	0	0.055
Tractor	10	0.173	0	0.127

is significant for halftone videos generated using FIFSED and GM algorithms. This is noticeable upon viewing the videos. This fact also made the choice making easier for the viewers who participated in the 2AFC study.

Tables 2.7, 2.8, and 2.9 compare the visual inspection results with the dirty-window-effect index, DWE for halftone videos generated using Gotsman's method (GM) and frame-independent ordered-dither (FIOD) methods. I first discuss the low spatial resolution videos (Tables 2.1 and 2.2) of Tables 2.7 and 2.8. Determining which video had a higher DWE from two halftone videos generated using GM and FIOD methods was not easy for low spatial resolution videos (Tables 2.7 and 2.8). Qualitatively speaking, this is because the two algorithms produce halftone videos which are very close to each other as far as DWE or flicker is concerned. For visual inspection, when comparing GM against FIOD videos, the viewers had to make a

choice regardless of this fact, since the experiment was 2AFC. As can be seen from these tables, the DWE Index, DWE, values are extremely close to each other for GM and FIOD videos. In fact, for the 30 fps Miss America sequence, DWE has the same value for both GM and FIOD sequences. The visual inspection results, however, depict that 7 viewers considered FIOD sequence to have more DWE. Regardless of how close the compared sequences appeared in DWE performance, each viewer had to choose one of the two sequences since the visual inspection experiment required them to make a choice. This is an inherent limitation of a 2AFC experiment for situations where two choices are extremely close. Consequently, responses of the viewers in Tables 2.7 and 2.8 exhibit a "mixed" behavior. Higher spatial resolution videos (Table 2.3) of Table 2.9 are a totally different case. Here you see that the observer response is fairly consistent. The visual inspection results are also in very good agreement with DWE predictions. Notice that the difference in DWE values for these videos is also significant. This indicates that making a choice for these videos was easier for observers.

2.4.4 Halftone Flicker Evaluation

The development of framework for halftone flicker evaluation will parallel the approach, utilized above, for the evaluation of DWE, since flicker and DWE are related artifacts. The development presented below is based on the framework proposed in [81]. Based on my discussion on flicker above, note that $F_i(m,n)$ is a function of $C_{s,i,i-1}(m,n)$, $D_{d,i,i-1}(m,n)$, and $W_i(m,n)$. Thus,

$$F_i(m,n) = f(C_{s,i,i-1}(m,n), D_{d,i,i-1}(m,n), W_i(m,n)). \tag{2.7}$$

Table 2.7: The DWE Index, DWE, and visual inspection results of 2AFC between 30 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for FIOD**	DWE **for FIOD**
Caltrain	4	0.151	6	0.156
Tempete	6	0.058	4	0.062
Miss America	3	0.065	7	0.065
Susie	1	0.071	9	0.077
Tennis	3	0.11	7	0.115
Trevor	3	0.042	7	0.044
Garden	6	0.18	4	0.198
Salesman	4	0.04	6	0.04
Football	4	0.113	6	0.143

Table 2.8: The DWE Index, DWE, and visual inspection results of 2AFC between 15 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for FIOD**	DWE **for FIOD**
Caltrain	3	0.202	7	0.225
Tempete	6	0.111	4	0.118
Miss America	3	0.049	7	0.052
Susie	0	0.096	10	0.11
Tennis	4	0.126	6	0.138
Trevor	5	0.063	5	0.069
Garden	4	0.204	6	0.244
Salesman	4	0.016	6	0.018
Football	5	0.138	5	0.181

Table 2.9: The DWE Index, DWE, and visual inspection results of 2AFC between 25 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for FIOD**	DWE **for FIOD**
Pedestrian-area	1	0.061	9	0.077
Rush-hour	1	0.039	9	0.044
Sunflower	1	0.088	9	0.102
Shields	2	0.188	8	0.214
Blue-sky	4	0.127	6	0.148
Station	1	0.084	9	0.086
Tractor	1	0.173	9	0.214

For the i^{th} halftone frame, the perceived average flicker is defined as

$$\widehat{F_i} = \frac{\sum_m \sum_n F_i(m,n)}{M \cdot N}. \tag{2.8}$$

The Perceptual Flicker Index (or simply stated, the Flicker Index) F of a halftone video V_d is defined as

$$F = \frac{\sum_i \widehat{F_i}}{(I-1)}. \tag{2.9}$$

The perceived average flicker $\widehat{F_i}$ can be plotted (against frame number) to evaluate flicker performance of individual halftone frames. The perceptual Flicker Index F gives a single number representing flicker performance of the entire halftone video. Next, I present a particular instantiation of the framework discussed thus far.

$F_i(m,n)$, $C_{s,i,i-1}(m,n)$, $D_{d,i,i-1}(m,n)$, and $W_i(m,n)$ constitute the maps (or images) F_i, $C_{s,i,i-1}$, $D_{d,i,i-1}$, and W_i respectively. Therefore, to evaluate $F_i(m,n)$

in (2.7), I need the contrast map of C_i, W_i, similarity map between continuous-tone frames C_i and C_{i-1}, $C_{s,i,i-1}$, and the dissimilarity map between the successive halftone frames D_i and D_{i-1}, $D_{d,i,i-1}$. I set $C_{s,i,i-1}$ to be a measure *derived* from the Structural Similarity (SSIM) Index Map [95] evaluated between the continuous-tone frames C_i and C_{i-1}. This will be denoted by $SSIM\{C_i, C_{i-1}\}$. $SSIM\{C_i, C_{i-1}\}$ is scaled to have its pixels values between 0 and 1 inclusive. Recall that in this dissertation, the filtering operations used to compute $SSIM\{C_i, C_{i-1}\}$ assume *symmetric* values of pixels outside the boundaries of the input images/frames, C_i and C_{i-1}. For the dissimilarity map, I set

$$D_{d,i,i-1} = (|D_i - D_{i-1}|) \circledast \widetilde{p}, \tag{2.10}$$

where $\widetilde{p}$ represents the point spread function (PSF) of the HVS, and $\circledast$ denotes the two-dimensional convolution. This is based on the assumption that the HVS can be represented by a linear shift-invariant system [20] represented by $\widetilde{p}$. $D_{d,i,i-1}$ can have its pixels take values between 0 and 1 inclusive. W_i is evaluated exactly as in the case of DWE, already described in Section 2.4.2. I define (2.7) as

$$F_i(m,n) = SSIM\{C_i, C_{i-1}\}(m,n) \cdot D_{d,i,i-1}(m,n) \cdot (1 - W_i(m,n)). \tag{2.11}$$

Note that $F_i(m,n) \in [0,1]$. This instantiation of the flicker assessment framework is depicted in Figure 2.10. In Figure 2.10, K, Q, and R each have a value of 1. P has a value of -1. L, and S have each a value of 0. The "Artifact Map" is F_i. $F_i(m,n)$ has the form described in [81]. I evaluate W_i differently in this paper. For clarity, I repeat the description of $F_i(m,n)$ as provided in [81]. $F_i(m,n)$

is a product of three terms. At pixel location (m, n), the first term measures the local similarity between the successive continuous-tone frames. A higher value of the first term, $SSIM\{C_i, C_{i-1}\}(m, n)$, will mean that the successive frames have a higher structural similarity in a local neighborhood of pixels centered at pixel location (m, n). This will in turn assign a higher weight to any flicker observed. This is desired because if the "local" scene does not change, perception of any flicker would be higher. The second term, $D_{d,i,i-1}(m, n)$, depends on the number of pixels that toggled in a neighborhood around (and including) pixel location (m, n). It gives a measure of the perceived flicker due to HVS filtering. Since the HVS is modeled as a low pass filter in this experiment, $D_{d,i,i-1}(m, n)$ will have a higher value, if the pixel toggles form a cluster as opposed to being dispersed. The third term, $(1-W_i(m, n))$, measures the *low* contrast content in a local neighborhood centered at $C_i(m, n)$. A higher value of this term will result in higher value of the perceived flicker. Finally, I incorporate the effect of scene changes by setting $F_i(m, n)$ to a low value (*zero* in this instance), if a scene change is detected between continuous-tone frames C_{i-1} and C_i. This is to account for temporal masking effects. For the results reported in this dissertation, I (manually) determined scene changes in the videos through visual inspection and manually set F_i to zero whenever a scene change is determined to have occurred between successive continuous-tone frames C_{i-1} and C_i.

2.4.5 Experimental Results on Flicker Assessment

In this section, flicker evaluation results on the standard Trevor sequence [88] are discussed. This sequence was halftoned using three techniques. The first halftone

sequence was formed by using ordered-dither technique on each frame independently (i.e. the FIOD method). The threshold array was formed by using a 32x32 void-and-cluster mask [54]. The second sequence was formed by halftoning the sequence using Gotsman's technique (GM) [49]. The third halftone sequence was formed by halftoning each frame independently using Floyd-Steinberg [47] error diffusion (i.e. the FIFSED method). Figure 2.14 depicts F_i plotted as a function of frame number. As you can see on this plot, the error diffusion based halftone sequence has the highest flicker. This is consistent with the visual inspection based evaluation of the sequences.

To evaluate the performance of flicker evaluation framework more thoroughly, I compare the results of the flicker evaluation measure developed in this dissertation with the visual inspection results of the experiment described in Section 2.3. Tables 2.10, 2.11, and 2.12 compare the visual inspection results with the Flicker Index, F for halftone videos generated using Gotsman's method (GM) and frame-independent Floyd-Steinberg error diffusion (FIFSED) methods. As can be observed by looking at the data in these tables that the objective evaluation measure for flicker, the Flicker Index, F predicts the flicker very well. As far as flicker is concerned, the Flicker Index, F ranks FIFSED as worse than GM in producing halftone videos. This flicker ranking of F is consistent with the visual inspections results.

One thing to notice from Tables 2.10, 2.11, 2.12 is that there is significant difference between F for halftone videos generated using FIFSED and GM algorithms. This is readily confirmed upon viewing these videos.

Tables 2.13, 2.14, and 2.15 compare the visual inspection results with the

Figure 2.14: The perceived average flicker evaluation in three different halftones of the Trevor sequence. The top curve is for (frame-independent Floyd-Steinberg error diffusion (FIFSED) halftone. The middle curve is for halftone sequence produced using (frame-dependent) Gotsman's method (GM). The lowest curve is for frame-independent ordered-dither (FIOD) halftone.

Table 2.10: The Flicker Index, F, and visual inspection results for 30 fps halftones videos generated using GM and FIFSED methods.

Sequence	**Number of viewers perceiving higher flicker for GM**	F **for GM**	**Number of viewers perceiving higher flicker for FIFSED**	F **for FIFSED**
Caltrain	0	0.048	10	0.333
Tempete	0	0.048	10	0.266
Miss America	0	0.011	10	0.262
Susie	0	0.047	10	0.4
Tennis	0	0.036	10	0.344
Trevor	0	0.023	10	0.31
Garden	0	0.082	10	0.232
Salesman	0	0.007	10	0.319
Football	0	0.108	10	0.329

Table 2.11: The Flicker Index, F, and visual inspection results for 15 fps halftones videos generated using GM and FIFSED methods.

Sequence	**Number of viewers perceiving higher flicker for GM**	F **for GM**	**Number of viewers perceiving higher flicker for FIFSED**	F **for FIFSED**
Caltrain	0	0.083	10	0.3
Tempete	0	0.064	10	0.254
Miss America	0	0.02	10	0.267
Susie	0	0.077	10	0.385
Tennis	0	0.055	10	0.33
Trevor	0	0.039	10	0.301
Garden	0	0.113	10	0.211
Salesman	0	0.013	10	0.323
Football	0	0.136	10	0.314

Table 2.12: The Flicker Index, F, and visual inspection results for 25 fps halftones videos generated using GM and FIFSED methods.

Sequence	**Number of viewers perceiving higher flicker for GM**	F **for GM**	**Number of viewers perceiving higher flicker for FIFSED**	F **for FIFSED**
Pedestrian-area	0	0.084	10	0.323
Rush-hour	0	0.054	10	0.329
Sunflower	0	0.083	10	0.261
Shields	0	0.087	10	0.211
Blue-sky	0	0.075	10	0.191
Station	0	0.021	10	0.302
Tractor	0	0.127	10	0.261

Flicker Index, F for halftone videos generated using Gotsman's method (GM) and frame-independent ordered-dither (FIOD) methods. For this pair of algorithms, notice the general discrepancy between the objective measure, F, and the visual inspection results. Let us now explore the possible reasons for this discrepancy.

Recall from my discussion in Section 2.4.3 that GM and FIOD methods produced halftone videos that had very similar performance for DWE. Correspondingly, GM and FIOD methods also do not differ much in their flicker performance. Note that the measures for both DWE and flicker are based on the binary pixels that toggle values between successive frames. Fewer pixels toggling value could imply higher DWE. More pixels toggling values could imply higher flicker. Perceptual flicker (DWE) is, however, determined by giving a weight to each pixel that toggles (does not toggle) value. What happens if the number of pixels that toggle values

between successive frames (in a video) is so low that even the weighting does not change the value of the perceptual objective measure significantly enough to match the visual inspection result? This effect is more pronounced if the videos compared have flicker that could be considered comparable. This could be said about videos generated using GM and FIOD methods. I analyze this observation in more detail now. I use the example of the 30 fps Caltrain sequence to illustrate this point. For the entire GM generated 30 fps Caltrain video, the average number of pixels that toggled values is 0.067. For the entire FIOD generated 30 fps Caltrain video, the average number of pixels that toggled values is 0.035. The difference between the average number of pixels toggling values is 0.032. The value of the Flicker Index, F, for the GM generated 30 fps Caltrain sequence is 0.048. The value of F for the FIOD generated 30 fps Caltrain sequence is 0.024. The difference in these two values (of F) is 0.024. Thus, the perceptual weighting reduced the gap in these differences from 0.032 to 0.024. However, the weighting was not "significant" or "heavy" enough to reduce this difference to a negative value to match the visual inspection results.

A second possible reason for the discrepancy is based on the observation that in FIOD videos a spatial periodicity of the 32x32 void-and-cluster mask [54] is observable. The flicker is also observed in a periodic spatial pattern in regions that are relatively "constant". This periodicity seemed to annoy the viewers more. This could be a reason for higher perception of flicker in FIOD videos. FIOD using a larger dither array could potentially reduce the perception of this annoying effect. Although the Flicker Index, F, calculation incorporates spatial contrast masking

Table 2.13: The Flicker Index, F, and visual inspection results of 2AFC between 30 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving higher flicker for GM**	F **for GM**	**Number of viewers perceiving higher flicker for FIOD**	F **for FIOD**
Caltrain	1	0.048	9	0.024
Tempete	0	0.048	10	0.025
Miss America	0	0.011	10	0.011
Susie	0	0.047	10	0.015
Tennis	0	0.036	10	0.019
Trevor	0	0.023	10	0.012
Garden	0	0.082	10	0.048
Salesman	0	0.007	10	0.011
Football	0	0.108	10	0.032

mechanism, its current design does not take into account spatial periodicities.

The observations discussed above point out a limitation of the flicker evaluation measure F, and suggest a direction for future research that could result in improving the perceptual flicker evaluation measure proposed in this dissertation. Performance of the Flicker Index, F, is further evaluated with results on a new video halftoning algorithm in Section 3.2.4.

Table 2.14: The Flicker Index, F, and visual inspection results of 2AFC between 15 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving higher flicker for GM**	F **for GM**	**Number of viewers perceiving higher flicker for FIOD**	F **for FIOD**
Caltrain	1	0.083	9	0.028
Tempete	2	0.064	8	0.033
Miss America	0	0.02	10	0.011
Susie	0	0.077	10	0.021
Tennis	0	0.055	10	0.023
Trevor	0	0.039	10	0.014
Garden	1	0.113	9	0.054
Salesman	0	0.013	10	0.006
Football	4	0.136	6	0.041

Table 2.15: The Flicker Index, F, and visual inspection results of 2AFC between 25 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving higher flicker for GM**	F **for GM**	**Number of viewers perceiving higher flicker for FIOD**	F **for FIOD**
Pedestrian-area	4	0.084	6	0.021
Rush-hour	4	0.054	6	0.014
Sunflower	6	0.083	4	0.025
Shields	6	0.087	4	0.037
Blue-sky	5	0.075	5	0.031
Station	3	0.021	7	0.014
Tractor	5	0.127	5	0.034

2.5 Summary

This chapter introduces the reader to the common quantization artifacts observed in image and video halftones. Artifact assessment techniques are discussed. Two key temporal artifacts typical to binary video halftones played back at frame rates of 15 to 30 fps are explained. The design of a visual inspection experiment to evaluate the two temporal artifacts is outlined. A generalized framework for the assessment of these artifacts is presented. Objective artifact assessment results obtained using the generalized artifact assessment framework are compared with the results of the visual inspection experiment to determine the validity of the proposed artifact assessment framework.

Chapter 3

Generation of Video Halftones

Chapter 2 dealt with the assessment of key temporal artifacts in medium frame rate binary video halftones. This chapter attempts to utilize the lessons learned in Chapter 2 to generate video halftones to be rendered at frame rates ranging between 15 to 30 fps. In doing so, two video halftoning algorithms will be developed. The first video halftoning algorithm will focus on reduction of DWE. The second video halftoning algorithm will focus on reduction of flicker. This chapter begins with a general discussion on the area of video halftone generation. This discussion is followed by the development of the two new video halftoning algorithms. This chapter expands upon part of the work that has been published in [82].

3.1 Video Halftoning

Video halftoning algorithms can also be divided into the same three categories that were used for classifying image halftoning algorithms. The criteria used in determining which category an algorithm belongs to is, again, the type of computation used. Video halftoning algorithms can be further classified as either frame-independent or frame-dependent. Frame-independent algorithms treat each frame as an independent image and generate the corresponding halftone frame by

employing an *image* halftoning algorithm on the continuous-tone frame. I will, alternatively, call frame-independent algorithms two-dimensional (2D) algorithms or intra-frame algorithms. Frame-dependent algorithms, on the other hand, do not treat each frame independently in the process of generating the halftone video. I will, alternatively, also refer to frame-dependent algorithms as three-dimensional (3D) algorithms or inter-frame algorithms.

As one might expect, 2D algorithms have the potential of producing video halftones that suffer from temporal artifacts. Indeed, as confirmed in Figure 2.1, the 2D error-diffusion algorithm suffers from intense flicker. However, Figure 2.2 tells us a different story. Although the video was generated by a 2D screening method, the video does not suffer from visibly annoying flicker. This quality of ordered-dither methods was pointed out by Hild *et al.* [65]. However, as noted in [65], 2D ordered-dither algorithms do not produce halftones of the quality that is achieved by 2D error diffusion based algorithms. Extension of 2D halftoning algorithms to their 3D counterparts was suggested by Mulligan in [17].

Hild *et al.* [65] incorporated the temporal dependence of frames and modified the 2D error diffusion algorithm to produce its 3D version. Their approach was to employ threshold modulation in the classical error diffusion algorithm, with the quantizer threshold changing in order to enforce temporal correlation between adjacent frames. The effect was reduced flicker. However, as noted in [49], the suggested 3D algorithm suffers from having arbitrary choice of parameters and various ad-hoc tweaks.

In [49], Gotsman suggested an iterative method to generate image halftones.

He extended his method to halftone videos as well. Although his video halftoning technique still utilized a two-dimensional HVS model, his approach tried to reduce flicker explicitly, thus making his technique a 3D method. In his technique, first frame of the halftone video is generated via 2D iterative refinement of an initial (starting point) halftone frame. After the first frame has been generated, it is used as the initial halftone for the second frame. This initial halftone for the second frame then goes through an iterative refinement process that attempts to minimize the perceptual error between second halftone and second continuous-tone frames. Since, the final output of this kind of iterative refinement is dependent on the initial halftone, choosing previous halftone frame as the initial halftone for the current frame has two advantages. First, after the refinement process has finished, the final halftone of the current and previous frames are much likely to have a very similar binary pattern resulting in reduced flicker. Second, if the two adjacent (current and previous) continuous-tone frames are similar, then choosing previous halftone frame as the starting point would result in faster convergence time. This advantage of faster convergence time turns to a disadvantage, if there is a scene change between the current and the previous continuous-tone frames.

A three-dimensional generalization of the 2D DBS technique [63] has been published in [46]. This particular technique utilized a three dimensional spatiotemporal model for the HVS developed by Kelly [44]. This seems to be a near-optimal approach. However, the search space for the 3D binary pattern that achieves even a local minimum is enormous. Despite the efficient implementation suggested in [46], the technique proves to be very time consuming. The authors compared their results

with 2D frame-independent DBS halftone as well as with the results of Gotsman's algorithm [49]. Based on subjective evaluation reported in [46], at 30 Hz frame rate, the 3D DBS technique yielded halftones with similar quality to those produced using 2D frame-independent approach. The technique proved to be slightly better for halftone videos rendered at the higher frame rate of 60 Hz. At 60 Hz frame rate, the halftone video generated using Gotsman's method is reported to have suffered from the dirty-window-effect [46]. The techniques were not compared in terms of flicker performance, however.

More recently, two error diffusion based video halftoning methods have emerged [48, 66]. Hsu *et al.* proposed a video halftoning algorithm for e-paper based display systems [66]. The algorithm proposed in [66] explicitly attempts to reduce flicker. This is done by propagating halftone pixel values from one frame to the next at locations where the pixel value differences in the corresponding continuous-tone frames are below a certain threshold. It is not clear how that threshold is determined, making it an *ad-hoc* parameter. The technique suffers from what the authors call "spot defects." Unfortunately, their suggested method to cope with "spot defects" reduces the spatial quality of the halftone frames. The method proposed in [48] utilizes motion vectors, and hence will be useful where motion vectors are readily available.

The review in the previous section points out the strengths and weaknesses of the existing video halftoning techniques. Clearly, a lot more work needs to be done in this area. When it comes to selecting a particular algorithm, there isn't much to choose from. The set of available useful algorithms may seem even smaller,

if there are additional constraints besides perceptual quality, such as a particular frame rate or a particular degree of computational complexity. Generation of video halftones for systems operating withing a certain range of frame rates should ensure minimization of artifacts in that particular range. At lower frame rates, higher temporal frequencies are not available. Availability of higher temporal frequencies is preferable because the HVS is less sensitive to such frequencies [2]. At higher frame rates, temporal averaging by the HVS can make the video appear smoother.

3.2 Proposed Algorithms

In this section, I propose two new video halftoning algorithms that aim to reduce the two key temporal artifacts present in binary video halftones played back at medium frame rates. Each of the proposed halftoning algorithm aims to reduce a single temporal artifact. To design the new algorithms, temporal artifact assessment criteria presented in Chapter 2 is utilized to modify existing video halftoning algorithms.

3.2.1 Generation of Halftone Videos with Reduced DWE

The goal of this section is to develop an iterative video halftoning algorithm that aims to reduce DWE in medium frame rate binary video halftones. The presented algorithm is a modification of an existing iterative video halftoning algorithm. The new algorithm is based on Gotsman's technique (GM) [49]. I will call the modified algorithm Modified Gotsman's method (MGM). Halftones generated using GM and MGM algorithms will be compared in Section 3.2.2.

I briefly describe Gotsman's method to generate a halftone video [49]. Gotsman's method is geared towards reducing flicker in halftone videos. It does not attempt to control DWE, however. The proposed modification attempts to explicitly control DWE, in addition to the already proposed [49] explicit attempt to control flicker. In Gotsman's method [49], the first frame of the halftone video is generated by independently halftoning the corresponding continuous-tone frame. This is done via an iterative technique which requires an initial halftone of the image as the initial guess (or the starting point). The initial halftone of the image is iteratively refined, via toggling the bits, until a convergence criteria is met. The technique results in achieving a local minimum of an HVS model based perceived error metric. For the first halftone frame, the initial guess or the starting point can be any halftone of the first continuous-tone frame. The starting point of each subsequent frame is taken to be the preceding halftone frame. This causes the subsequent frame to converge to a halftone which has a lot of pixels that do not toggle, particularly when there is no scene change. This results in producing halftone frames that are temporally better correlated than those generally produced using a frame-independent approach.

My modification to Gotsman's technique [49] is as follows. The first halftone frame is generated independently, just like in Gotsman's original technique. However, unlike Gotsman's technique [49], the initial guess for a subsequent frame is not taken to be the preceding halftone frame in its entirety. Instead, I only copy certain pixels from the previous frame. In particular, to determine the initial guess of a frame (other than the first frame), I produce a frame-independent halftone of the corresponding continuous-tone frame using a 32x32 void-and-cluster mask [54].

Then certain pixels of this frame that meet a criteria, to be described next, are replaced by pixels from the previous halftone frame. What pixels from the previous frame need to be copied is determined based on my DWE assessment technique. For the i^{th} halftone frame ($i \geq 2$), D_i, if a pixel location (m, n) in the initial halftone is such that $((1 - SSIM\{C_i, C_{i-1}\}(m, n)) \cdot (1 - W_i(m, n))) \leq T$, then the pixel from the preceding halftone frame is copied into the initial halftone frame. Here T is a threshold that controls the amount of dirty-window-effect reduction. With $T = 0.09$, I produced the Caltrain halftone and compared with Gotsman's technique. Visual inspection of the two halftone sequences confirmed the reduction in the perceived DWE due to the proposed modification to Gotsman's technique. Figure 3.1 depicts the results of my DWE evaluation framework for the two sequences. Note the reduction in DWE due to my modification of Gotsman's algorithm. This is consistent with the visual inspection results. In the next section, the performance of MGM is evaluated for several different videos.

3.2.2 Dirty-window-effect Evaluation of the Proposed Video Halftoning Algorithm

This section presents the evaluation of the performance of MGM using the DWE Index, DWE as well as results of the visual inspection experiment described in Section 2.3. Tables 3.1, 3.2, and 3.3 compare the visual inspection results with the dirty-window-effect index, DWE for halftone videos generated using Gotsman's method (GM) and the Modified Gotsman's method (MGM) algorithms. For the modified method, MGM, $T = 0.07$. Two points can be concluded based on the results reported in the tables. For most sequences the change in DWE is marginal.

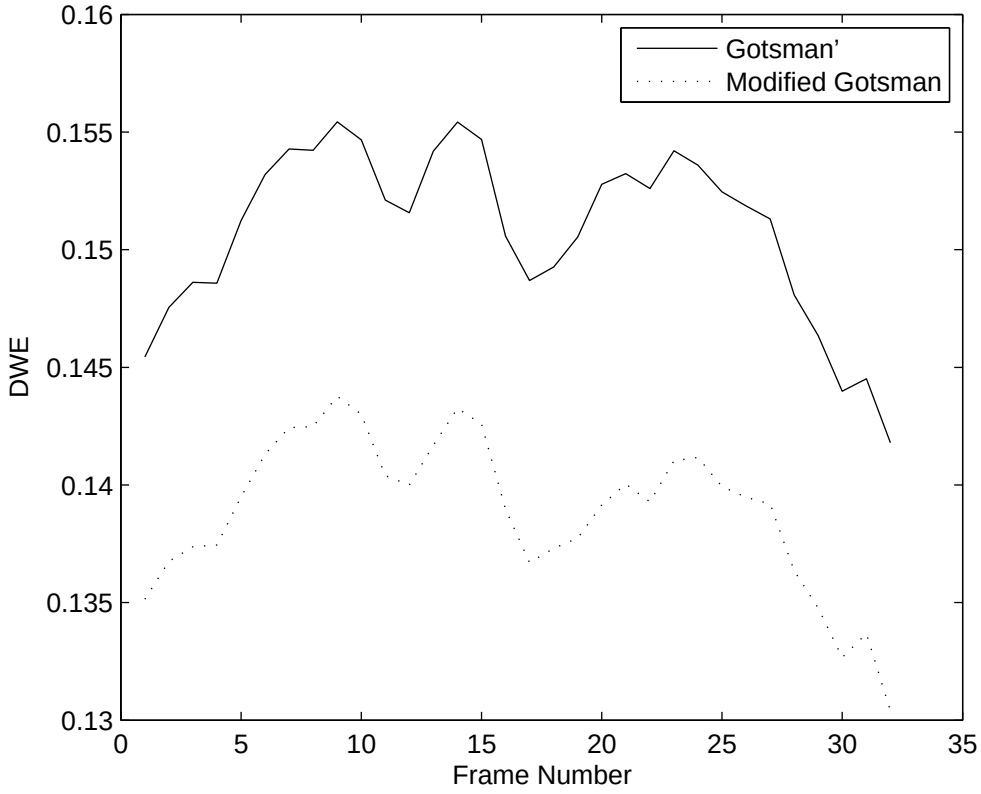


Figure 3.1: The perceived average dirty-window-effect (DWE) comparison between the Gotsman's method(GM) and the modified Gotsman's method (MGM). The bottom curve (dashed) depicts perceptual improvement in DWE achieved with MGM.

For some sequences, such as most 30 fps sequences (Table 3.1) and some 15 fps sequences (Table 3.2), the improvement in the perception of DWE due to modified Gotsman's method is marginal but noticeable. For some other sequences, MGM slightly increased DWE. This is the case for most (high resolution) 25 fps sequences (Table 3.3). Since the change, whether increase or reduction in DWE, is marginal, the results of visual inspection are not reliable. The viewers had to choose a sequence, however, since the visual inspection was based on a 2AFC experiment.

Why would MGM not always reduce the DWE? The answer lies in the fact that MGM is an iterative technique. The only parameter that I could change was the initial frame supplied. Consequently, it is the content of the initial frame halftone that is controlled via the modified method. However, since the method iteratively improves the halftone frame, there is no explicit control on how the halftone frame changes subsequently and there are no guarantees that the final (converged) frame would be such that it would have reduced DWE. The process could, however, be biased more towards reducing DWE by reducing the threshold T. However, choosing too low a value of T could in effect make this almost a frame independent technique that could suffer from increased flicker, an artifact that Gotsman's technique was designed to avoid! There is a trade off between reducing flicker and reducing DWE. I leave it to the consumers of this algorithm to decide which they prefer to reduce more. If flicker is to be reduced then a higher value of the threshold T could be used, but this would increase DWE perception. On the other hand, if DWE is to be reduced, a lower value of T could be used in MGM at the expense of increasing flicker.

Table 3.1: The DWE Index, DWE, and visual inspection results of 2AFC between 30 fps halftone videos generated using GM and MGM methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for MGM**	DWE **for MGM**
Caltrain	10	0.151	0	0.139
Tempete	10	0.058	0	0.055
Miss America	9	0.065	1	0.062
Susie	7	0.071	3	0.07
Tennis	8	0.11	2	0.104
Trevor	7	0.042	3	0.041
Garden	10	0.18	0	0.171
Salesman	9	0.04	1	0.039
Football	8	0.113	2	0.127

Table 3.2: The DWE Index, DWE, and visual inspection results of 2AFC between 15 fps halftone videos generated using GM and MGM methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for MGM**	DWE **for MGM**
Caltrain	10	0.202	0	0.2
Tempete	10	0.111	0	0.104
Miss America	9	0.049	1	0.049
Susie	6	0.096	4	0.099
Tennis	10	0.126	0	0.125
Trevor	8	0.063	2	0.063
Garden	9	0.204	1	0.21
Salesman	7	0.016	3	0.016
Football	9	0.138	1	0.161

Table 3.3: The DWE Index, DWE, and visual inspection results of 2AFC between 25 fps halftone videos generated using GM and MGM methods.

Sequence	**Number of viewers perceiving higher DWE for GM**	DWE **for GM**	**Number of viewers perceiving higher DWE for MGM**	DWE **for MGM**
Pedestrian-area	8	0.061	2	0.071
Rush-hour	3	0.039	7	0.04
Sunflower	5	0.088	5	0.092
Shields	8	0.188	2	0.195
Blue-sky	8	0.127	2	0.134
Station	10	0.084	0	0.079
Tractor	8	0.173	2	0.189

3.2.3 Generation of Halftone Videos with Reduced Flicker

The goal of this section is design of a new neighborhood based video halftoning algorithm that attempts to reduce flicker in medium frame rate binary video halftones. The presented design is a modified error diffusion algorithm that is based on the classical Floyd-Steinberg image error diffusion algorithm [47]. The design is based on the temporal artifact quality assessment framework developed in Chapter 2. The design parameters of an error diffusion halftoning system are discussed in general first. This discussion leads to the development of the new error diffusion algorithm.

3.2.3.1 Design Parameters of an Image Error Diffusion System

Error diffusion is a very popular halftoning technique. The original algorithm was the best of its time, and still ranks among the best halftoning algorithms [61]. The original algorithm introduced by Floyd and Steinberg [47] offers the dual advantage of simplicity and good visual quality. The algorithm is also publicly available. Due to these advantages, it has been used quite extensively [60]. The basic error diffusion image halftoning algorithm has been explained in Chapter 1. A detailed analysis essential to understanding the design parameters is presented below. Recall that in error diffusion an input pixel from the continuous-tone image is compared against a threshold to determine the binary output. The quantization error thus produced is distributed via an error filter to causal pixel neighbors, thus modifying their values. The pixels are processed in a processing or scan order. A typical processing order is raster scan. Figure 1.8 depicts the general error diffusion system. Raster scan and the original error filter weights are depicted in Figure 1.9. To formalize this discussion, I adopt the notation and development of [61] below:

- $i(m,n)$: the input pixel at location (m,n) in the continuous-tone image;
- $i^*(m,n)$: the modified input pixel at location (m,n);
- $e(m,n)$: the quantization error *accumulated* at location (m,n);
- $b(m,n)$: the output pixel at location (m,n) in the halftone image;
- $a(k,l)$: the weight of error propagation in the (k,l) direction;
- $Q[.]$: quantization operation.

In the light of the notation introduced above, the error diffusion system is characterized by the following equations [61]:

$$i^*(m,n) = i(m,n) + \sum_{k,l} a(k,l) \cdot e(m-k, n-l), \tag{3.1}$$

$$b(m,n) = Q[i^*(m,n)] = Q[i(m,n) + \sum_{k,l} a(k,l) \cdot e(m-k, n-l)], \tag{3.2}$$

$$e(m,n) = i^*(m,n) - b(m,n) = i(m,n) + \sum_{k,l} a(k,l) \cdot e(m-k, n-l) - b(m,n). \tag{3.3}$$

Generally speaking, due to the nonlinearity introduced by the quantization operation in (3.2), a closed form solution of the system is not possible [61]. The *accumulated* quantization error $e(m,n)$ in (3.3) is dependent on not only the current input and output pixels, but also on the past quantization errors. The design parameters of an error diffusion system can identified by examining the characterization equations described above. The main parameters of this system are [60, 61]:

- The extent of error filter and the distribution of error filter weights;
- Quantization operation;
- Processing order of pixels.

In producing binary halftones, the quantization operation is achieved via a simple step-function. It is implemented by introducing a thresholding operation. The threshold is typically a constant value of 0.5 aimed at minimization of quantization error. The classical error diffusion algorithm introduced by Floyd and Steinberg [47]

suffered from several artifacts. These artifacts were introduced in Chapter 2. In highlight and shadow regions (i.e. extreme grayscale values) of a grayscale image, error diffusion can produce unwanted textures known as "worms" [59, 60]. Error diffusion also suffers from stable periodic textures at certain intensity levels including 1/2, 1/3, and 1/4 [59, 60]. Edge enhancement is another commonly observed artifact [58]. I now discuss how each of the system design parameters has been manipulated in some publications to reduce these artifacts:

- *Modification of Error Filter*: To cope with "worms" of error diffusion, Jarvis, Judice, and Ninke [96] introduced an error filter with a larger extent. The filter had 12 coefficients. Stucki also introduced a larger error filter [97] to reduce the "worms." The larger filter size had a favorable impact on the "worms" artifact but worsened the unwanted edge enhancement and introduced mid-tone noise patterns [59]. Ulichney introduced the concept of introducing some randomness in the use of error filter weights [7]. This approach reduced "worms" artifact but introduced noise [59]. It was pointed out by Shiau and Fan that "worms" were caused by asymmetric diffusion of error, and a new error filter was proposed to introduce more symmetry in [98]. Kolpatzik and Bouman designed the filter based on HVS model to reduce the artifacts [31]. Ostromoukhov reduced the number of error distribution coefficients and designed tone-dependent weights to reduce error diffusion artifacts [60]. The reduced size of the error diffusion kernel resulted in computational superiority as well. To further improve computational performance, the size of the error diffusion kernel was further reduced to two coefficients in [99]. Other main contributions

in the design of tone-dependent error weights include [100–103].

- *Quantizer Threshold Modulation*: Some of the previous work on the use of variable thresholds in error diffusion includes [33, 100, 104–108]. The quantization operation in error diffusion is typically implemented as a step-function making the quantizer threshold a system parameter [61]. The quantizer threshold directly influences the binary output of an error diffusion system. The form of quantizer relevant to the discussion in this dissertation is given by

$$Q[f(m,n)] = step[f(m,n)], \tag{3.4}$$

where $f(m,n)$ represents the argument of the quantizer function at the pixel location (m,n). Here, $step[f(m,n)]$ is defined as

$$step[f(m,n)] = \begin{cases} 1 & \text{if } f(m,n) \geq 0, \\ 0 & \text{Otherwise.} \end{cases} \tag{3.5}$$

This is the form of the quantizer used in the original error diffusion algorithm [47]. Following the development presented in [61], I can then write

$$b(m,n) = step[i^*(m,n) - t(m,n) - 0.5]. \tag{3.6}$$

Thus, the argument of the step function in (3.4), $f(m,n)$, is given by

$$f(m,n) = i^*(m,n) - t(m,n) - 0.5. \tag{3.7}$$

If, $f(m,n) \geq 0$, the output of the quantizer is, $Q[f(m,n)] = 1$. Otherwise, the output is 0. In (3.6) and (3.7), $t(m,n)$ is the threshold modulation function

that, clearly, has an impact on the binary output of the quantizer. If $t(m,n) = 0$, there is no threshold modulation. Generally speaking two kinds of threshold modulation has been employed in error diffusion algorithms. The first kind is referred to as output dependent threshold modulation [61]. In this kind of threshold modulation, the threshold is modified to inhibit undesirable output patterns as they get discovered. In output dependent threshold modulation, the threshold modulation function in generally of the form

$$t(m,n) = g[b(m,n)], \tag{3.8}$$

where $g[.]$ is a function whose argument is the output $b(m,n)$. In general, typically several pixels around and including the pixel location (m,n) are considered in determining the value of $t(m,n)$.

The second kind of threshold modulation in error diffusion is called input dependent threshold modulation [61]. As the name suggests, in this kind of threshold modulation, the threshold gets modified based on pixel values in the input image. The threshold modulation function takes the general form

$$t(m,n) = g[i(m,n)], \tag{3.9}$$

where $g[.]$ is a function whose argument is the input $i(m,n)$. In general, typically several pixels around and including the pixel location (m,n) are considered in determining the value of $t(m,n)$.

- *Modified Scan Path*: The third parameter of an error diffusion system is the order in which the pixels of the input (continuous-tone) image are processed. A typical scan path is the raster scan path depicted in Figure 1.9. However, different scan paths are possible and have been employed with varying degrees of success in reducing output halftone artifacts [7, 109, 110].

3.2.3.2 Design of Error Diffusion Algorithm for Video Halftoning

Section 3.2.3.1 introduced the main design parameters that are typically controlled to achieve desired results in error diffusion image halftoning. The discussion of image error diffusion design parameters serves to facilitate the presentation of material in this section. In this section, design of a new video halftoning algorithm is proposed. The new video halftoning algorithm will be a modified version of frame-independent Floyd-Steinberg error diffusion (FIFSED) algorithm. The flicker assessment framework developed in Section 2.4.4 will be utilized to modify FIFSED. The halftones generated by the new algorithm are evaluated both subjectively and objectively in Section 3.2.4. Besides obtaining a new video halftoning algorithm, if the new algorithm results in improvement of perceived flicker at medium frame rates, then the proposed framework is valid. This is indeed the case as will be shown shortly.

I modify frame-independent Floyd-Steinberg error diffusion algorithm to reduce flicker. As described before, frame-independent Floyd-Steinberg error diffusion (FIFSED) algorithm halftones each frame of the continuous-tone video independently using Floyd-Steinberg error diffusion [47] algorithm for halftone images. The

general set up for image error diffusion is shown in Figure 1.8. The main design parameters of an image error diffusion system have been discussed in Section 3.2.3.1. In this system, each input pixel, from the continuous-tone image, to the quantizer is compared against a threshold to determine its binary output in the halftoned image.

To reduce flicker, FIFSED is modified by introducing frame-dependence in the algorithm. Since frame dependence in introduced in the newly designed algorithm, it will be called frame-dependent Floyd-Steinberg error diffusion (FDFSED) algorithm. FIFSED uses a quantization threshold with a fixed value of 0.5. To make the new algorithm frame-dependent, I will incorporate threshold modulation for flicker reduction. The other two parameters, the error filter weights and the scan path, remain unchanged (i.e. the same as used in FIFSED). In video halftoning, the idea of threshold modulation to reduce flicker was originally conceived by Hild and Pins [65], and later used in [48]. FDFSED works as follows. The first halftone frame is generated by halftoning the first continuous-tone frame using image error diffusion algorithm. In this algorithm, the error diffusion quantization threshold is kept a constant [47]. For the generation of subsequent halftone frames, the quantization threshold is not constant. Instead, the quantization threshold is modulated based on my flicker assessment framework. In the generation of each i^{th} halftone frame for $(i \geq 2)$, D_i. The quantization threshold $T_i(m,n)$ for a pixel location (m,n) is determined as follows:

$$T_i(m,n) = \begin{cases} 0.5 - Z \cdot (SSIM\{C_i, C_{i-1}\}(m,n) \cdot (1 - W_i(m,n))) & \text{if } D_{i-1}(m,n) = 1 \ , \\ 0.5 + Z \cdot (SSIM\{C_i, C_{i-1}\}(m,n) \cdot (1 - W_i(m,n))) & \text{if } D_{i-1}(m,n) = 0 \ . \end{cases} \tag{3.10}$$

As seen in (3.10), the amount of threshold perturbation is determined by $Z \cdot (SSIM\{C_i, C_{i-1}\}(m,n) \cdot (1 - W_i(m,n)))$, where Z is a constant that controls the effect of $(SSIM\{C_i, C_{i-1}\}(m,n) \cdot (1 - W_i(m,n)))$ on $T_i(m,n)$. The threshold modulation is designed to reduce flicker in the halftone video.

Let us now take a closer look at (3.10), and evaluate the form or type of threshold modulation designed in FDFSED. In the light of my discussion on threshold modulation in Section 3.2.3.1, the threshold modulation function for the i^{th} frame is of the form

$$t_i(m,n) = \begin{cases} +Z \cdot (SSIM\{C_i, C_{i-1}\}(m,n) \cdot (1 - W_i(m,n))) & \text{if } D_{i-1}(m,n) = 1 \ , \\ -Z \cdot (SSIM\{C_i, C_{i-1}\}(m,n) \cdot (1 - W_i(m,n))) & \text{if } D_{i-1}(m,n) = 0 \ , \end{cases} \tag{3.11}$$

where the subscript i in $t_i(m,n)$ has been introduced to identify the frame for which the threshold modulation function is valid. In (3.11), $C_i(m,n)$, $C_{i-1}(m,n)$, and $W_i(m,n)$ are evaluated from the *input* continuous-tone video. However, $D_{i-1}(m,n)$ is actually part of the *output* halftone video. Therefore, the threshold modulation used in FDFSED algorithm is both input and output dependent! I will call this a hybrid threshold modulation, where hybrid refers to the fact that the threshold modulation function is dependent on both input (i.e. continuous-tone) and output (i.e. halftone) videos.

With $Z = 0.1$ in (3.10), I produced the Trevor halftone using FDFSED and compared with that generated using FIFSED. Figure 3.2 depicts the reduction in the perceived average flicker in the Trevor halftone produced using FDFSED. Visual

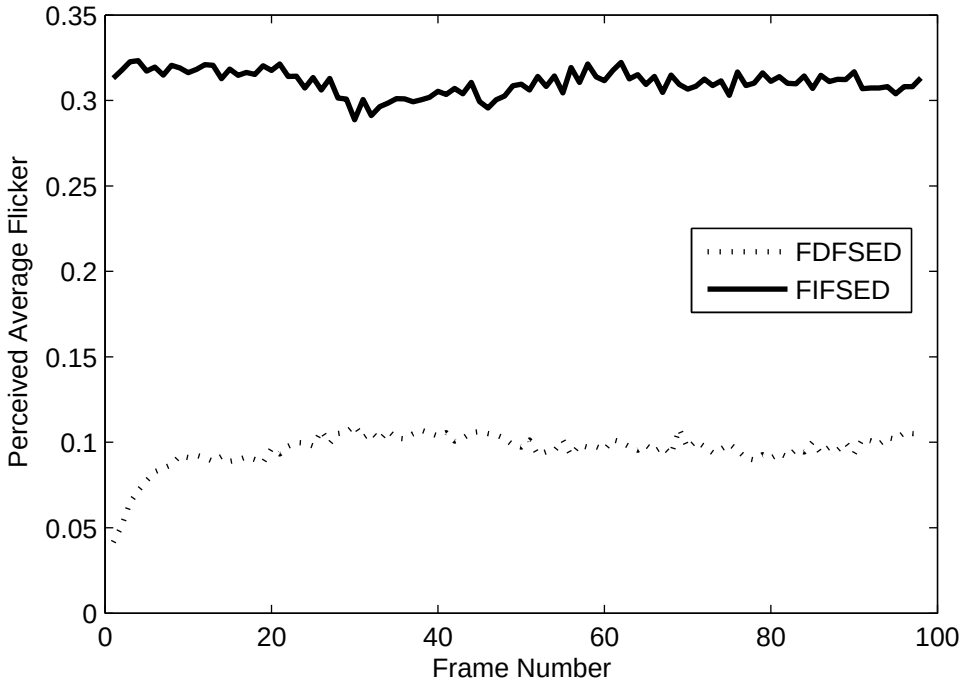


Figure 3.2: The perceived average flicker comparison between the frame-dependent Floyd-Steinberg error diffusion (FDFSED) and frame-independent Floyd-Steinberg error diffusion (FIFSED) halftones of the Trevor sequence. FDFSED results in reduced flicker.

inspection confirmed the reduction in the perceived average flicker. The performance of FDFSED, on several different videos, is evaluated in the next section.

3.2.4 Flicker Evaluation of the Proposed Video Halftoning Algorithm

In this section, I use several videos to evaluate the performance of FDFSED using the Flicker Index, F as well as the results of the visual inspection experiment described in Section 2.3. The comparison between the visual inspection results and the Flicker Index, F will also serve to further evaluate flicker evaluation framework

Table 3.4: The Flicker Index, F, and visual inspection results of 2AFC between 30 fps halftone videos generated using FIFSED and FDFSED methods.

Sequence	**Number of viewers perceiving higher flicker for FIFSED**	F **for FIFSED**	**Number of viewers perceiving higher flicker for FDFSED**	F **for FDFSED**
Caltrain	10	0.333	0	0.131
Tempete	10	0.266	0	0.108
Miss America	10	0.262	0	0.081
Susie	10	0.4	0	0.105
Tennis	10	0.344	0	0.096
Trevor	10	0.31	0	0.092
Garden	10	0.232	0	0.134
Salesman	10	0.319	0	0.081
Football	10	0.329	0	0.123

of Section 2.4.4.

Tables 3.4, 3.5, and 3.6 compare the visual inspection results with the Flicker Index, F for halftone videos generated using frame-independent Floyd-Steinberg error diffusion (FIFSED) and frame-dependent Floyd-Steinberg error diffusion (FDFSED) technique developed in this section. For FDFSED algorithm, I used $Z = 0.1$ in (3.10). Note that for all the sequences FDFSED halftones have a significantly lower flicker than the FIFSED halftones. Visual inspection results confirm this improvement. Visual inspection results also confirm the accuracy of the predictions of the Flicker Index, F for all the 25 halftone pairs.

Table 3.5: The Flicker Index, F, and visual inspection results of 2AFC between 15 fps halftone videos generated using FIFSED and FDFSED methods.

Sequence	**Number of viewers perceiving higher flicker for FIFSED**	F **for FIFSED**	**Number of viewers perceiving higher flicker for FDFSED**	F **for FDFSED**
Caltrain	10	0.3	0	0.131
Tempete	10	0.254	0	0.117
Miss America	10	0.267	0	0.079
Susie	10	0.385	0	0.111
Tennis	10	0.33	0	0.099
Trevor	10	0.301	0	0.094
Garden	10	0.211	0	0.13
Salesman	10	0.323	0	0.074
Football	10	0.314	0	0.129

Table 3.6: The Flicker Index, F, and visual inspection results of 2AFC between 25 fps halftone videos generated using FIFSED and FDFSED methods.

Sequence	**Number of viewers perceiving higher flicker for FIFSED**	F **for FIFSED**	**Number of viewers perceiving higher flicker for FDFSED**	F **for FDFSED**
Pedestrian-area	10	0.323	0	0.108
Rush-hour	10	0.329	0	0.114
Sunflower	10	0.261	0	0.121
Shields	10	0.211	0	0.114
Blue-sky	10	0.191	0	0.095
Station	10	0.302	0	0.115
Tractor	10	0.261	0	0.139

3.3 Computational Complexity of Video Halftone Generation Algorithms

In this, two new video halftone generation algorithms were introduced. Each of the two new algorithms was a modification of a previous video halftone generation algorithm. Each of the two new algorithms was designed to reduce one of the two key temporal artifacts discussed in this dissertation. The question that this section attempts to answer is: Compared to the previous algorithm, how much extra computation is needed by the *modified* algorithm? In other words, what is the computational cost of using the new algorithms? To answer this question, I need to compare FIFSED with FDFSED, and GM with MGM.

Refer to the notation in Chapter 2 and recall that I is the total number of frames in the the continuous-tone video V_c. Also, M is the total number of pixel rows in each frame of V_c, and N is the total number of pixel columns in each frame of V_c. The analysis presented in this section is based on the mathematical calculations (theoretically) required by the algorithms and is approximate. Note that it might be possible to reduce some of the operations in an actual algorithm implementation depending on the type of implementation.

3.3.1 Computational Complexity Comparison of FIFSED and FDFSED Algorithms

Frame-independent Floyd-Steinberg error diffusion (FIFSED) produces a halftone sequence by halftoning each continuous-tone frame independently using Floyd-Steinberg [47] error diffusion (FSED) algorithm. Generally speaking, to pro-

duce a halftone frame, FSED requires MN comparison operations, $4(M-1)(N-2)+3(M-1)+2(M-1)+(N-1)+(MN-1)$ addition and subtraction operations, and about $4(M-1)(N-2)+3(M-1)+2(M-1)+(N-1)$ multiplication operations. All operations combined add up to a total of $10MN-6M-6N+3$ operations per frame. There are I frames, so the total number of operations required to produce the halftone video, V_d, is about $10MNI-6MI-6NI+3I$. This implies an asymptotic computational complexity of $O(MNI)$ for the FIFSED algorithm.

Frame-dependent Floyd-Steinberg error diffusion (FDFSED) (Section3.2.3) produces the first frame of the halftone video, V_d, using the FSED algorithm, which requires a total of $10MN-6M-6N+3$ operations, as discussed above. To produce each remaining i^{th} frame of V_d, FDFSED requires threshold modulation as given by (3.10). Note for the pixel located at (m,n), (3.10) requires $(SSIM\{C_i,C_{i-1}\}(m,n)$ as well as $W_i(m,n)$, 2 addition/subtraction operations, 1 comparison operation, and 2 multiplication operations. Recall from Chapter 2 that $SSIM\{C_i,C_{i-1}\}(m,n)$ is obtained from the map $(SSIM\{C_i,C_{i-1}\}$ and $W_i(m,n)$ is obtained from the map W_i.

Let's analyze the computational complexity of calculating $SSIM\{C_i,C_{i-1}\}$ and W_i. If X is the number of pixels in the local neighborhood (i.e. window) around (m,n), to calculate $SSIM\{C_i,C_{i-1}\}(m,n)$ using the expression in [95], approximately a total of $12X+18$ operations (additions, subtractions, multiplications, and divisions) are needed. The total number of pixels in each frame of the video V_c is MN, and recall from Chapter 2 that the similarity map $SSIM\{C_i,C_{i-1}\}$ also has MN pixels. Thus, the total number of required operations is $MN(12X+18)$. Sim-

ilarly, if Y is the number of pixels in the local neighborhood (i.e. window) around (m, n), based on the definition of $W_i(m, n)$, approximately $(4Y + 1)$ computational operations (additions, subtractions, multiplications, square-roots, and divisions) are needed. To get W_i, a total of $MN(4Y + 1)$ computational operations are required.

Therefore, to generate V_d using FDFSED, the approximate total number of required computational operations is $I(10MN - 6M - 6N + 3) + (I - 1)(5MN + MN(12X + 18) + MN(4Y + 1))$. Note that since X is the number of pixels in the local neighborhood (i.e. window) around (m, n), to calculate $SSIM\{C_i, C_{i-1}\}(m, n)$, it does not grow with the image or video size. That is, for standard implementation, X is a constant. Indeed, for SSIM, Moorthy and Bovik report an asymptotic complexity of $O(MN)$ [111]. Similarly, Y is a constant in a typical implementation. So, the asymptotic computational complexity is $O(MNI)$ for FDFSED.

It should be noted that compared to FIFSED, the number of extra computational operations required by FDFSED is about $(I - 1)(5MN + MN(12X + 18) + MN(4Y + 1))$. Thus, FDFSED is computationally more expensive.

Memory usage is implementation dependent. However, it is possible to have an approximate comparison under some implementation assumptions. For a relative comparison of memory usage, I assume that the entire (current) continuous-tone frame, to be halftoned, is stored in memory. I also assume that the output halftone frame is directly output to the display device, as it gets produced, and is not stored in memory. Further, I assume that the maps $SSIM\{C_i, C_{i-1}\}$, and W_i, whenever used by an algorithm, are stored in their entirety. The memory estimate is approximate and is in terms of number of pixels stored for producing an output halftone frame.

Since, typically, the frames are halftoned sequentially, memory usage comparison is done in terms of storage required to produce one output frame.

In FIFSED, the current frame gets modified as the output halftone frame is produced. There are 4 filter taps for the Floyd-Steinberg filter [47]. Assume that each tap requires the same amount of space needed for each pixel. This means that the memory usage of FIFSED is $MN + 4$ pixels per frame. In FDFSED, the first frame gets produced the same way as the first frame for FIFSED. However, for each subsequent frame, MN pixels are needed to store the current (input) continuous-tone frame, MN pixels are needed to store the map $SSIM\{C_i, C_{i-1}\}$, MN pixels are needed to store the map W_i, MN pixels are used to store the previous halftone frame (under the simplistic assumption that a binary halftone pixel requires the same storage space as a continuous-tone pixel), and 4 pixels for filter taps. The memory requirement for producing each output frame, other than the first output frame, using FDFSED is, therefore, $4MN + 4$. Thus, compared to FIFSED, FDFSED requires a storage of $3MN$ more pixels for producing an output frame.

3.3.2 Computational Complexity Comparison of GM and MGM Algorithms

As discussed earlier in this dissertation, iterative processes are computationally most expensive! GM and MGM are iterative video halftone generation algorithms. GM and MGM generate halftones through a process commonly called direct binary search (DBS) [20]. If an exhaustive search for the best possible binary halftone pattern for each frame is performed, then the total number of possible

candidates to consider for each frame is 2^{MN} [20]. In the implementations used to generate the results in this dissertation, exhaustive search strategy was not adopted. Instead, the implementations were based on the iterative optimization technique suggested in [64] was used. Iterative optimization techniques are computationally more feasible but find only a local minimum [20]. Iterative optimization assumes an initial estimate for each halftone frame. This means that the final output, and hence the convergence time is dependent on the initial estimate of each frame [49, 64]. For example, if the initial estimate is close to a local minimum, it might take relatively fewer iterations to converge to the final halftone. Thus, the exact number of iterations required to produce each (output) halftone frame cannot be predicted. Computational complexity is obviously dependent on the number of iterations taken to generate each halftone frame. In his implementation of his method (GM), Gotsman reports that it took up to 12 passes to achieve absolute convergence for each frame [49].

As discussed above, GM and MGM are iterative processes whose convergence time is hard to predict in an exact manner. The goal of this section is, however, relative complexity analysis. It is possible to have an approximate comparison of GM and MGM by comparing the design of GM and MGM algorithms to estimate the number of additional computational operations needed by MGM. Recall from Section 3.2.1, after the generation of first video halftone frame, MGM requires computation of $((1-SSIM\{C_i, C_{i-1}\}(m,n)) \cdot (1-W_i(m,n))) \leq T$ for each pixel of each subsequent frame. This computation is not required by GM. Evaluation of $((1-SSIM\{C_i, C_{i-1}\}(m,n)) \cdot (1-W_i(m,n))) \leq T$ requires computation

of $SSIM\{C_i, C_{i-1}\}(m,n)$, and $W_i(m,n)$ for each pixel. Furthermore, to evaluate this expression, two subtraction operations, one multiplication operation, and one comparison operation are also required for each pixel. Based on the result of the evaluation of $((1 - SSIM\{C_i, C_{i-1}\}(m,n)) \cdot (1 - W_i(m,n))) \leq T$, an assignment operation may also be needed. However, since the outcome of the evaluation of $((1 - SSIM\{C_i, C_{i-1}\}(m,n)) \cdot (1 - W_i(m,n))) \leq T$ is content dependent, the (one) possible assignment operation at each pixel can either be ignored or a worst case of this operation taking place at each pixel can be assumed. Let us assume that this assignment operation is done at each pixel. As discussed in Section 3.3.1, the generation of the similarity map $SSIM\{C_i, C_{i-1}\}$ requires $MN(12X+18)$ computational operations and the generation of W_i requires a total of $MN(4Y+1)$ computational operations. Also, recall from Section 3.2.1, that starting from second output frame, computation of a FIOD halftone for each frame is also needed to prepare the initial estimate of the output (final) halftone frame. Computation of a FIOD halftone frame requires MN thresholding operations. Adding all these results in a total of approximately $(I-1)(6MN + MN(4Y+1) + MN(12X+18))$ extra operations for MGM, when compared to GM. Thus, MGM is computationally more expensive than GM.

Let us compare memory usage of GM and MGM under the same assumptions that were made for comparing memory usage of FIFSED and FDFSED algorithms.

In GM, assuming that the HVS filter size is RxR. Then, to store the autocorrelation matrix (autocorrelation matrix of the HVS filter) in the implementation based on [64], $(2R-1)(2R-1)$ pixels are needed. To store the cross-correlation ma-

trix (cross-correlation between the error image and the HVS filter), MN pixels are needed [64]. The initial estimate for the output halftone requires MN pixels (under the simplistic assumption that the binary halftone pixel requires the same storage space as a continuous-tone pixel). The initial halftone frame estimate changes iteratively and, eventually, converges to halftone to be output. So, GM requires about $2MN+(2R-1)(2R-1)$ pixels storage space to produce each output halftone frame.

In MGM, the first frame gets produced the same way as the first frame for GM. However, for each subsequent frame, the initial estimate for the halftone frame is determined differently. To generate the initial estimate for the output halftone frame, MN pixels are needed to store the FIOD halftone of current frame, MN pixels are needed to store the previous halftone frame (which is modified to form the initial estimate for current halftone frame), MN pixels are needed to store the map $SSIM\{C_i, C_{i-1}\}$, and MN pixels are needed to store the map W_i. The memory requirement for producing the initial estimate is then $4MN$ pixels. Once the initial estimate has been generated, the memory requirement for MGM becomes the same as GM, i.e. $2MN+(2R-1)(2R-1)$ pixels. Thus, MGM requires at most $4MN$ pixels of storage space to generate an output halftone frame, which is larger (assuming the support of the HVS filter is smaller than the image dimensions) than the requirement for GM.

3.3.3 Computational Complexity of FIOD

The previous two sections discussed computational complexity of four video halftone generation algorithms. In this section, I briefly discuss the computational

complexity of the FIOD video halftone generation method.

Note that FIOD is an example of a point process, and recall that point processes are computationally least expensive. To produce an output frame, FIOD requires MN thresholding operations. This means that a total of MNI operations are needed to compute the FIOD video halftone.

For memory usage, I assume that the current input continuous-tone frame is stored and so is the threshold array. The total storage is thus $2MN$ pixels, assuming that the output halftone is not stored and output to the display device directly.

3.4 Summary

This chapter presents the design of two new video halftoning algorithms. The video halftoning algorithms have been designed to produce binary videos to be displayed at frame rates ranging between 15 to 30 fps. The algorithm design is based on the temporal artifact evaluation framework developed in Chapter 2. An iterative algorithm is designed with the aim to produce video halftones with reduced DWE. A neighborhood based algorithm is designed with the aim to produce video halftones with reduced flicker. Performance of the two algorithms is objectively evaluated using the artifact assessment framework of Chapter 2. Results of visual inspection are also presented and discussed. Finally, a relative comparison of computation requirements of five video halftone generation algorithms was presented.

Chapter 4

Power Analysis

Chapters 2 and 3 focused on video halftone temporal artifact assessment and generation of video halftones, respectively. In this chapter, I focus on the typical display platform that might benefit from the contributions discussed in Chapters 2 and 3. The typical requirements of a handheld display device were discussed in Section 1.1.2.3. These included increased portability, longer battery life, and higher perceptual quality display of multimedia content. This chapter provides a "link" between power consumption and amount of temporal artifacts present in the halftone video displayed on a bistable display device.

The chapter introduces to the reader the impact of display on the overall power consumption of the portable multimedia device. The advantages of bistable display technology over conventional display technologies are discussed. This discussion is followed by the introduction of a simple model that will be utilized to analyze the power consumption of bistable display component of a multimedia device. Finally, a comparison of the power performance of the five video halftoning algorithms, already discussed in this dissertation, is presented.

With advancements in semiconductor and wireless communication technology, transmission of high-bandwidth data has been becoming increasingly feasi-

ble [112]. As a result, mobile phones are no longer just used for voice communications [113]. Users of mobile multimedia devices expect to see high quality images and high quality real-time videos on their mobile devices. This user requirement translates to having a mobile device with powerful processing capability. With advancements in semiconductor and processor technologies, using a higher processing capability processor in a mobile device is not a big problem [112]. Power is an issue though. A powerful multimedia processor requires more power. Such a processor might drain the battery of the mobile multimedia device quicker than a less powerful processor. This would mean that the user would need to recharge the battery more frequently, something that a user would not really want! Another solution would be to use a larger battery. A larger battery could increase the overall size and weight of the mobile multimedia device. A heavier and/or bulky mobile device is something that most users would rather not have. Portability is an important requirement for any handheld device.

Although a powerful processor requires more power, in a typical portable multimedia system, it is not its processor that consumes most power, it is the display system that drains the battery most [114, 115]! In the past, Liquid Crystal Display (LCD) technology has been the dominant display technology for portable communication devices [116]. LCD technology continues to be typical choice for mobile phone displays [115]. LCD displays consume maximum power in portable multimedia devices and, hence, the battery life is determined more by its use than the use of any other mobile device component [115]. To get a better idea of how LCD display affects the battery while viewing video, in the following section, I will

discuss the power analysis reported in [114].

4.1 Power Consumption of a Typical Back-lit LCD Display Handheld Embedded System

An LCD display panel requires a light source for operation [115]. Transmissive LCD display systems utilize a backlight source. Reflective LCD displays utilize ambient light, but do not produce high quality display [115]. Transflective LCD displays, that both reflect and transmit light, also require a backlight for operation. For backlighting source, cold cathode fluorescent lamp (CCFL) or a white light emitting diode (LED) are common choices for thin film transistor (TFT) LCDs [115].

Choi *et al.* estimated the typical power distribution in a handheld device using a color TFT LCD display [114]. The reference platform for their report used a 32-bit RISC processor running at 206 MHz, and was equipped with a 640x480 color TFT LCD display. Power usage by major components of this platform was estimated while running an MPEG-4 player. Figure 4.1 depicts these power statistics based on the values reported in [114]. It can be readily observed that the display components consumed the most power. Specifically, LCD backlight was the highest power consumer (29.8%). The LCD panel (21.6%) and the frame buffer (13.2%) consumed the second and third most power, respectively.

MPEG-4 playback is computationally intensive, however. Choi *et al.* further report that while running a document viewer or a word processor approximately all the energy was consumed by the display components [114].

Some of the techniques to reduce power consumption of the LCD display

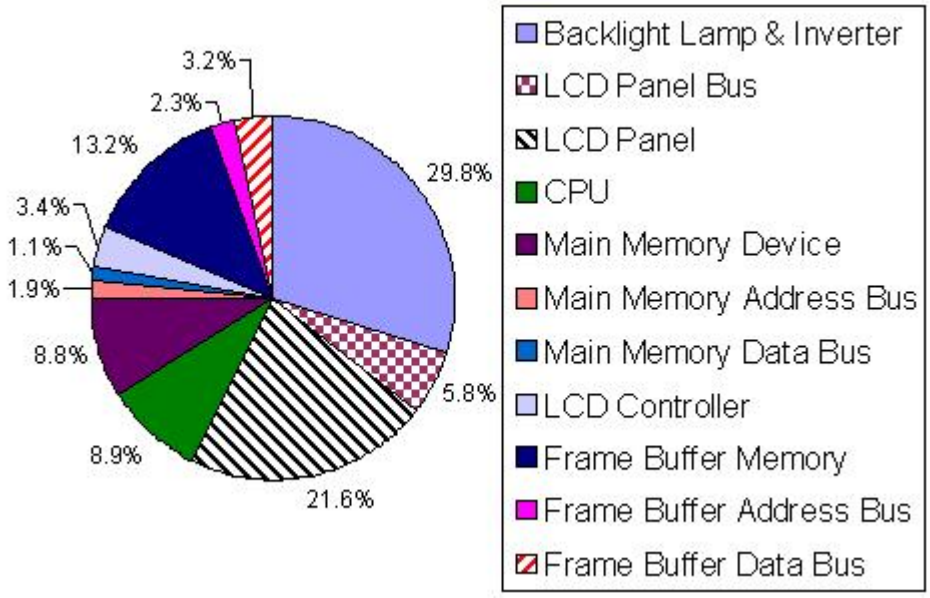


Figure 4.1: Power distribution among major components of a typical color TFT LCD based handheld embedded system during MPEG-4 video display.

systems include [114, 117–120]. Of these, methods to reduce power consumption by the backlighting system have been discussed in [114, 117]. Shim *et al.* proposed frame buffer compression to reduce display system power consumption [118]. Another technique for frame buffer compression was introduced in [119]. A reduction in frame buffer power consumption has also been proposed in [120].

4.2 Power Advantage of Bistable Display Technology

If the power consumed by the display system could be reduced, the power savings thus gained could increase the battery life of a portable multimedia device. As the name suggests, a bistable display system has its pixels "stable" in one of the two states. Each pixel (or sub-pixel) is either in an "on" or "off" state. Retaining

a pixel in its current state requires no power [121]. Power is required only to switch the pixel state. This means that once an image is displayed on a device with bistable display, it takes no power to continue to display the image. This is not the case in a conventional LCD display system, where a periodic refresh of the display screen is required to continue displaying the same image [121]. Bistable displays are hence very power efficient. These displays have recently been shown to be reliable too. On a particular bistable display system, the pixels have been reported to have retained their state, and hence the displayed image, for more than five days after the power was turned off [122].

As discussed in Section 4.1, backlighting sources in LCDs can consume a lot of power. Power savings can thus be gained by eliminating the need for a backlight. A reflective display utilizes ambient lighting to display an image, and, therefore, requires no backlighting source. Reflective display devices thus are relatively power efficient. A electrophoretic display (EPD) is an example of a reflective display [123–125]. EPDs also offer the additional benefits of bistability, wide-viewing angle and high contrast ratio [126]. Another class of bistable reflective displays utilize the bistable reflective cholesteric liquid crystal display (ChLCD) technology [127].

4.3 Estimation of Power Consumed by the Bistable Display Component

As discussed in Section 2.2.1, during a binary video display, certain display devices' power consumption may be dependent on flicker [1]. This can, for example, be the case with display devices having bistable pixels. In such a case, more power is

consumed every time the state of a pixel is changed, whereas no power is required to maintain a pixel state. This fact establishes a relationship between flicker and power consumption. Power analysis presented in this chapter is applicable to devices whose power consumption is dependent on flicker present in a binary video halftone. The results presented in this chapter indicate a relationship between power consumption and the amount of temporal artifacts present in the binary halftone video.

I make the following assumptions on the power behavior of the bistable display device:

1. No power is consumed in maintaining the pixel state;
2. α Watts is consumed in turning a pixel to the bright or on state (represented by a binary 1);
3. β Watts is consumed in turning a pixel to the dark or off state (represented by a binary 0);

Let A_i be the number of pixels that change to binary 1 in the i^{th} frame. Let B_i be the number of pixels that change to binary 0 in the i^{th} frame. Let $P_i(m, n)$ be the power consumed for the pixel located at spatial coordinates (m, n) in the i^{th} frame. Let P_i be the power consumed in displaying the i^{th} frame of the binary halftone video. Then, I have

$$P_i = \alpha \cdot A_i + \beta \cdot B_i. \tag{4.1}$$

For simplicity, I assume that $\alpha = \beta = 1$. Therefore,

$$P_i = A_i + B_i. \tag{4.2}$$

For the i^{th} halftone frame ($i > 1$), the average power consumed is

$$\widehat{P}_i = \frac{\sum_m \sum_n P_i(m,n)}{M \cdot N}. \tag{4.3}$$

For the entire halftone video, I define the Power Index P of a halftone video V_d as

$$P = \frac{\sum_i \widehat{P}_i}{(I-1)} \text{ for } i > 1 \ . \tag{4.4}$$

With this development, I am now ready to compare different algorithms in terms of their power performance and the degree of temporal artifacts present in them. The next section presents this comparison.

4.4 Power, Flicker, and DWE Comparisons

This section presents a comparison of five different halftoning algorithms. The video halftoning algorithms include frame-independent Floyd-Steinberg error diffusion (FIFSED), frame-dependent Floyd-Steinberg error diffusion (FDFSED) introduced in Section 3.2.3, Gotsman's iterative method (GM) [49], Modified Gotsman's method (MGM) introduced in Section 3.2.1, and frame-independent ordered-dither (FIOD). The characteristics compared include the Power Index, P, the Flicker Index, F, and the DWE Index, DWE. These characteristics are evaluated for a total of twenty five halftone videos, including nine 30 fps halftone videos, nine 15 fps halftone videos, and seven 25 fps videos.

Tables 4.1 through 4.25 depict a comparison of halftone videos. Each table presents the power, flicker, and dirty-window-effect performance of five halftoning methods. Figures 4.2, 4.3, and 4.4 graphically show the flicker, power, and DWE

performance, respectively, of different halftoning algorithms for 30 fps videos. Figures 4.5, 4.6, and 4.7 graphically show the flicker, power, and DWE performance, respectively, of different halftoning algorithms for 15 fps videos. Figures 4.8, 4.9, and 4.10 graphically show the flicker, power, and DWE performance, respectively, of different halftoning algorithms for 25 fps videos.

From the data presented in the tables and the figures, observe the general relationship between the Flicker Index, F and the DWE Index, DWE. Increase in one temporal artifact typically results in a decrease of the other temporal artifact. Table 4.26 shows the correlation coefficients between F and P, DWE and P, and F and DWE for the nine sets of 30 fps halftone videos, where each set of videos was generated using five different halftoning methods (FIFSED, FDFSED, GM, MGM, and FIOD) on each of the nine 30 fps continuous-tone videos. Table 4.27 shows the correlation coefficients between F and P, DWE and P, and F and DWE for the nine sets of 15 fps halftone videos, where each set of videos was generated using five different halftoning methods (FIFSED, FDFSED, GM, MGM, and FIOD) on each of the nine 15 fps continuous-tone videos. Table 4.28 shows the correlation coefficients between F and P, DWE and P, and F and DWE for the seven sets of 25 fps halftone videos, where each set of videos was generated using five different halftoning methods (FIFSED, FDFSED, GM, MGM, and FIOD) on each of the seven 25 fps continuous-tone videos. Based on the data presented in this section, one could also deduce a clear correlation between the Flicker Index, F, and the Power Index, P.

Cases of either extreme are those of the halftone videos generated by FIFSED

Table 4.1: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Caltrain Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.427	0.333	0.092
FDFSED	0.178	0.131	0.128
GM	0.067	0.048	0.151
MGM	0.125	0.092	0.139
FIOD	0.035	0.024	0.156

Table 4.2: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Tempete Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.34	0.266	0.042
FDFSED	0.146	0.108	0.053
GM	0.068	0.048	0.058
MGM	0.092	0.066	0.055
FIOD	0.037	0.025	0.062

and FIOD algorithms. FIFSED halftone videos have the highest value of the Flicker Index, F (Figures 4.2, 4.5, and 4.8) and the lowest value of the DWE Index, DWE (Figures 4.4, 4.7, and 4.10). Figures 4.3, 4.6, and 4.9 depict that FIFSED halftone videos also consume the most power. This observation points out a direct relationship between flicker and power. Higher value of F implies a higher value of P. On the other hand, FIOD halftone videos, have the lowest value of the Flicker Index, F (Figures 4.2, 4.5, and 4.8)and the highest value of the DWE Index, DWE (Figures 4.4, 4.7, and 4.10). Figures 4.3, 4.6, and 4.9 depict that FIOD halftone videos also consume the least power.

Table 4.3: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Miss America Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.3	0.262	0.044
FDFSED	0.094	0.081	0.058
GM	0.013	0.011	0.065
MGM	0.04	0.033	0.062
FIOD	0.013	0.011	0.065

Table 4.4: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Susie Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.456	0.4	0.043
FDFSED	0.128	0.105	0.064
GM	0.06	0.047	0.071
MGM	0.068	0.053	0.07
FIOD	0.02	0.015	0.077

Table 4.5: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Tennis Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.436	0.344	0.066
FDFSED	0.140	0.096	0.096
GM	0.062	0.036	0.11
MGM	0.08	0.049	0.104
FIOD	0.035	0.019	0.115

Table 4.6: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Trevor Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.366	0.31	0.027
FDFSED	0.112	0.092	0.038
GM	0.03	0.023	0.042
MGM	0.043	0.033	0.041
FIOD	0.015	0.012	0.044

Table 4.7: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Garden Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.408	0.232	0.127
FDFSED	0.262	0.134	0.157
GM	0.168	0.082	0.18
MGM	0.213	0.107	0.171
FIOD	0.104	0.048	0.198

Table 4.8: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Salesman Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.361	0.319	0.026
FDFSED	0.092	0.081	0.037
GM	0.009	0.007	0.04
MGM	0.024	0.02	0.039
FIOD	0.013	0.011	0.04

Table 4.9: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 30 fps Football Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.457	0.329	0.087
FDFSED	0.196	0.123	0.116
GM	0.181	0.108	0.113
MGM	0.127	0.074	0.127
FIOD	0.061	0.032	0.143

Table 4.10: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Caltrain Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.429	0.3	0.134
FDFSED	0.204	0.131	0.18
GM	0.131	0.083	0.202
MGM	0.139	0.089	0.2
FIOD	0.048	0.028	0.225

Table 4.11: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Tempete Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.358	0.254	0.079
FDFSED	0.18	0.117	0.098
GM	0.099	0.064	0.111
MGM	0.142	0.091	0.104
FIOD	0.056	0.033	0.118

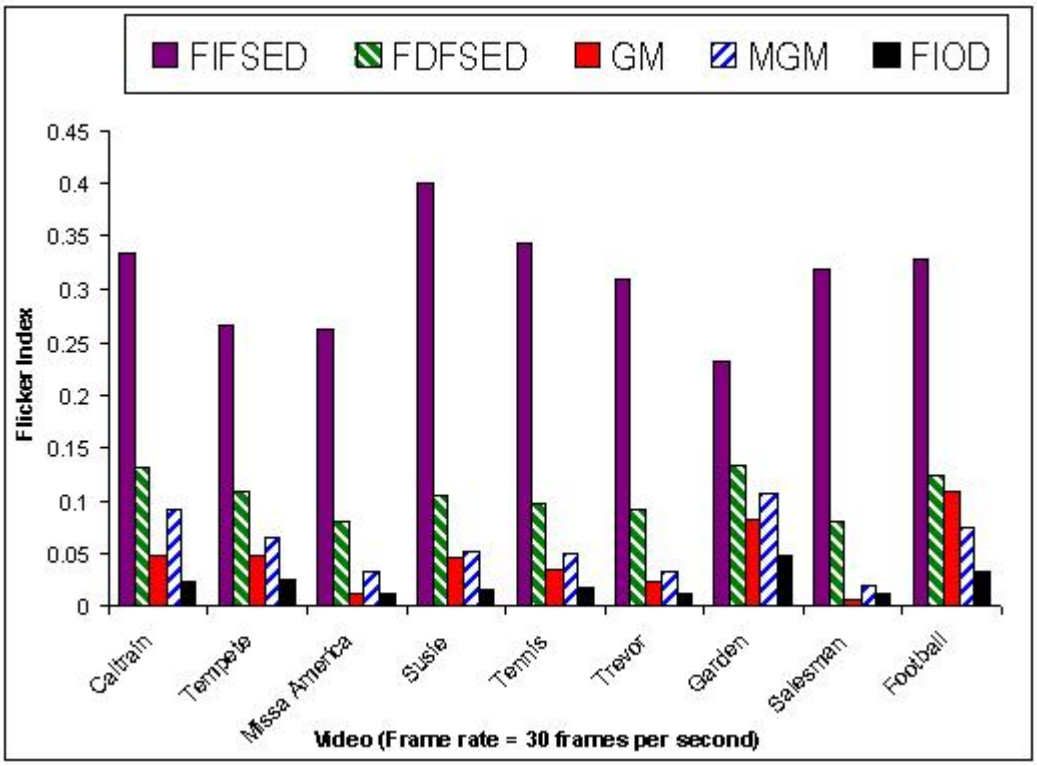


Figure 4.2: The Flicker Index for different 30 fps halftone videos.

Table 4.12: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Miss America Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.299	0.267	0.036
FDFSED	0.091	0.079	0.046
GM	0.026	0.02	0.049
MGM	0.035	0.028	0.049
FIOD	0.013	0.011	0.052

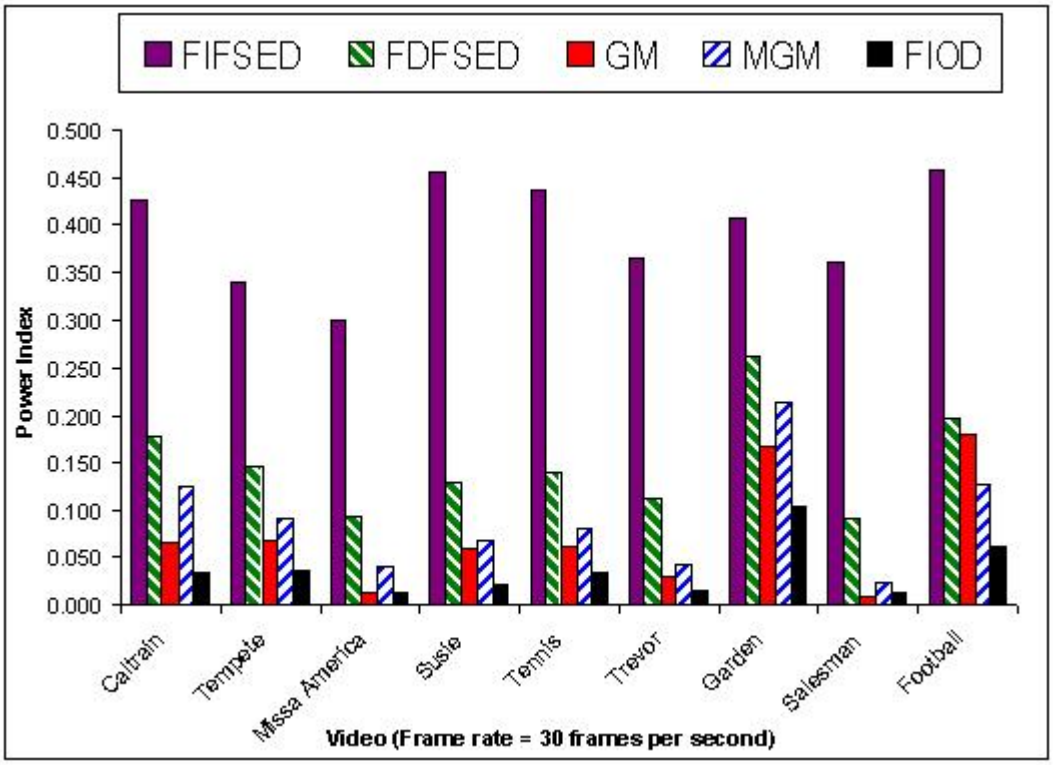


Figure 4.3: The Power Index for different 30 fps halftone videos.

Table 4.13: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Susie Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.458	0.385	0.063
FDFSED	0.146	0.111	0.091
GM	0.103	0.077	0.096
MGM	0.092	0.068	0.099
FIOD	0.03	0.021	0.11

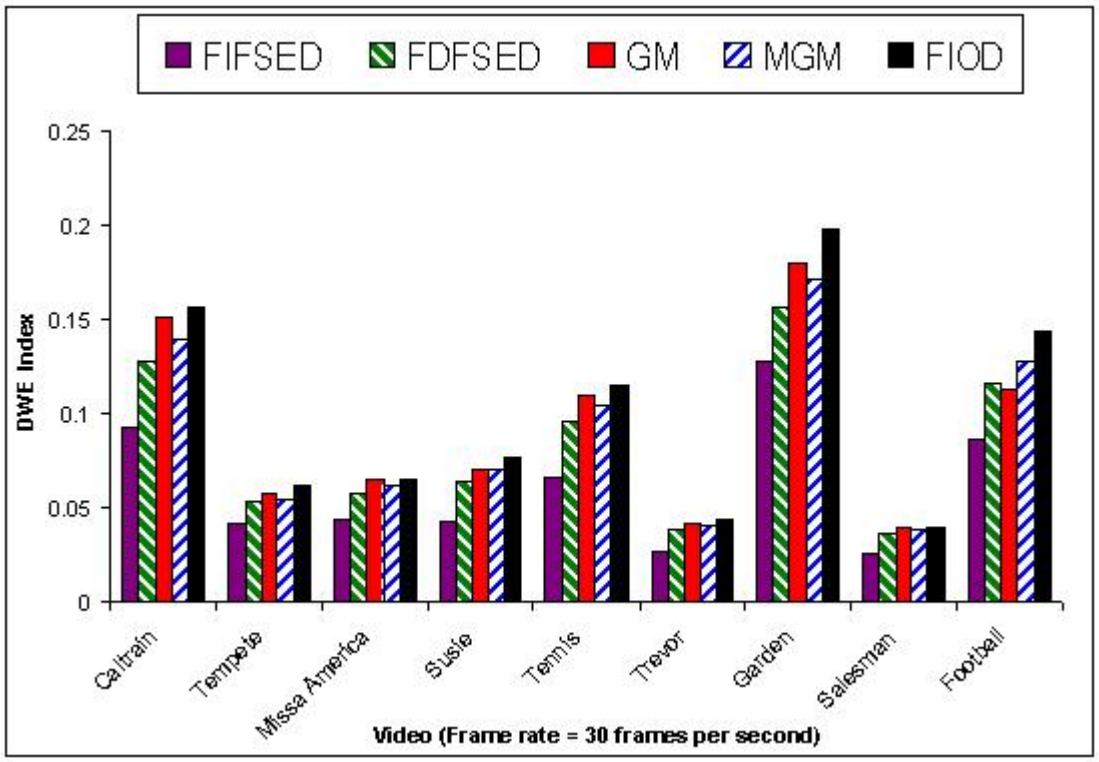


Figure 4.4: The DWE Index for different 30 fps halftone videos.

Table 4.14: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Tennis Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.444	0.33	0.08
FDFSED	0.157	0.099	0.115
GM	0.102	0.055	0.126
MGM	0.099	0.056	0.125
FIOD	0.046	0.023	0.138

Table 4.15: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Trevor Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.367	0.301	0.042
FDFSED	0.122	0.094	0.058
GM	0.054	0.039	0.063
MGM	0.06	0.043	0.063
FIOD	0.02	0.014	0.069

Table 4.16: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Garden Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.421	0.211	0.16
FDFSED	0.294	0.13	0.191
GM	0.257	0.113	0.204
MGM	0.238	0.103	0.21
FIOD	0.133	0.054	0.244

Table 4.17: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Salesman Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.357	0.323	0.011
FDFSED	0.084	0.074	0.015
GM	0.017	0.013	0.016
MGM	0.017	0.013	0.016
FIOD	0.008	0.006	0.018

Table 4.18: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 15 fps Football Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.468	0.314	0.109
FDFSED	0.22	0.129	0.147
GM	0.236	0.136	0.138
MGM	0.158	0.089	0.161
FIOD	0.08	0.041	0.181

Table 4.19: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Pedestrian-area Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.388	0.323	0.051
FDFSED	0.142	0.108	0.065
GM	0.12	0.084	0.061
MGM	0.067	0.046	0.071
FIOD	0.033	0.021	0.077

Table 4.20: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Rush-hour Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.383	0.329	0.027
FDFSED	0.137	0.114	0.037
GM	0.068	0.054	0.039
MGM	0.066	0.052	0.04
FIOD	0.019	0.014	0.044

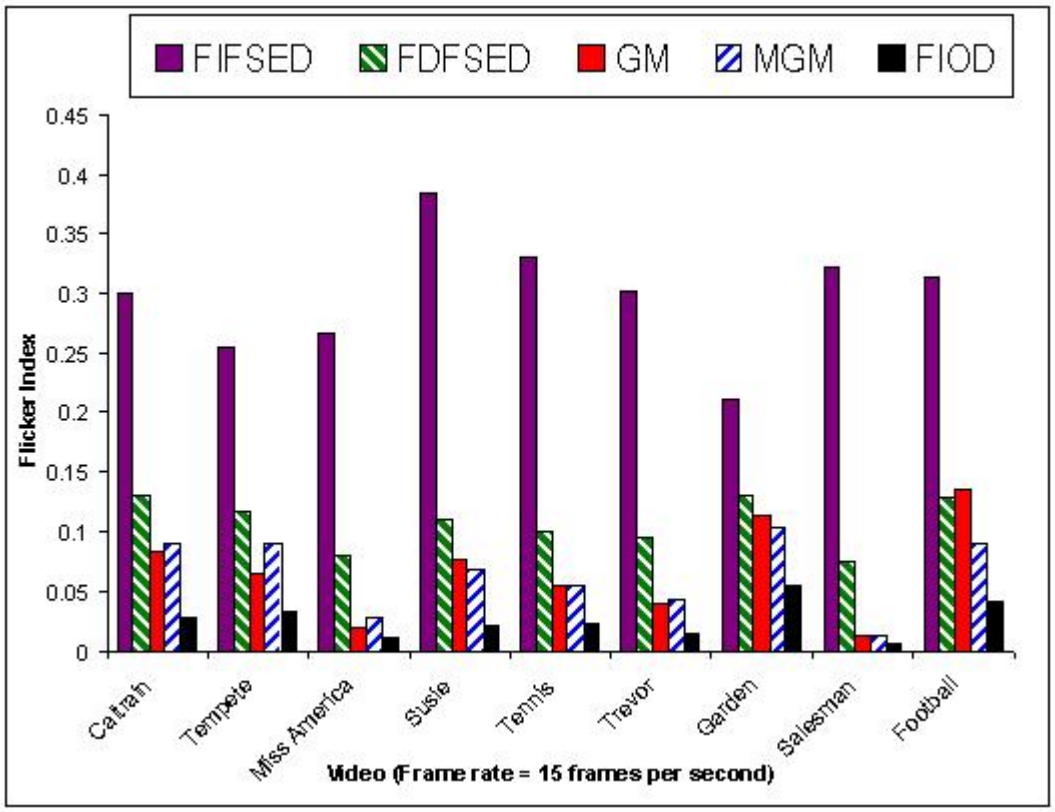


Figure 4.5: The Flicker Index for different 15 fps halftone videos.

Table 4.21: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Sunflower Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.339	0.261	0.07
FDFSED	0.165	0.121	0.086
GM	0.12	0.083	0.088
MGM	0.106	0.076	0.092
FIOD	0.037	0.025	0.102

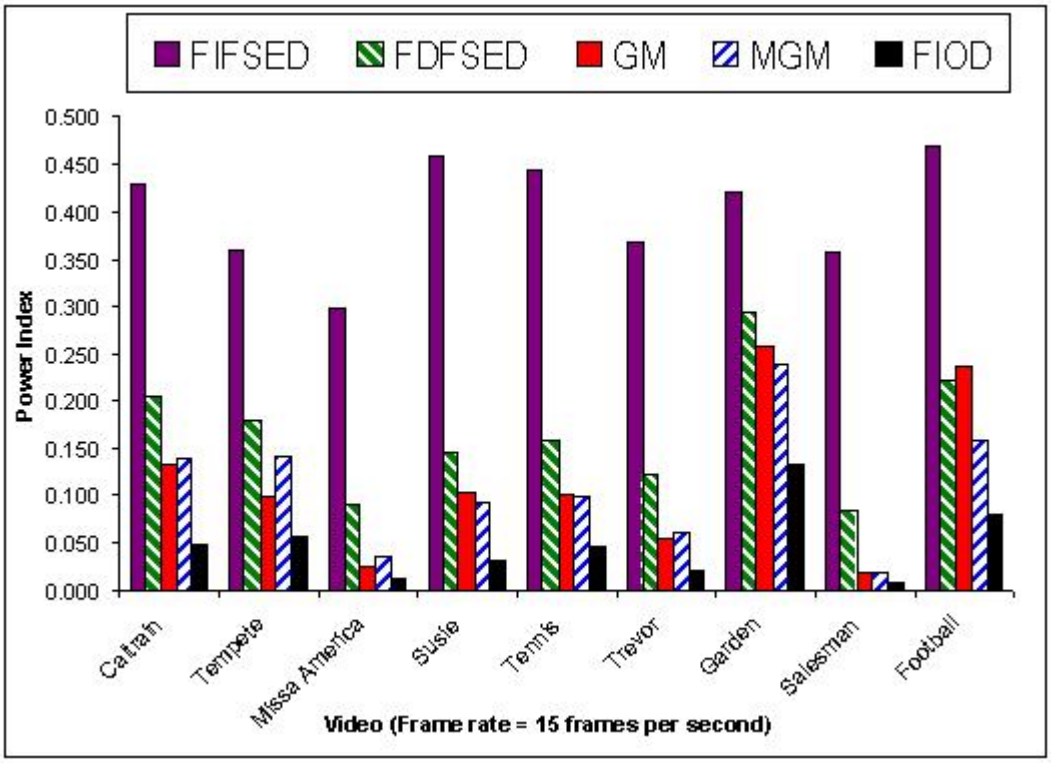


Figure 4.6: The Power Index for different 15 fps halftone videos.

Table 4.22: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Shields Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.32	0.211	0.152
FDFSED	0.186	0.114	0.181
GM	0.149	0.087	0.188
MGM	0.137	0.082	0.195
FIOD	0.067	0.037	0.214

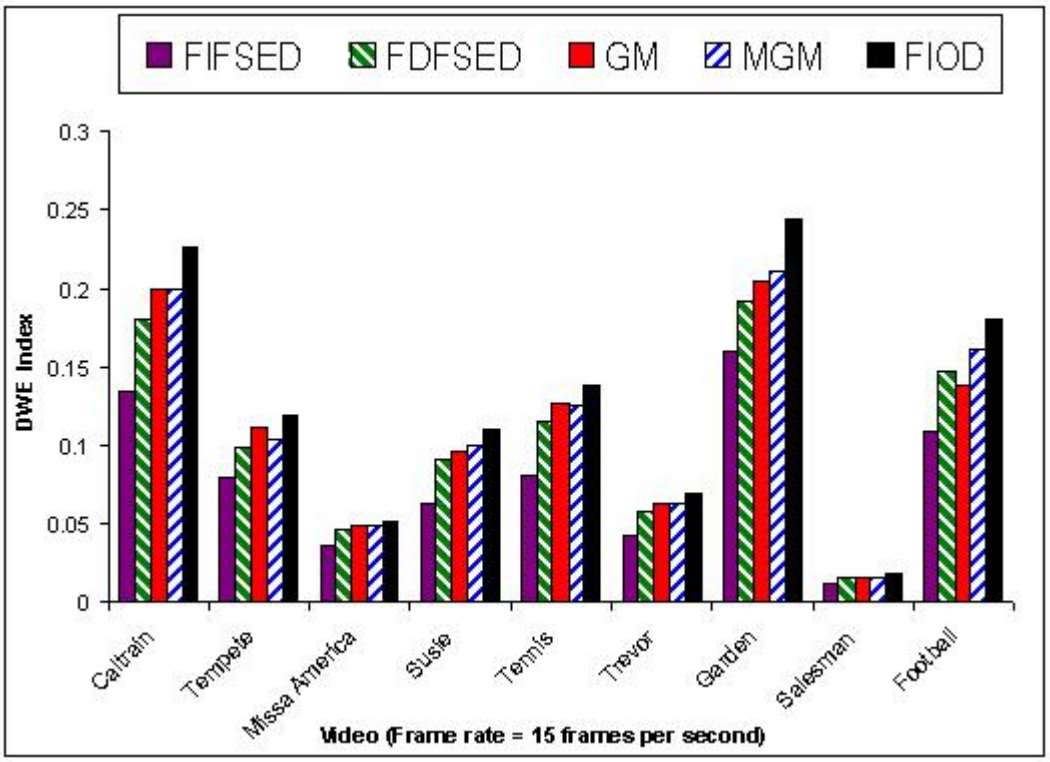


Figure 4.7: The DWE Index for different 15 fps halftone videos.

Table 4.23: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Blue-sky Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.301	0.191	0.112
FDFSED	0.18	0.095	0.126
GM	0.157	0.075	0.127
MGM	0.127	0.06	0.134
FIOD	0.071	0.031	0.148

Table 4.24: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Station Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.381	0.302	0.055
FDFSED	0.155	0.115	0.072
GM	0.03	0.021	0.084
MGM	0.071	0.051	0.079
FIOD	0.02	0.014	0.086

Table 4.25: Comparison of the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for the 25 fps Tractor Sequence.

Halftoning Method	P	F	DWE
FIFSED	0.417	0.261	0.127
FDFSED	0.234	0.139	0.168
GM	0.213	0.127	0.173
MGM	0.157	0.094	0.189
FIOD	0.06	0.034	0.214

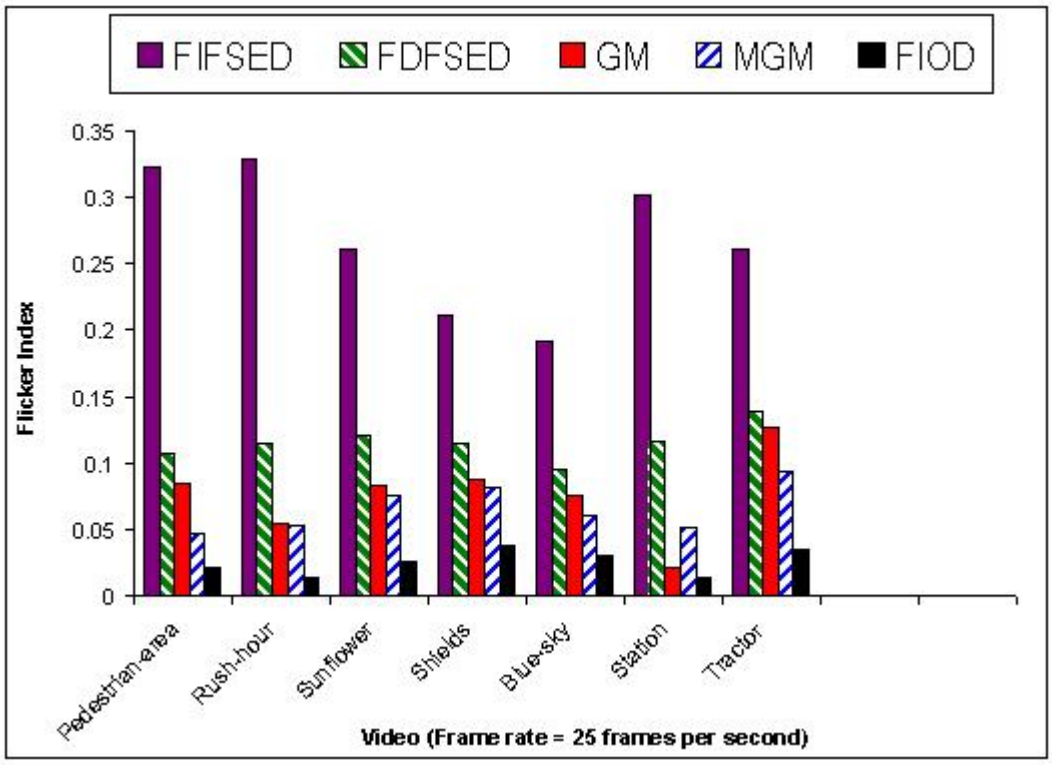


Figure 4.8: The Flicker Index for different 25 fps halftone videos.

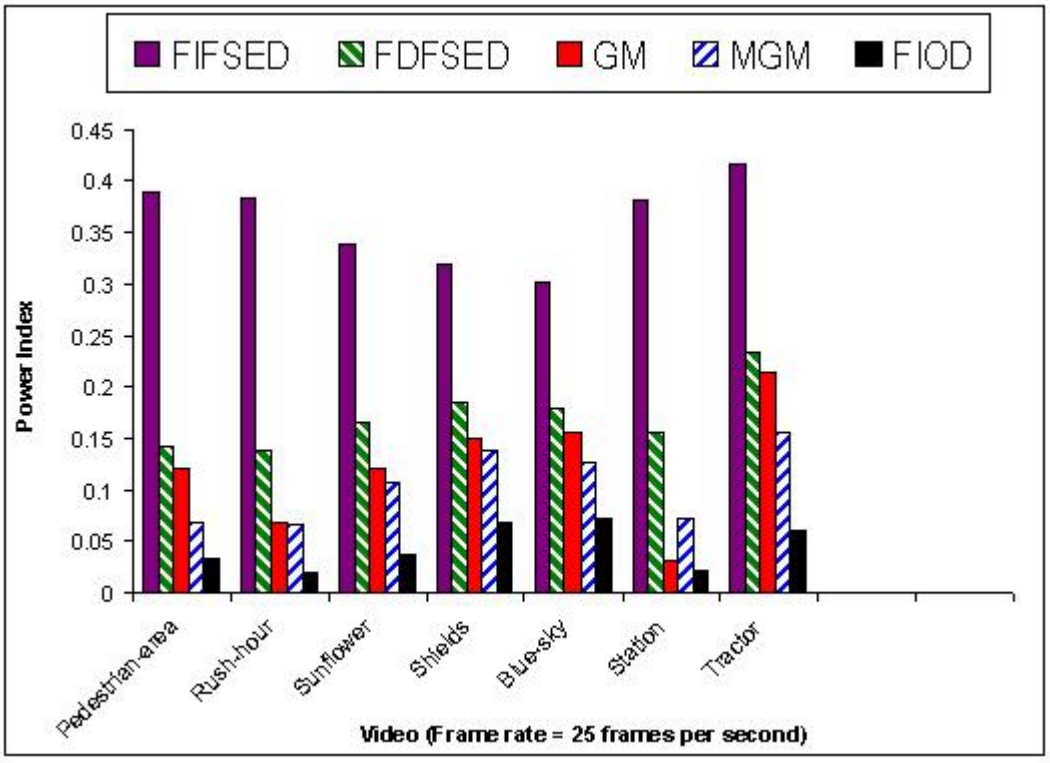


Figure 4.9: The Power Index for different 25 fps halftone videos.

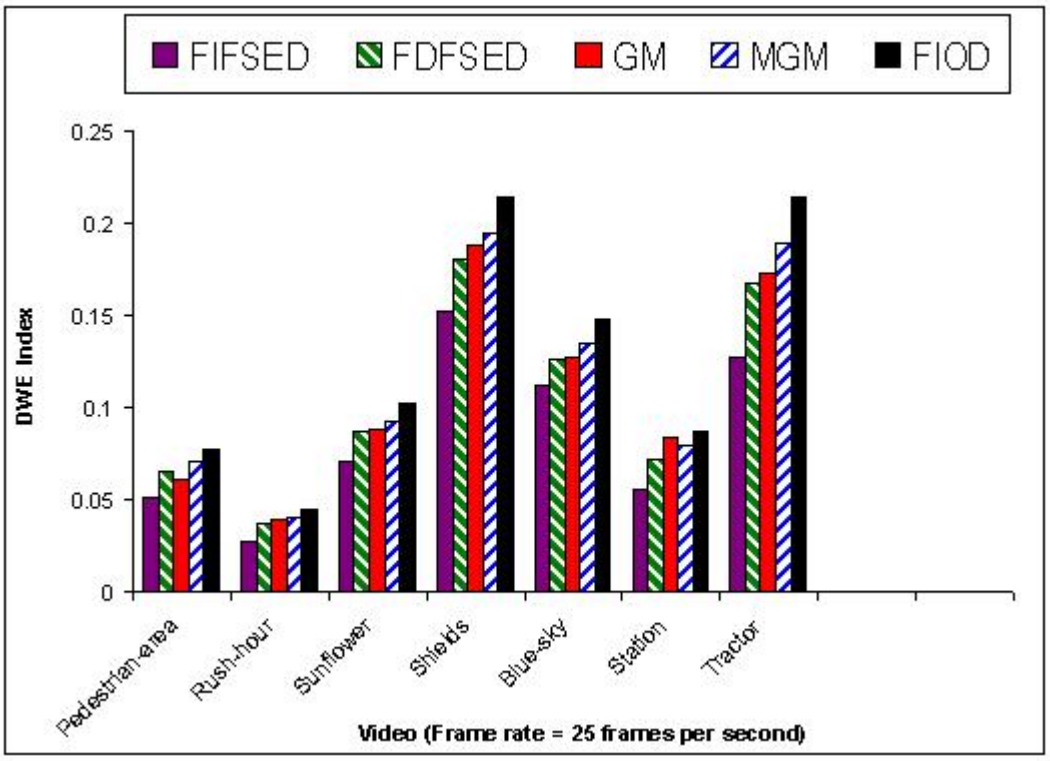


Figure 4.10: The DWE Index for different 25 fps halftone videos.

Table 4.26: Correlation coefficients between the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for nine 30 fps halftone video sets. Each set was generated using five video halftoning methods.

Sequence	**Corr. Coeff. Between** F & P	**Corr. Coeff. Between** DWE & P	**Corr. Coeff. Between** F & DWE
Caltrain	1	-0.997	-0.995
Tempete	1	-0.984	-0.982
Miss America	1	-0.998	-0.998
Susie	1	-0.99	-0.988
Tennis	1	-0.989	-0.985
Trevor	1	-0.995	-0.994
Garden	0.998	-0.998	-0.992
Salesman	1	-0.999	-0.999
Football	0.999	-0.965	-0.95

Table 4.27: Correlation coefficients between the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for nine 15 fps halftone video sets. Each set was generated using five video halftoning methods.

Sequence	**Corr. Coeff. Between** F **&** P	**Corr. Coeff. Between** DWE **&** P	**Corr. Coeff. Between** F **&** DWE
Caltrain	0.999	-0.995	-0.992
Tempete	0.999	-0.993	-0.988
Miss America	1	-0.989	-0.987
Susie	1	-0.987	-0.982
Tennis	0.999	-0.991	-0.986
Trevor	1	-0.989	-0.986
Garden	0.996	-0.996	-0.984
Salesman	1	-0.952	-0.951
Football	0.998	-0.976	-0.960

Table 4.28: Correlation coefficients between the Power Index, P, the Flicker Index, F, and the DWE Index, DWE for seven 25 fps halftone video sets. Each set was generated using five video halftoning methods.

Sequence	**Corr. Coeff. Between** F **&** P	**Corr. Coeff. Between** DWE **&** P	**Corr. Coeff. Between** F **&** DWE
Pedestrian-area	0.999	-0.924	-0.909
Rush-hour	1	-0.986	-0.984
Sunflower	1	-0.984	-0.978
Shields	0.999	-0.994	-0.988
Blue-sky	0.993	-0.967	-0.931
Station	1	-0.995	-0.993
Tractor	1	-0.998	-0.996

4.4.1 Power and Flicker

Tables 4.1 through 4.25 exhibit similar rank ordering for the Power Index and the Flicker Index. This section attempts to answer the question: Does the Power Index, P, give us the same information as the Flicker Index, F? To prove that the Power Index, P, does not give us the same information as the Flicker Index, F, I use example cases. These examples serve to be the counter examples that show that P is different than F, and that P is not a perceptual measure for flicker.

In Chapter 2, it was shown that the Flicker Index, F is an estimate of the *perceived* flicker. The Power Index, P is not a perceptual measure. It is based on average power $\widehat{P}_i$ as determined in (4.3). Note that the average power $\widehat{P}_i$, as determined in this chapter, is equivalent to the flicker measure, average flicker rate (AFR), used by Hsu *et al.* in [66]. It was argued in Section 2.2.1 that AFR is not a perceptual measure because it does not incorporate the masking properties of the HVS. This is shown through examples that follow.

Three halftone video pairs were formed for playback at 30 fps. Another three halftone video pairs were formed for playback at 15 fps. The videos were halftoned using FIFSED. Each pair had two videos that differed in content. This was needed to emphasize the dependence of content on the perception of flicker and, hence, to demonstrate the need for a perceptual quality assessment measure. These six video pairs were viewed by ten human viewers on an LCD screen. The videos were viewed indoors under artificial lighting conditions. The viewers were advised to keep the viewing distance between 18 to 36 inches. Each viewer was asked to view each pair simultaneously and then indicate which video (of the pair) exhibited higher flicker.

No time constraints were imposed. The viewers were allowed to take as much time as was needed to determine the video with the higher flicker in each pair. Viewers were allowed to replay (or play in a continuous loop) the videos as needed.

Table 4.29 shows the details of videos used to form the pairs. Please refer to Tables 2.1 and 2.2 for the *spatial* resolution of each of the videos used to form the pairs. Note that for each video pair, the orderings of F, and P do not agree. Table 4.30 depicts the results of the visual inspection experiment described in this section. It is clear that the results of the visual inspection experiment (i.e. the responses of the viewers shown in Table 4.30) agree more with the flicker comparison done using the Flicker Index, F (in Table 4.29). The aggregate results of Table 4.30, however, do not agree with the flicker comparison done using the Power Index, P (in Table 4.29). All 10 viewers, who viewed the six video pairs, indicated that:

- flicker in the 30 fps Susie sequence was higher than flicker in the 30 fps Football sequence;
- flicker in the 30 fps Salesman sequence was higher than flicker in the 30 fps Garden sequence;
- flicker in the 30 fps Trevor sequence was higher than flicker in the 30 fps Garden sequence;
- flicker in the 15 fps Salesman sequence was higher than flicker in the 15 fps Garden sequence;

- flicker in the 15 fps Trevor sequence was higher than flicker in the 15 fps Garden sequence;

This is consistent with the flicker comparison done using the Flicker Index, F. This is, however, not consistent with the flicker comparison done using the Power Index, P. Of the ten observers, who viewed the 15 fps Susie-Football halftone pair, only two observers' responses disagree with the flicker comparison done using the Flicker Index, F, while the responses of the remaining eight observers agree with the flicker comparison done using the Flicker Index, F.

Based on the findings of this section, it is concluded that the Power Index, P, does not give us the same information as the Flicker Index, F, and that the Flicker Index, F, gives a better objective measure of the *perceived* flicker in medium frame rate binary halftone videos.

Table 4.29: Video pairs for comparison of flicker. The videos used to form the pairs were halftoned using FIFSED.

Pair No.	**Video**	**Number of Frames**	**Frame Rate**	P	F
1	Susie	60	30 fps	0.455	0.4
	Football	60	30 fps	0.457	0.329
2	Garden	61	30 fps	0.408	0.232
	Salesman	61	30 fps	0.361	0.317
3	Garden	61	30 fps	0.408	0.232
	Trevor	61	30 fps	0.367	0.31
4	Susie	30	15 fps	0.457	0.384
	Football	30	15 fps	0.468	0.314
5	Garden	31	15 fps	0.421	0.211
	Salesman	31	15 fps	0.358	0.32
6	Garden	31	15 fps	0.421	0.211
	Trevor	31	15 fps	0.368	0.298

Table 4.30: Results of video comparisons for flicker.

Frame Rate	**Video 1**	**Video 2**	**Number of viewers indicating higher flicker in Video 1**	**Number of viewers indicating higher flicker in Video 2**
30 fps	Susie	Football	10	0
	Garden	Salesman	0	10
	Garden	Trevor	0	10
15 fps	Susie	Football	8	2
	Garden	Salesman	0	10
	Garden	Trevor	0	10

4.5 Overall Quality

The focus of quality assessment work presented in this dissertation is on perceptual assessment of two key temporal artifacts, flicker and DWE, in medium frame rate binary halftone videos. Overall quality assessment of binary video halftones is a topic that deserves a separate treatment of its own and is not the goal of this dissertation. Although overall quality assessment of binary halftone videos is not the topic of this dissertation, in this section, I discuss a preliminary approach that can be taken to assess the relative overall quality of medium frame rate binary halftone videos. This section attempts to answer the following question: What could be a possible way to assess overall quality of medium frame rate binary video halftones? The discussion in this section is only a preliminary attempt to explore one of the possible ways to form an overall quality assessment measure for medium frame rate binary video halftones. The purpose of this section is to facilitate any future research in the area of overall quality assessment of binary video halftones.

To develop a measure for overall quality of medium frame rate binary video halftones, a visual inspection experiment was carried out. The visual inspection experiment was, again, as in the case of the experiment for temporal artifact assessment, a 2AFC experiment. The results of the visual inspection experiment were used to estimate the parameters of a function that was assumed to predict the overall perceptual quality of medium frame rate binary video halftones. Since the results of the 2AFC experiment were used to estimate these parameters, the estimated parameter values are valid for the visual setup used for the 2AFC experiment. In other words, the parameter values are valid only for the screen (and other visual

conditions) used to view the halftone videos. Using a different screen or monitor might change the perception of artifacts. Therefore, estimation of the parameters for a different screen or monitor requires a visual inspection experiment using the relevant screen/monitor. Regardless, the purpose of this section is to only demonstrate a preliminary approach to estimate the overall quality of medium frame rate binary video halftones.

4.5.1 Visual Inspection for Relative Overall Quality

In this section, I discuss the visual inspection experiment conducted to evaluate the overall quality of binary video halftones viewed at medium frame rates (i.e. between 15 and 30 fps). Ten human viewers with normal or corrected-to-normal vision participated in visual inspection of binary video halftones. The viewers viewed the videos on an LCD screen in standard indoor lighting conditions. The viewers were advised to keep the viewing distance between 18 to 36 inches. As was the case for the visual inspection experiment of Section 2.3, the format of the experiment was two-alternative forced choice (2AFC). As has already been explained, in a 2AFC experiment, a viewer is asked to make a choice from a pair of stimuli. In this experiment, each displayed video simultaneously showed two different halftones (i.e. a halftone pair) of the same continuous-tone video/sequence. Each viewer was asked to determine which of the two halftone videos had a better perceptual quality. The advantages and disadvantages of 2AFC setup have already been discussed in Section 2.3. No time limits were imposed on the viewing of each video/sequence pair. Viewers were allowed to repeatedly watch the same video pair until they could

make a decision.

4.5.1.1 Videos for Visual Inspection

In the visual inspection experiment, five different video halftoning methods were evaluated. These included FIOD, GM, MGM, FIFSED, and FDFSED video halftoning algorithms. For evaluating overall halftone video quality using 2AFC, four sets of videos were used. Note that these are the same video sets that were used to evaluate flicker and DWE (Section 2.3). Each set comprised of nine videos displayed at 30 fps, nine videos displayed at 15 fps, and seven videos at 25 fps. The first set was used to compare the overall quality performance of GM versus MGM. The second set was used to compare the overall performance of GM versus FIOD. The third set was used to compare the overall performance of GM versus FIFSED. The fourth set was used to compare the overall performance of FIFSED versus FDFSED. For a description of the videos used in these sets, please refer to Tables 2.1, 2.2, and 2.3.

4.5.2 Estimating Objective Function for Overall Quality

In this section, I attempt to predict the parameters of a function that represents a possible choice for determining the overall quality of medium frame rate binary halftone videos. I assume a form of this function and use the results of the visual inspection experiment to estimate the unknown parameters of the function.

The function for overall quality assessment is assumed to be of the form given

below:

$$Q = (1 - F)^x \cdot (1 - DWE)^y \cdot S^z, \tag{4.5}$$

where Q is the overall quality of the medium frame rate binary halftone video having the Flicker Index F, the DWE Index DWE, and Spatial Quality Index S. Spatial Quality Index S is discussed in Section 5.5. However, in this section, S is modified to have its values between 0 and 1. Note for Section 5.5, I compute average of the MSSIM index [95] for all the frames in the video to get $S(V_c, V_d)$. This can result in a number between -1 and 1 inclusive. To constrain the values between 0 and 1 for determining Q in this section, I set $S = (S(V_c, V_d) + 1)/2$. Recall that $F \in [0, 1]$, and $DWE \in [0, 1]$. Therefore, $Q \in [0, 1]$.

The validity of this expression can be confirmed, if proper estimates of the three unknown parameters x, y, and z can be found. By proper estimates, I mean values of x, y, and z, that, when used in (4.5), yield objective measures that agree with the results of the overall halftone video quality visual inspection experiment discussed in this chapter.

Note that the visual inspection experiment is a 2AFC experiment. That means I have comparisons between videos rather than absolute quality measures. To estimate x, y, and z based on this data, I propose the following approach. F, DWE, and S can be calculated for each of the halftone videos in each of the pairs used in the visual inspection experiment. The experiment has given additional information as to which video of a pair is of superior quality. Say, if Q_1 represents the quality of the first video in a pair, and Q_2 represents the quality of the second video, then the visual inspection experiment tells us whether Q_1 is higher than Q_2

or not. Let's assume that $Q_1 > Q_2$ because, say, 6 viewers indicated that the first video (i.e. video 1) is of higher quality and the remaining 4 voted in favor of the second video. The degree by which the first video is superior to the second video can be quantified by stating that the quality of video 1 exceeds the quality of video 2 by $(6 - 4)$. This quantity is normalized by division by 30. That is,

$$Q_1 - Q_2 = (0.2)/3. \tag{4.6}$$

Note that (4.6) can be used to form a rank ordering of the overall quality performance for each video halftoned using the five video halftone generation algorithms. I adopt the following approach. The overall quality of each FDFSED video is assumed to be (2.6)/3 or 0.867. Note that a value of 3/3 or 1 corresponds to the best possible quality. The results of the visual inspection experiment for 30 fps halftones show that 7 users preferred Caltrain FDFSED halftone over Caltrain FIFSED halftone, while 3 users preferred Caltrain FIFSED halftone over Caltrain FDFSED halftone. So, $Q_1 - Q_2 = (0.4)/3$, following the approach depicted in (4.6). In this example, $Q_1 = (2.6)/3$ based on the assumption discussed above. Therefore, $Q_2 = Q_1 - (0.4)/3$. Thus, the overall quality of FIFSED is determined to be (2.6-0.4)/3 or 0.733. Similarly the values of Q, overall quality, for the 30 fps Caltrain GM, MGM, and FIOD halftones are determined to be 0.933, 0.8, 0.6 respectively. This process was carried out for all results from the visual inspection experiment to obtain 125 overall quality numbers, which were plugged in (4.5) to form 125 equations. Using logarithm on both sides of (4.5), a linear equation is obtained, as follows:

$$\ln Q = x \ln (1 - F) + y \ln (1 - DWE) + z \ln S. \tag{4.7}$$

Thus 125 linear equations are obtained using (4.7) and the approach discussed thus far. This system of equations was used to get the estimate values of x, y, and z that would satisfy the results of the visual inspection experiment. It was found that using the results involving FIOD halftones yields values of x, y, and z that do not agree well with the results of the visual inspection experiment. Based on the discussion on FIOD flicker evaluation in Section 2.4, this was expected for FIOD halftones. As has already been explained in Section 2.4.5, the Flicker Index, F is not as accurate for FIOD halftones. Another possible reason could be that the spatial index S may not be as accurate for FIOD halftones. Note that for FIOD images/frames, the spatial quality is lower due to the "gridding" artifact common to FIOD halftones (Section 2.1). This artifact is more pronounced, if the periodic threshold mask is smaller size. This needs to be reflected in S, but is apparently not reflected very well in the MSSIM [95] based S. In the results for this section, the mask size used for generating FIOD halftones is 32x32 pixels. Not including the FIOD results means we have a system of 100 equations that yield values of Q that agree with the visual inspection experiment results more. For these reasons, I use the 100 equations that exclude the ones that relate to the FIOD halftones. This system of linear equations is overdetermined. A least squares solution yields $x = 0.5594$, $y = 0.5796$, and $z = 0.2028$.

Tables 4.31 through 4.42 show the results of the visual inspection experiment. Tables 4.31 through 4.39 show the results of the visual inspection experiment as well

as the predicted overall quality measures for each of the halftone videos. Tables 4.31 through 4.39 show the results for 75 halftone video pairs. How many of these 75 video pairs' visual inspection results agree with the *relative* comparison done using the objective values, predicted by Q. The values of Q reported in these tables were rounded to three decimal places. In these tables, twelve (12) halftone video pairs were tied at 5 versus 5 (i.e. 5 viewers preferred one video, while the remaining 5 preferred the other video in the pair). Excluding these 12 pairs, note that Q predicts overall quality measures which, if used to compare the videos in each pair, give us comparison results that agree with 54 of the 63 results of the visual inspection experiment. This means a prediction accuracy of about 85.7%.

Table 4.31: Overall perceptual quality for 30 fps halftone videos generated using GM and MGM methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for GM**	Q **for GM**	**Number of viewers perceiving better quality for MGM**	Q **for MGM**
Caltrain	7	0.773	3	0.760
Tempete	7	0.828	3	0.823
Miss America	5	0.833	5	0.823
Susie	5	0.813	5	0.811
Tennis	3	0.804	7	0.803
Trevor	5	0.840	5	0.836
Garden	3	0.752	7	0.75
Salesman	7	0.851	3	0.845
Football	2	0.767	8	0.777

Table 4.32: Overall perceptual quality for 30 fps halftone videos generated using GM and FIFSED methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for GM**	Q **for GM**	**Number of viewers perceiving better quality for FIFSED**	Q **for FIFSED**
Caltrain	8	0.773	2	0.66
Tempete	7	0.828	3	0.733
Miss America	7	0.833	3	0.716
Susie	8	0.813	2	0.639
Tennis	7	0.804	3	0.672
Trevor	7	0.840	3	0.699
Garden	5	0.752	5	0.718
Salesman	6	0.851	4	0.696
Football	6	0.767	4	0.668

Table 4.33: Overall perceptual quality for 30 fps halftone videos generated using FIFSED and FDFSED methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for FIFSED**	Q **for FIFSED**	**Number of viewers perceiving better quality for FDFSED**	Q **for FDFSED**
Caltrain	3	0.66	7	0.748
Tempete	1	0.733	9	0.811
Miss America	3	0.716	7	0.803
Susie	2	0.639	8	0.788
Tennis	2	0.672	8	0.789
Trevor	4	0.699	6	0.809
Garden	1	0.718	9	0.752
Salesman	5	0.696	5	0.818
Football	2	0.668	8	0.762

Table 4.34: Overall perceptual quality for 15 fps halftone videos generated using GM and MGM methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for GM**	Q **for GM**	**Number of viewers perceiving better quality for MGM**	Q **for MGM**
Caltrain	8	0.731	2	0.729
Tempete	5	0.794	5	0.787
Miss America	6	0.836	4	0.832
Susie	3	0.785	7	0.789
Tennis	2	0.786	8	0.788
Trevor	4	0.822	6	0.820
Garden	2	0.726	8	0.731
Salesman	3	0.86	7	0.86
Football	2	0.741	8	0.753

Table 4.35: Overall perceptual quality for 15 fps halftone videos generated using GM and FIFSED methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for GM**	Q **for GM**	**Number of viewers perceiving better quality for FIFSED**	Q **for FIFSED**
Caltrain	6	0.731	4	0.66
Tempete	6	0.794	4	0.722
Miss America	6	0.836	4	0.717
Susie	6	0.785	4	0.64
Tennis	6	0.786	4	0.674
Trevor	5	0.822	5	0.698
Garden	5	0.726	5	0.713
Salesman	5	0.86	5	0.7
Football	4	0.741	6	0.667

Table 4.36: Overall perceptual quality for 15 fps halftone videos generated using FIFSED and FDFSED methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for FIFSED**	Q **for FIFSED**	**Number of viewers perceiving better quality for FDFSED**	Q **for FDFSED**
Caltrain	3	0.66	7	0.721
Tempete	1	0.722	9	0.783
Miss America	4	0.717	6	0.81
Susie	2	0.64	8	0.772
Tennis	1	0.674	9	0.778
Trevor	4	0.698	6	0.799
Garden	0	0.713	10	0.736
Salesman	3	0.7	7	0.832
Football	1	0.667	9	0.743

Table 4.37: Overall perceptual quality for 25 fps halftone videos generated using GM and MGM methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for GM**	Q **for GM**	**Number of viewers perceiving better quality for MGM**	Q **for MGM**
Pedestrian-area	5	0.801	5	0.815
Rush-hour	4	0.827	6	0.828
Sunflower	6	0.79	4	0.791
Shields	5	0.741	5	0.74
Blue-sky	4	0.782	6	0.786
Station	4	0.819	6	0.807
Tractor	4	0.727	6	0.734

Table 4.38: Overall perceptual quality for 25 fps halftone videos generated using GM and FIFSED methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for GM**	Q **for GM**	**Number of viewers perceiving better quality for FIFSED**	Q **for FIFSED**
Pedestrian-area	6	0.801	4	0.681
Rush-hour	6	0.827	4	0.687
Sunflower	6	0.79	4	0.708
Shields	5	0.741	5	0.704
Blue-sky	5	0.782	5	0.735
Station	6	0.819	4	0.691
Tractor	6	0.727	4	0.684

Table 4.39: Overall perceptual quality for 25 fps halftone videos generated using FIFSED and FDFSED methods. Table compares visual inspection results with the predicted overall perceptual quality objective measure Q.

Sequence	**Number of viewers perceiving better quality for FIFSED**	Q **for FIFSED**	**Number of viewers perceiving better quality for FDFSED**	Q **for FDFSED**
Pedestrian-area	2	0.681	8	0.788
Rush-hour	1	0.687	9	0.798
Sunflower	1	0.708	9	0.773
Shields	4	0.704	6	0.736
Blue-sky	2	0.735	8	0.776
Station	0	0.691	10	0.781
Tractor	3	0.684	7	0.725

Table 4.40: Overall perceptual quality visual inspection results of 2AFC between 30 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving better quality for GM**	**Number of viewers perceiving better quality for FIOD**
Caltrain	10	0
Tempete	10	0
Miss America	10	0
Susie	10	0
Tennis	10	0
Trevor	10	0
Garden	10	0
Salesman	10	0
Football	9	1

Table 4.41: Overall perceptual quality visual inspection results of 2AFC between 15 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving better quality for GM**	**Number of viewers perceiving better quality for FIOD**
Caltrain	10	0
Tempete	10	0
Miss America	10	0
Susie	10	0
Tennis	10	0
Trevor	10	0
Garden	10	0
Salesman	10	0
Football	10	0

Table 4.42: Overall perceptual quality visual inspection results of 2AFC between 25 fps halftone videos generated using GM and FIOD methods.

Sequence	**Number of viewers perceiving better quality for GM**	**Number of viewers perceiving better quality for FIOD**
Pedestrian-area	10	0
Rush-hour	10	0
Sunflower	10	0
Shields	9	1
Blue-sky	9	1
Station	10	0
Tractor	9	1

4.6 Summary

This chapter discusses the power issues pertaining to portable multimedia devices. Conventional LCD display systems that utilize backlighting mechanism are very power inefficient. Bistable display technology offers an attractive and power efficient alternative. Reflective bistable display devices utilize ambient light for their operation, thus, there is no need for a backlight when ambient lighting conditions are sufficient for the reflective mechanism to work. Furthermore, bistable display devices do not require power to sustain a static image on the display. This chapter develops a simple model to evaluate power consumption of bistable display component of a multimedia handheld device. Five different video halftoning algorithms are compared using this model. A correlation between flicker and power consumption is observed. Of the five halftoning algorithms evaluated, FIFSED algorithm produces halftones that have the highest flicker and, also, the highest power requirement. This chapter also presents a preliminary approach that could be taken to develop an overall quality assessment measure for medium frame rate binary halftone videos.

Chapter 5

Video Halftone Enhancement via Reduction of Flicker under a Spatial Quality Constraint

Chapter 2 discussed two key temporal artifacts of medium frame rate binary video halftones. Chapter 2 also developed a generalized framework for the evaluation of these key temporal artifacts. Chapter 3 utilized the temporal artifact assessment framework of Chapter 2 to design algorithms to *generate* halftone videos such that the temporal artifacts were reduced. Chapter 4 presented a power analysis applicable to bistable display devices. In Chapter 4, the power performance of halftone videos generated using five video halftoning algorithms was compared and a statistical relationship was developed between power consumption (of the bistable display component) and the key temporal artifacts. After presenting the above stated contributions, a problem natural to address now is that of video halftone *enhancement*. In solving the problem of video halftone enhancement, a halftone video that has *already* been generated using a video halftone *generation* algorithm is enhanced by reducing artifacts. In Chapters 5, 6, and 7, I present several different solutions to the problem of video halftone enhancement.

As opposed to the video halftone *generation* problem addressed in Chapter 3, the focus of this chapter is on the problem of constrained *enhancement* of

video halftones. Consequently, the algorithms developed in this chapter are video halftone enhancement algorithms and *not* video halftone generation algorithms. The video halftone enhancement methods described in this chapter do not depend on and, hence, do not utilize the temporal artifact assessment framework developed in Chapter 2.

In this chapter, the term enhancement means enhancement by reducing flicker in a binary halftone video. The problem that I solve in this chapter can be stated as follows. Given a binary halftone video produced from a continuous-tone grayscale video, it is desired to reduce flicker under the constraint that the spatial quality of each frame of the halftone video does not deteriorate as a result of flicker reduction. The more general constraint is that the flicker reduction be carried out such that, in the process of flicker reduction, any deterioration in spatial quality does not exceed a certain limit. This limit is quantified by means of using a threshold value that quantitatively represents the "amount" of "perceptually tolerable" additional deterioration in the spatial quality of the frames of the halftone video. Constraining the spatial quality of individual frames ensures that the perceptual quality of each frame of the halftone sequence is acceptable when a frame is viewed as an image. Reduction of flicker, under the constraint of preserving spatial quality, gives the benefit of improved perceptual quality when the frames are viewed in a sequence (i.e. as a video).

This chapter begins by introducing some supplemental notation necessary for understanding the development of video halftone enhancement algorithms. Human visual system modeling is discussed next. Then, a new video halftone enhancement

algorithm is developed. The *theoretical* bound on the perceptual error (or degradation) introduced, in each frame of the halftone video, by the algorithm is discussed. The computational disadvantages of the developed algorithm are discussed. This discussion is followed by the development of a computationally efficient video halftone enhancement algorithm. Theoretical bounds on the perceptual error introduced by the computationally efficient algorithm are discussed. Next, results of enhancing medium frame rate binary halftone videos are presented and discussed. Finally, the chapter concludes with a summary of the contributions detailed in this chapter. Part of the work presented in this chapter has been submitted to be considered for publication [128].

5.1 Notation

Below, I introduce some supplemental notation relevant to the discussion in this chapter. For clarity some notation already introduced in Section 2.4.1 is included as well.

- C_i: the i^{th} frame of the continuous-tone (original) video, V_c;
- $C_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the continuous-tone frame C_i;
- $\widetilde{C}_i$: the i^{th} perceived (by a human viewer) frame of the continuous-tone video, V_c;
- $\widetilde{C}_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the perceived continuous-tone frame $\widetilde{C}_i$;

- D_i: the i^{th} frame of the halftoned video, V_d;
- $D_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the halftone frame D_i;
- $\widetilde{D}_i$: the i^{th} perceived (by a human viewer) frame of the halftone video, V_d;
- $\widetilde{D}_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the perceived halftone frame $\widetilde{D}_i$;
- DE_i: the i^{th} frame of the enhanced halftone video, V_{de};
- $DE_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the enhanced halftone video DE_i;
- $\widetilde{DE_i}$: the i^{th} perceived (by a human viewer) frame of the enhanced halftone video, V_{de};
- $\widetilde{DE_i}\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the perceived enhanced halftone frame $\widetilde{DE_i}$;
- V_c: the continuous-tone (contone) video;
- V_d: the corresponding halftone video;
- V_{de}: the enhanced halftone video produced by reducing artifacts in the halftone video, V_d.

Also, recall from Chapter 2 that I represents the total number of frames in V_c, M represents the total number of pixel rows in each frame of V_c, and N

represents the total number of pixel columns in each frame of V_c. This means that $1 \leq i \leq I$, $1 \leq m \leq M$, and $1 \leq n \leq N$. For ease of description, I further express the continuous-tone, the halftone, and the enhanced halftone videos as the following (time) sequences:

$$V_c = \{C_i\}_{i=1}^{I}, \tag{5.1}$$

$$V_d = \{D_i\}_{i=1}^{I}, \tag{5.2}$$

$$V_{de} = \{DE_i\}_{i=1}^{I}. \tag{5.3}$$

With the supplemental notation introduced in this section, I am now ready to present a technique that can be used to enhance medium frame rate halftone videos.

5.2 Human Visual System Model and Preliminaries

The video halftone enhancement algorithms that I describe in this chapter rely on a model of the HVS. I model the spatial properties of the HVS by a two-dimensional linear shift-invariant filter [28] with low-pass characteristics [38]. This type of HVS modeling has been discussed in Section 1.2.1.1. Nasanen's model is used to represent the HVS [38]. As discussed in Section 1.2.1.1, the frequency response, $H_r(f_r)$, for Nasanen's model is given by

$$H_r(f_r) = aL^b e^{-(f_r/[c\ln(L)+d])}, \tag{5.4}$$

where f_r is the radial spatial frequency, L is average luminance, and a, b, c, and d are constants. The unit of spatial frequency is cycles per degree.

Let p_{HVS} represent the point spread function of the HVS. Then, ignoring the effects of the display device, the perceived i^{th} continuous-tone frame is given by

$$\widetilde{C}_i = C_i \circledast p_{HVS}, \tag{5.5}$$

where $\circledast$ represents two-dimensional convolution.

Similarly, the perceived i^{th} halftone frame is given by

$$\widetilde{D}_i = D_i \circledast p_{HVS}, \tag{5.6}$$

The perceived i^{th} enhanced halftone frame is given by

$$\widetilde{DE}_i = DE_i \circledast p_{HVS}, \tag{5.7}$$

The i^{th} error frame is defined as the difference of the i^{th} continuous-tone and halftone frames. Let $E_{i,d,c}$ be the i^{th} error frame corresponding to the i^{th} halftone frame D_i. Each pixel of $E_{i,d,c}$, $E_{i,d,c}(m,n)$ is given by

$$E_{i,d,c}(m,n) = C_i(m,n) - D_i(m,n). \tag{5.8}$$

The associated perceived i^{th} error frame corresponding to the i^{th} halftone frame D_i is

$$\widetilde{E}_{i,d,c} = E_{i,d,c} \circledast p_{HVS}. \tag{5.9}$$

The perceived total squared error of D_i (with respect to C_i), $\widetilde{E}_{i,d,c,total}$, is defined as

$$\widetilde{E}_{i,d,c,total} = \sum_m \sum_n \left| \widetilde{E}_{i,d,c}\,(m,n) \right|^2. \tag{5.10}$$

This is the general form of the (frame) error metric that will be used to constrain the perceptual degradation that the i^{th} halftone frame might get as a result of reducing flicker in the halftone video. Note from (5.9) that the value of this error metric is dependent on the HVS model. Thus, different HVS models, or even different implementations of the same HVS model, will predict different values of the perceptual error.

5.3 Halftone Video Enhancement Algorithm

The algorithm that I propose in this section enhances the perceptual quality of a halftone video by reducing flicker, which is a temporal artifact. The algorithm requires no knowledge of the halftoning method used to generate the input halftone video V_d. To generate the output halftone video, V_{de}, the algorithm requires the halftone video V_d, and the corresponding continuous-tone video V_c. These are the only two video data inputs to the halftone video enhancement algorithm.

In the process of halftone video enhancement (i.e. flicker reduction), some pixels of the input halftone video V_d must, in general, change to produce the (final) output halftone video V_{de}. For a precise understanding of the algorithm proposed

in this section, it is important to distinguish between the *initial* V_{de} and the *final* V_{de}. The output of the algorithm is the *final* V_{de}. The algorithm begins by setting the *initial* V_{de} to have the same pixel values as the input halftone video, V_d. The *initial* V_{de} is then changed, during the process of enhancement, to produce its *final* version that is output as the enhanced halftone video.

Figure 5.1 shows the general concepts involved in enhancing a binary halftone video using the proposed algorithm. As pointed out in the previous paragraph, initially the i^{th} frame of V_{de}, DE_i has the same pixel values as the corresponding input halftone frame, D_i. The pixels of DE_i then get changed in the course of enhancement. I define ψ_i to be the *ordered* set of pixels that change, as a result of enhancement, in the halftone frame DE_i. The order in which elements appear in this set indicates the order in which the pixels get changed. Let U_i be the total number of pixels that get changed in the *initial* DE_i to produce the *final* DE_i. Let k index the elements of ψ_i. Let the k^{th} pixel in the ordered set ψ_i be denoted by $u_{i,k}$. I also let ζ_i represent the *ordered* set of pixel locations corresponding to the pixels in the set ψ_i. Note that the order of the elements of the set ζ_i depends on the order of the elements of the set ψ_i. The preceding discussion suggests

$$U_i \leq M \cdot N, \tag{5.11}$$

$$\psi_i = \{u_{i,k} : 1 \leq k \leq U_i\}, \tag{5.12}$$

$$|\psi_i| = U_i, \tag{5.13}$$

$$\zeta_i = \{x_{i,k} : DE_i(x_{i,k}) \in \psi_i\}, \tag{5.14}$$

and

$$|\zeta_i| = |\psi_i|. \tag{5.15}$$

Note that k indexes the elements of ζ_i as well, and that the k^{th} element of ζ_i is denoted by $x_{i,k}$.

In Figure 5.1, the algorithm begins by setting the output enhanced halftone video V_{de} to be the same as the input halftone video V_d. Then, the output halftone video V_{de} is modified frame-by-frame starting from the first frame DE_2 and sequentially processing the rest. Since the algorithm seeks to enhance the halftone video by reducing flicker, which is a temporal artifact, it starts from the second frame. The first output frame, DE_1, remains unchanged. For $i > 1$, to generate the i^{th} output frame DE_i, the output frame DE_i is traversed pixel-by-pixel at only those pixel locations where DE_i and DE_{i-1} differ. Since these pixels are toggling values between the successive frames DE_i and DE_{i-1}, these are the pixels whose values could *potentially* be the cause of any perceived flicker. Let ξ_i be the ordered set of pixel locations that have different values between the two adjacent frames DE_i and DE_{i-1}. The elements of ξ_i are indexed by j. Let us denote the j^{th} element of ξ_i by $w_{i,j}$. Then, $w_{i,j}$ represents a pixel location vector. In other words, if (m, n) is the spatial location whose value toggles between the adjacent frames DE_i and DE_{i-1},

then $w_{i,j} = (m, n)$ for some value of j. In this case, we could also write $DE_i(m, n)$ more succinctly as $DE_i(w_{i,j})$. Therefore,

$$\xi_i = \{w_{i,j} : DE_i(w_{i,j}) \neq DE_{i-1}(w_{i,j})\}. \tag{5.16}$$

The element order of ξ_i will depend on how the pixels are traversed during a particular scan of a frame. The scan order could be raster for example. While processing the i^{th} enhanced halftone frame, at the start of the first scan, the pixel $DE_i(w_{i,1})$ (of frame DE_i) is replaced by the pixel $DE_{i-1}(w_{i-1,1})$ from the previous enhanced frame, DE_{i-1}. I will call this change a trial change. If the trial change causes the difference in the perceptual error between the enhanced halftone frame DE_i, and the continuous-tone frame C_i, and the perceptual error between the original halftone frame D_i and the continuous-tone frame C_i to be lower than (or equal to) a certain threshold T_0, discussed later in this section, then the change in pixel value is accepted. Otherwise, the change is discarded. This process is repeated at each pixel location in the set ξ_i until all the pixel locations in the set ξ_i have been processed. This completes the first full scan of the frame. At the end of a full scan, the elements of ξ_i that represent locations of pixels that were changed during the scan are removed from the set ξ_i.

After a full scan of the frame DE_i, the possibility of another scan of the same frame, DE_i, is determined. This is done by checking if a convergence criterion is met. Convergence criterion could, for example, be based on the number of pixel changes in the (last) completed scan. I discuss the convergence criterion further

later in this chapter. If the convergence criterion is not met, the scan is repeated on the enhanced frame DE_i. In the next scan, if the order of pixel traversal is changed, then the ordering of elements of the set ξ_i is accordingly changed before beginning the scan. On the other hand, if the convergence criterion is satisfied, then the algorithm moves on to enhance the next frame DE_{i+1}. This process is continued until all frames have been processed. When this happens, the enhanced video V_{de} is the halftone video with reduced flicker.

Let us make some observations. Refer to Figure 5.1 again, and note that during a scan of the i^{th} frame DE_i, $w_{i,j}$ ($\in \xi_i$) denotes the spatial coordinates of the j^{th} pixel location whose value is under a trial change. Let $DE_i^{w_{i,j}}$ denote the enhanced frame DE_i after the pixel at location $w_{i,j}$ is changed for trial. The associated perceived error frame is given by

$$\widetilde{E}_{i,de,c}^{w_{i,j}} = (C_i - DE_i^{w_{i,j}}) \circledast p_{HVS}. \tag{5.17}$$

The perceived total squared error of $DE_i^{w_{i,j}}$ (with respect to C_i), $\widetilde{E}_{i,de,c,total}^{w_{i,j}}$ is then given by

$$\widetilde{E}_{i,de,c,total}^{w_{i,j}} = \sum_m \sum_n \left| \widetilde{E}_{i,de,c}^{w_{i,j}}(m,n) \right|^2. \tag{5.18}$$

Again, it is important to note that $DE_i^{w_{i,j}}$ represents the most current state/form of DE_i. That is, $DE_i^{w_{i,j}}$ reflects all pixel changes accepted thus far, including those changes accepted in the previous scans of DE_i, as well as the current trial change at pixel location $w_{i,j}$.

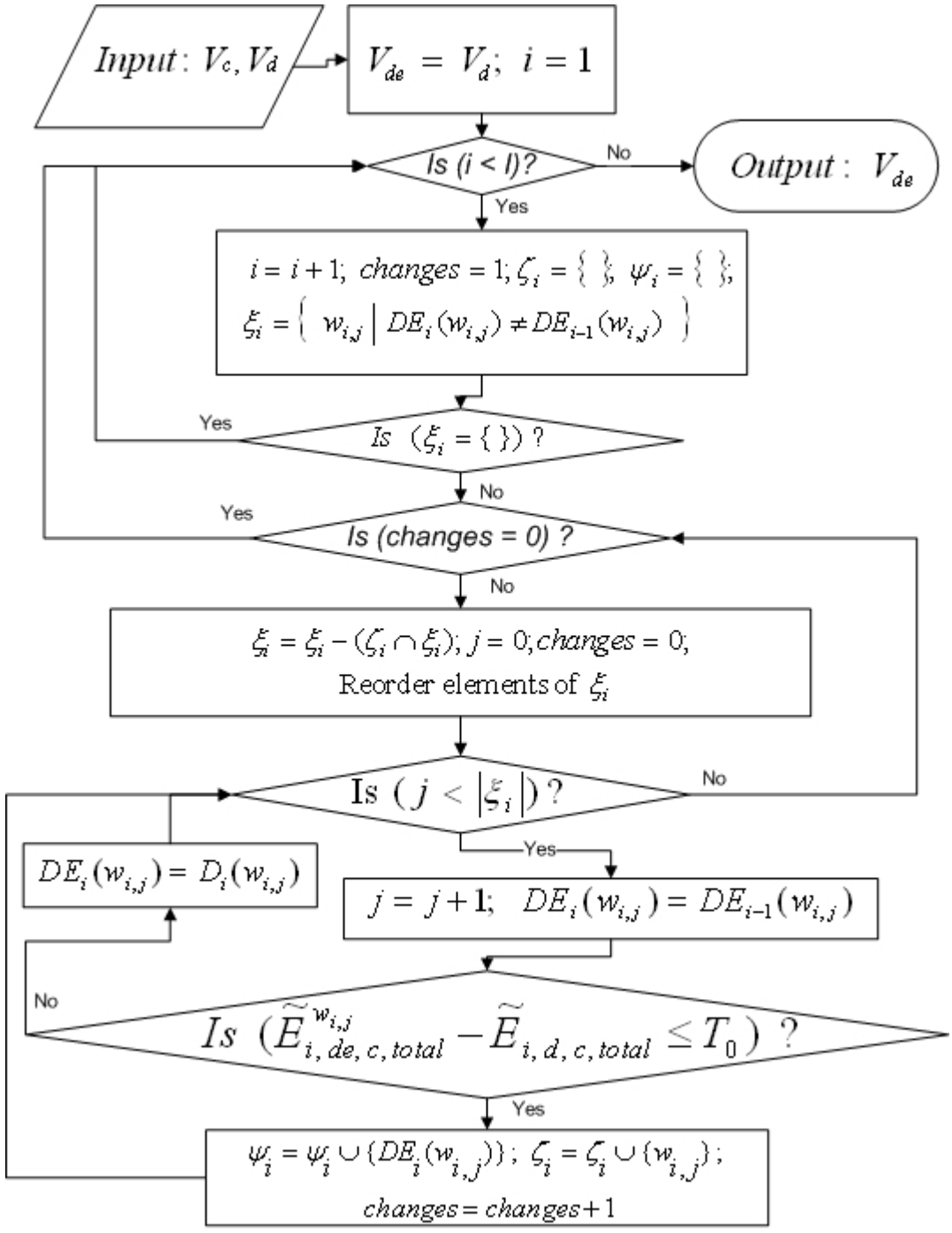


Figure 5.1: Binary halftone video enhancement algorithm. The halftone video is enhanced by reducing flicker.

5.3.1 Convergence Criterion and Determination of Threshold T_0

The flicker reduction method described in this chapter processes frames sequentially. The processing of one frame must be completed before the next successive frame can be processed. Since the algorithm is designed to keep scanning a frame until a certain condition is met, this "certain condition" must be met before proceeding to the next frame in the sequence. I call this "certain condition" the convergence criterion. A number of possibilities exist for establishing the convergence criterion based on the number of accepted pixel value changes that take place in a scan of DE_i, the i^{th} halftone frame under enhancement. One possibility is to keep on repeating the frame scans until the number of accepted pixel value changes during a scan reaches a number, zero, for instance. This is the criterion that is assumed for the discussion in this dissertation.

The threshold value T_0 determines the amount of flicker reduction in the enhanced halftone video V_{de}. It also constrains the spatial artifacts that might get introduced in the frame DE_i as a result of flicker reduction. Generally speaking, a lower value of T_0 will constrain introduction of spatial artifacts, but will not reduce flicker as much. On the other hand, a higher threshold value will reduce flicker more, but could result in perceptual quality degradation of individual frames. For the results reported in this dissertation, T_0 is assigned a value of zero. With T_0 set to zero, based on the error metric used in (5.10) and (5.18), the enhanced (output) frames should *theoretically* have at least as good a perceptual spatial quality as the original (input) halftone frames. In addition flicker will also be reduced. If T_0 is greater than zero, flicker will be alleviated more at the expense of individual frame

quality.

5.3.2 Computational Issues

The flicker reduction algorithm developed in this section is computationally not very efficient. To appreciate this fact, please refer to Figure 5.1 and note that the evaluation of $(\widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total} \leq T_0)$ is done each time a pixel of the frame DE_i has a value different from the same spatial location pixel in the previous frame DE_{i-1} . For the evaluation of this expression, both $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ and $\widetilde{E}_{i,d,c,total}$ need to be computed. Of these two, $\widetilde{E}_{i,d,c,total}$ needs to be computed only once per frame. On the other hand, $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ needs to be computed every time a pixel trial change in made in the (enhanced) frame DE_i. Recall from (5.17) and (5.18), that the evaluation of $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ requires a convolution operation between the error image $(C_i - DE_i^{w_{i,j}})$ and the HVS filter p_{HVS}. The two-dimensional convolution operations are very expensive, particularly when the spatial resolution of DE_i is high. Even if a complete convolution operation is not performed to evaluate the effect of a trial change, the required number of computational operations is still going to be high. The proposed enhancement algorithm is an iterative algorithm, and it is desirable to evaluate the effect of trial changes in a binary halftone frame more efficiently.

5.4 Computationally Efficient Halftone Video Enhancement Algorithm

In this section, I design an algorithm that enhances the perceptual quality of a binary halftone video in a computationally efficient manner. The enhancement

algorithm of this section attempts to relatively efficiently do the job of the enhancement algorithm of the previous section. The algorithm reduces flicker to improve the perceptual quality of the halftone video.

Let $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ represent the *change* in perceptual error due to trial-changing the pixel at location $w_{i,j}$ in the frame DE_i. Also, let $\widetilde{E}^{\delta}_{i,de,c,total}$ be the perceptual error between DE_i and C_i prior to making the trial change at pixel location $w_{i,j}$. Then, $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ is given by

$$\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} = \begin{cases} \widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}^{\delta}_{i,de,c,total} & \text{for } 2 \leq j \leq M \cdot N, \\ \widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total} & \text{for } j = 1. \end{cases} \tag{5.19}$$

To reduce computational complexity, an efficient method to evaluate the effect of trial pixel changes has been developed in [28, 63, 64]. To reduce computational complexity of my video halftone enhancement algorithm, I will use this computationally efficient method [28, 63, 64] to calculate $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$. Let us assume that $w_{i,1}$ is the pixel location whose value is under a trial change. This is the first pixel location in the i^{th} frame DE_i that undergoes a trial change. Now take a closer look at how the derivation of [28] applies to evaluating $\Delta\widetilde{E}^{w_{i,1}}_{i,de,c,total}$. For the efficient evaluation of $\Delta\widetilde{E}^{w_{i,1}}_{i,de,c,total}$, correlation matrices $c_{p_{HVS}p_{HVS}}$ and $c_{p_{HVS}\widetilde{E}_{i,de,c}}$ are needed. These correlation matrices are, respectively, given by

$$c_{p_{HVS}p_{HVS}} = p_{HVS} \otimes p_{HVS}, \tag{5.20}$$

and

$$c_{p_{HVS}\widetilde{E}_{i,de,c}} = p_{HVS} \otimes \widetilde{E}_{i,de,c}, \tag{5.21}$$

where $\otimes$ represents the two-dimensional correlation operation. I define $a_i(w_{i,j})$ to be

$$a_i(w_{i,j}) = DE_{i-1}(w_{i,j}) - DE_i(w_{i,j}). \tag{5.22}$$

Since the halftone videos discussed have binary pixel values and $w_{i,j} \in \xi_i$, $a_i(w_{i,j})$ is either 1 or -1. Based on the derivation explained in detail in [28], $\Delta\widetilde{E}^{w_{i,1}}_{i,de,c,total}$ can be expressed in terms of $a_i(w_{i,1})$, $c_{p_{HVS}p_{HVS}}$ and $c_{p_{HVS}\widetilde{E}_{i,de,c}}$ as

$$\Delta\widetilde{E}^{w_{i,1}}_{i,de,c,total} = a_i^2(w_{i,1})c_{p_{HVS}p_{HVS}}(0) - 2a_i(w_{i,1})c_{p_{HVS}\widetilde{E}_{i,de,c}}(w_{i,1}). \tag{5.23}$$

The above expression is for $j = 1$. The calculation of $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ for $j > 1$ is performed in a similar fashion.

Note that the correlation matrix $c_{p_{HVS}p_{HVS}}$ stays the same for the entire video! Thus, only one evaluation of $c_{p_{HVS}p_{HVS}}$ is needed. The cross-correlation matrix $c_{p_{HVS}\widetilde{E}_{i,de,c}}$ does not stay constant and must change every time a *trial* change in DE_i is *accepted*. The initial matrix $c_{p_{HVS}\widetilde{E}_{i,de,c}}$ is calculated by a correlation operation (once per frame) and, thereafter, only needs an updating every time a *trial* change in the enhanced frame DE_i is *accepted*. This updating operation has also been derived [28], and, if $w_{i,j}$ is the pixel location in DE_i where a trial change is accepted, it is given by

$$c_{p_{HVS}\widetilde{E}_{i,de,c}}(l) = c_{p_{HVS}\widetilde{E}_{i,de,c}}(l) - a_i(w_{i,j})c_{p_{HVS}p_{HVS}}(l - w_{i,j}), \tag{5.24}$$

where $l = (m, n)^T$ denotes a pixel location in $c_{p_{HVS}\widetilde{E}_{i,de,c}}$.

Note that in the evaluation of $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$, only scalar arithmetic is needed! Figure 5.2 depicts the algorithm that utilizes this computational efficiency to evaluate the effect of a trial change in the value of a pixel. The algorithm evaluates the effect of a trial change via a calculation of $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$, rather than via evaluating $(\widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total})$. As depicted in (5.23), calculation of $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ requires only a few scalar multiplications and a subtraction operation. Whereas, as discussed in Section 5.3.2, calculating $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ for use in $(\widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total})$, to evaluate the effect of a trial change, was relatively more expensive in a computational sense.

Note that calculating $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ does not help us evaluate $(\widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total} \leq T_0)$, however! Accepting a trial change in a pixel's value based on $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} \leq T_0$ will likely result in improving the (spatial) perceptual quality of the i^{th} frame DE_i, if $T_0 \leq 0$. This, however, will not reduce flicker as effectively. Remember though, the primary goal of the algorithm, currently under design, is effective flicker reduction while preserving spatial quality of frames. To meet this goal, I replace the constant threshold T_0 by an adaptive threshold $T_{w_{i,j}}$. $T_{w_{i,j}}$ represents the threshold used to evaluate the effect of trial changing the pixel at location $w_{i,j}$ in DE_i. With threshold modulation in place, I can now reduce flicker effectively by evaluating the effect of trial pixel changes via $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} \leq T_{w_{i,j}}$. The trial change at pixel location $w_{i,j}$ is accepted only if $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} \leq T_{w_{i,j}}$ is true. The threshold

$T_{w_{i,j}}$ needs to be changed every time a trial pixel change is accepted. It is modulated according to (5.25) and (5.26), which are given as

$$T_{w_{i,1}} = T_0, \tag{5.25}$$

where T_0 is an initial value for the threshold, and

$$T_{w_{i,j+1}} = \begin{cases} T_{w_{i,j}} - \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} & \text{if the trial change at location } w_{i,j} \text{ is accepted,} \\ T_{w_{i,j}} & \text{if the trial change at location } w_{i,j} \text{ is rejected.} \end{cases} \tag{5.26}$$

Figure 5.2 is a detailed formal flowchart representation of the efficient flicker reduction algorithm explained in this section. Now, I explain the impact of the initial value of the threshold $T_{w_{i,j}}$ on the temporal and the spatial perceptual quality of the enhanced video V_{de}.

5.4.1 The Initial Threshold, T_0

The initial value of threshold $T_{w_{i,j}}$, T_0, provides a trade-off between the degree by which flicker is reduced and the amount of any *additional* degradation introduced in the spatial quality of the halftone frame DE_i as a result of reducing flicker using this algorithm. A higher value of T_0 will reduce flicker more at the expense of potentially degrading the spatial quality of DE_i with respect to the spatial quality of D_i.

5.4.2 Theoretical Error Bound

From Figure 5.1, it is clear that after the enhancement algorithm of Section 5.3 has processed the i^{th} frame DE_i of the enhanced halftone video V_{de}, the (spatial)

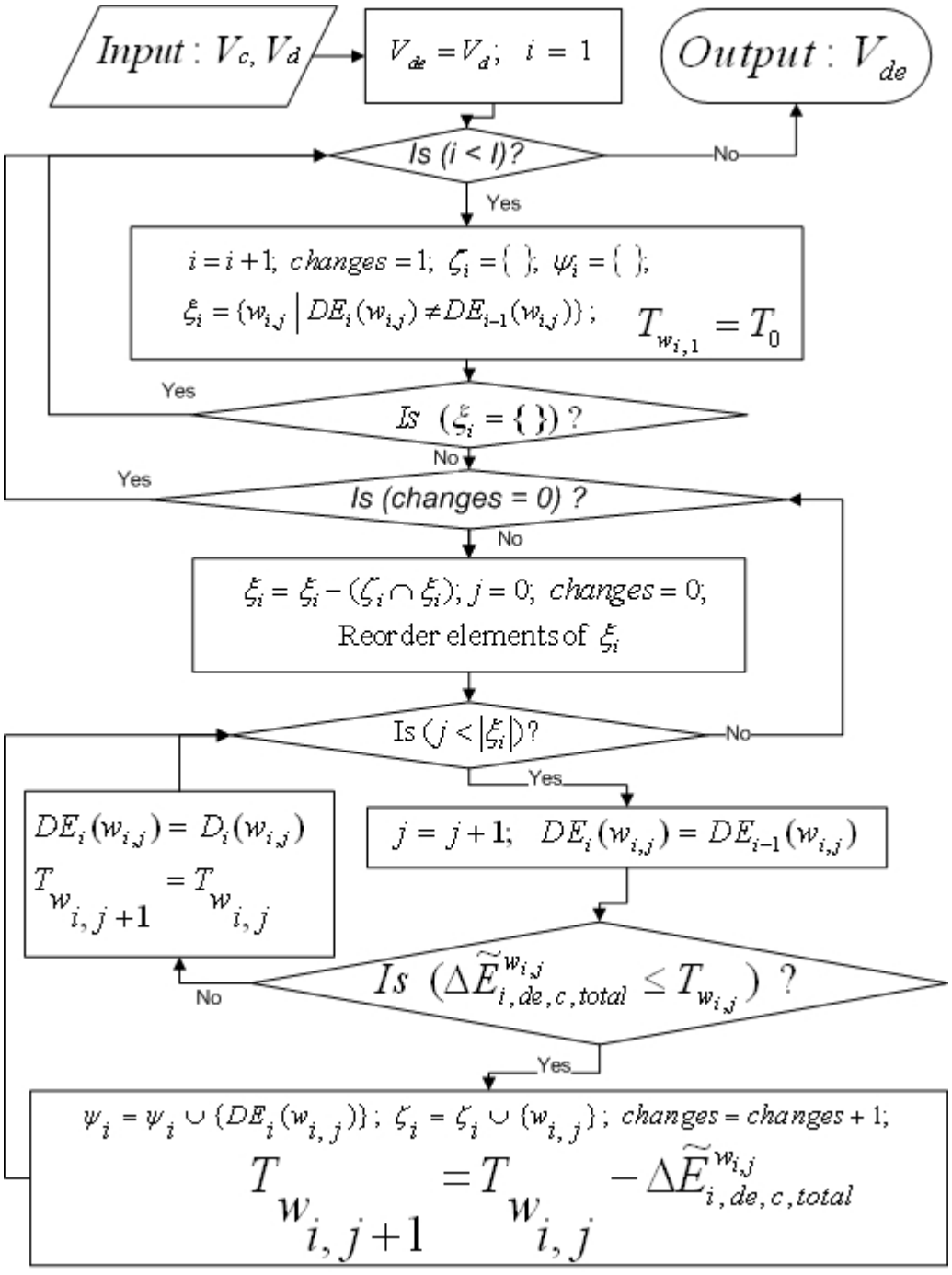


Figure 5.2: An efficient algorithm to reduce flicker in binary video halftones.

perceptual error in the frame DE_i has increased by at most T_0 as compared to the (spatial) perceptual error of the (input) frame D_i. However, it is not readily clear as to how much of spatial perceptual error gets introduced in a frame DE_i as a result of applying the efficient flicker reduction algorithm of this section. To present a clearer picture of what is going on, below, I derive an error bound on the perceptual error introduced in the i^{th} frame DE_i as a result of applying flicker reduction algorithm (on V_d) of this section.

Referring back to Section 5.3, note that for the (enhanced) halftone video V_{de}, after the i^{th} frame DE_i has been enhanced, the total number of pixels that changed in DE_i is U_i. ψ_i is the ordered set of pixels that changed in DE_i and the k^{th} pixel in the ordered set ψ_i is denoted by $u_{i,k}$. Let $\widetilde{E}^{u_{i,k}}_{i,de,c,total}$ represent the total perceptual error between C_i and DE_i after the pixel $u_{i,k}$ ($\in U_i$) gets changed (in DE_i). Thus, I can write

$$\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total} = \begin{cases} \widetilde{E}^{u_{i,k}}_{i,de,c,total} - \widetilde{E}^{u_{i,k-1}}_{i,de,c,total} & \text{for } 2 \leq k \leq U_i, \\ \widetilde{E}^{u_{i,k}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total} & \text{for } k = 1. \end{cases} \tag{5.27}$$

Let $T_{u_{i,k}}$ represent the threshold that had been used to evaluate the effect of trial changing the pixel $u_{i,k}$. Then, based on (5.25) and (5.26), (5.28) and (5.29) are given as

$$T_{u_{i,1}} = T_0, \tag{5.28}$$

where T_0 is an initial value for the threshold, and

$$T_{u_{i,k+1}} = T_{u_{i,k}} - \Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total} \text{ for } 1 \leq k < U_i. \tag{5.29}$$

Notice the relationship between $\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$ and $T_{u_{i,k}}$. $\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$ is the change in perceptual error between the i^{th} enhanced halftone and continuous-tone frames (DE_i and C_i, respectively) due to a change of value of pixel $u_{i,k}$. $T_{u_{i,k}}$ is the threshold that was used to allow the change of value of pixel $u_{i,k}$. Since this change was allowed, based on the algorithm described in Figure 5.2,

$$\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total} \leq T_{u_{i,k}}. \tag{5.30}$$

The three corresponding sequences are

$$\left\{\widetilde{E}^{u_{i,k}}_{i,de,c,total}\right\}_{k=1}^{U_i}, \tag{5.31}$$

$$\left\{\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}\right\}_{k=1}^{U_i}, \tag{5.32}$$

and

$$\left\{T_{u_{i,k}}\right\}_{k=1}^{U_i+1}. \tag{5.33}$$

With these explained, I am ready to introduce Lemma 1, and Theorem 1.

Lemma 1. $T_{u_{i,k+1}} \geq 0$ *for* $1 \leq k \leq U_i$

5.4.2.1 Proof of Lemma 1

Base Step: $k = 1$

$$T_{u_{i,2}} = T_{u_{i,1}} - \Delta\widetilde{E}^{u_{i,1}}_{i,de,c,total} \text{ (from (5.29))}$$

$\Rightarrow$ $T_{u_{i,2}} \geq 0$, since $\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total} \leq T_{u_{i,k}}$ (from (5.30))

Inductive Hypothesis: $k = r - 1$

Assume $T_{u_{i,r}} \geq 0$

Inductive Step: $k = r$

$$T_{u_{i,r+1}} = T_{u_{i,r}} - \Delta\widetilde{E}^{u_{i,r}}_{i,de,c,total} \text{ (from (5.29))}$$

$\Rightarrow$ $T_{u_{i,r+1}} \geq 0$
since $\Delta\widetilde{E}^{u_{i,r}}_{i,de,c,total} \leq T_{u_{i,r}}$ from (5.30),
and $T_{u_{i,r}} \geq 0$ based on the inductive hypothesis. $\square$

Theorem 1. $\widetilde{E}^{u_{i,U_i}}_{i,de,c,total} \leq \widetilde{E}_{i,d,c,total} + T_0$ *for* $U_i \geq 1$

5.4.2.2 Proof of Theorem 1

From (5.27)

$\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total} = \widetilde{E}^{u_{i,k}}_{i,de,c,total} - \widetilde{E}^{u_{i,k-1}}_{i,de,c,total}$ for $2 \leq k \leq U_i$

$$\Rightarrow \quad \widetilde{E}^{u_{i,k}}_{i,de,c,total} - \widetilde{E}^{u_{i,k-1}}_{i,de,c,total} = \Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$$

$$\Rightarrow \quad \sum_{k=2}^{U_i}(\widetilde{E}^{u_{i,k}}_{i,de,c,total} - \widetilde{E}^{u_{i,k-1}}_{i,de,c,total}) = \sum_{k=2}^{U_i}\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$$

$$\Rightarrow \quad (\widetilde{E}^{u_{i,U_i}}_{i,de,c,total} - \widetilde{E}^{u_{i,U_i-1}}_{i,de,c,total}) + (\widetilde{E}^{u_{i,U_i-1}}_{i,de,c,total} - \widetilde{E}^{u_{i,U_i-2}}_{i,de,c,total}) + ...$$
$$+(\widetilde{E}^{u_{i,2}}_{i,de,c,total} - \widetilde{E}^{u_{i,1}}_{i,de,c,total}) = \sum_{k=2}^{U_i}\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$$

$$\Rightarrow \quad (\widetilde{E}^{u_{i,U_i}}_{i,de,c,total} - \widetilde{E}^{u_{i,1}}_{i,de,c,total}) = \sum_{k=2}^{U_i}\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} = \widetilde{E}^{u_{i,1}}_{i,de,c,total} + \sum_{k=2}^{U_i}\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} = \widetilde{E}_{i,d,c,total} + \Delta\widetilde{E}^{u_{i,1}}_{i,de,c,total} + \sum_{k=2}^{U_i}\Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$$

Since from (5.27), $\Delta\widetilde{E}^{u_{i,1}}_{i,de,c,total} = \widetilde{E}^{u_{i,1}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total}$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} = \widetilde{E}_{i,d,c,total} + (T_{u_{i,1}} - T_{u_{i,2}}) + \sum_{k=2}^{U_i}(T_{u_{i,k}} - T_{u_{i,k+1}})$$

Since from (5.29), $T_{u_{i,k+1}} = T_{u_{i,k}} - \Delta\widetilde{E}^{u_{i,k}}_{i,de,c,total}$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} = \widetilde{E}_{i,d,c,total} + (T_{u_{i,1}} - T_{u_{i,2}}) + (T_{u_{i,2}} - T_{u_{i,3}}) + ... + (T_{u_{i,U_i}} - T_{u_{i,U_i+1}})$$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} = \widetilde{E}_{i,d,c,total} + T_{u_{i,1}} - T_{u_{i,U_i+1}}$$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} = \widetilde{E}_{i,d,c,total} + T_0 - T_{u_{i,U_i+1}}$$

Since from (5.28), $T_{u_{i,1}} = T_0$

$$\Rightarrow \quad \widetilde{E}^{u_{i,U_i}}_{i,de,c,total} \leq \widetilde{E}_{i,d,c,total} + T_0$$

Since from Lemma 1, $T_{u_{i,U_i+1}} \geq 0 \quad \square$

Once the halftone enhancement algorithm of Figure 5.2 has enhanced the i^{th} frame DE_i, $k = U_i$ because the last changed pixel is u_{i,U_i} ($\in \psi_i$). At this point, the total perceptual error of the enhanced frame DE_i, $\widetilde{E}^{u_{i,U_i}}_{i,de,c,total}$ is, as given by Theorem 1, bounded according to

$$\widetilde{E}^{u_{i,U_i}}_{i,de,c,total} \leq \widetilde{E}_{i,d,c,total} + T_0. \tag{5.34}$$

I therefore conclude that the theoretical error bounds are identical for both the video halftone enhancement algorithms described in this chapter.

5.5 Objective Measure for Evaluating Spatial Quality

This chapter proposes video halftone enhancement algorithms that, in the process of flicker reduction, attempt to preserve spatial quality of the frames of a

halftone video. The flicker performance of the resulting enhanced halftone videos is evaluated using the Flicker Index, F, of Chapter 2. The DWE performance of the resulting enhanced halftone videos is evaluated using the DWE Index, DWE, of Chapter 2. The power performance of the resulting enhanced halftone videos is assessed using the Power Index, P, of Chapter 4. To see how well these algorithms preserve the spatial quality of the frames constituting the enhanced halftone video, an image (i.e. frame) quality assessment measure is needed. In this section, I discuss the measure used to evaluate the spatial quality of the enhanced halftone video.

Recall, in Chapter 2, I used the Structural SIMilarity (SSIM) index *map* proposed by Wang *et al.* [95] to measure local similarity between successive frames of a continuous-tone video. For algorithms of Chapter 2, I, however, modify the SSIM index map to have its values range between 0 and 1 inclusive. The implementation of the SSIM algorithm for Chapter 2 algorithms also assumes symmetric values of image pixels at boundaries while carrying out any filtering operations. The use of SSIM index map is different in Chapter 2.

The traditional use of SSIM index [95] is to assess the quality of a distorted image with reference to the original (undistorted) version of the image. It is a full-reference measure. As its name suggests, the SSIM index attempts to quantify the loss of structural information in the distorted image. To assess spatial quality of halftone videos, I use the original mean SSIM index (MSSIM) [95] without any modifications. It can take values between -1 and 1 inclusive. The MSSIM index [95] is for images. To get a single number for the entire video, I compute average of the MSSIM index for all the frames in the video.

Let $S_i(C_i, D_i)$ be a measure of perceptual quality of the i^{th} halftone frame D_i with respect to the continuous-tone frame C_i. Let $S(V_c, V_d)$ represent the Spatial Quality Index of the halftone video V_d with respect to the continuous-tone video V_c. The Spatial Quality Index $S(V_c, V_d)$ for the halftone video with a total of I frames is given by

$$S(V_c, V_d) = \frac{\sum_i S_i(C_i, D_i)}{I} \text{ for } i > 0 \ . \tag{5.35}$$

In this dissertation, I set $S_i(C_i, D_i)$ equal to MSSIM(C_i,D_i) of [95]. For conciseness of notation in this dissertation, I shall use S to denote $S(V_c, V_d)$.

5.6 Implementation and Results

In this section, using the concepts of the algorithms described in this chapter, I present the results of reducing flicker in medium frame rate binary halftone videos. The implementation used to generate the results for enhanced videos is *based* on the efficient enhancement algorithm of Section 5.4. The implementation utilizes the general concepts of this chapter and is *not* an exact reflection of the flow charts of Figures 5.1 and 5.2. The results presented here are based on one particular implementation/instantiation of the *theoretical* concepts described in Section 5.4. Different implementations of the same algorithm could potentially result in variation of results.

As discussed in Section 5.3.1, there are different possibilities for convergence criterion used to determine whether the processing of a frame was complete. For gen-

erating the enhancement results of this section, convergence criterion was checked differently than it was in the originally proposed methods of this chapter. For the implementation used to generate the halftone video enhancement results, the convergence criterion was checked after *two* full scans of the frame (as opposed to the suggestions of Figures 5.1 and 5.2). Before checking for convergence, the two completed scans comprised of a horizontal raster scan and a vertical raster scan. Also, a trial change at a pixel location $w_{i,j}$ was accepted when $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} < T_{w_{i,j}}$ was true. For the initial threshold, a value of $T_0 = 0$ was used. Note, however, that the error bounds discussed earlier in this chapter are *theoretical*. Since any implementation is also constrained by practical limitations (such as those sometimes encountered in handling the pixels at the boundaries of a frame), actual value of *introduced* perceptual error might be different than the theoretical prediction. Furthermore, recall from (5.9) and (5.10) that the error metric, used to constrain the degradation of spatial quality of a frame, is dependent on the HVS filter implementation. Any filter used to represent the HVS is typically tuned to a particular application [20]. The tuning might require modification of the filter parameters to suit the needs of the display designer. It is up to the designer to choose a filter that produces the "best" results for his or her application. Here, I have used an HVS filter *based* on Nasanen's model [38] already discussed in Section 5.2. The filter was tuned for better performance on my LCD screen. For the design of my filter, the parameter values (see Section 5.2) are $a = 131.6$, $b = 0.3188$, $c = 0.525$, $d = 3.91$, and $L = 400$. I used a filter support of 5x5 pixels and assumed a screen resolution of 94 pixels per inch, as well as a viewing distance of 18 inches.

In the implementation of the enhancement algorithm used to generate the results presented in this section, before attempting flicker reduction (Section 5.4), some preprocessing was performed on the input halftone and continuous-tone videos. The continuous-tone video, V_c was preprocessed by performing an edge enhancement operation on each of its frames. The first frame of the halftone video, V_d, was improved using DBS algorithm [64].

Since the FIFSED method has produced videos with most flicker (see Chapters 2, 3, and 4), for halftone video enhancement, I chose the videos generated using FIFSED. Tables 5.1, 5.2, and 5.3 compare the performance of the videos generated using the FIFSED algorithm with the enhanced (FIFSED) videos. The flicker performance is evaluated using the Flicker Index, F, of Chapter 2. The DWE performance is evaluated using the DWE Index, DWE, of Chapter 2. The power performance is assessed using the Power Index, P, of Chapter 4. The spatial quality of the halftone videos is evaluated using the spatial quality index, S, discussed in Section 5.5. For F, DWE, and P, a *lower* value indicates better performance. On the other hand, a higher value of S indicates better performance.

From Tables 5.1, 5.2, and 5.3, note the considerable improvement in flicker, as shown by a lower value of Flicker Index, F. There is also some worsening of DWE performance as indicated by an increase in DWE. The value of spatial quality measure S for the original and the enhanced halftone videos is generally close indicating that the spatial quality of the halftone videos is not reduced by much, if at all, using my implementation of the enhancement algorithm. Table 5.3 depicts the results for videos that have relatively higher spatial resolution (See Table 2.3 for

a description of the resolution of these videos). It can be observed from Table 5.3 that for *some* videos, the enhancement resulted in reduction of flicker as well as a slight increase in the spatial quality of the frames!

Table 5.1: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 30 fps FIFSED and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIFSED	0.427	0.333	0.092	0.041
	Enhanced	0.302	0.233	0.111	0.038
Tempete	FIFSED	0.34	0.266	0.042	0.141
	Enhanced	0.165	0.127	0.053	0.13
Miss America	FIFSED	0.3	0.262	0.044	0.015
	Enhanced	0.191	0.166	0.052	0.015
Susie	FIFSED	0.456	0.4	0.043	0.017
	Enhanced	0.294	0.257	0.055	0.016
Tennis	FIFSED	0.436	0.344	0.066	0.096
	Enhanced	0.147	0.107	0.099	0.082
Trevor	FIFSED	0.366	0.31	0.027	0.029
	Enhanced	0.186	0.158	0.036	0.028
Garden	FIFSED	0.408	0.232	0.127	0.19
	Enhanced	0.29	0.166	0.156	0.168
Salesman	FIFSED	0.361	0.319	0.026	0.044
	Enhanced	0.177	0.157	0.033	0.045
Football	FIFSED	0.457	0.329	0.087	0.068
	Enhanced	0.226	0.156	0.119	0.059

Table 5.2: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 15 fps FIFSED and enhanced halftone Videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIFSED	0.429	0.3	0.134	0.041
	Enhanced	0.32	0.221	0.158	0.038
Tempete	FIFSED	0.358	0.254	0.079	0.141
	Enhanced	0.199	0.138	0.099	0.125
Miss America	FIFSED	0.299	0.267	0.036	0.016
	Enhanced	0.196	0.175	0.041	0.015
Susie	FIFSED	0.458	0.385	0.063	0.017
	Enhanced	0.309	0.258	0.079	0.016
Tennis	FIFSED	0.444	0.33	0.08	0.096
	Enhanced	0.173	0.117	0.117	0.081
Trevor	FIFSED	0.367	0.301	0.042	0.029
	Enhanced	0.207	0.168	0.054	0.028
Garden	FIFSED	0.421	0.211	0.16	0.19
	Enhanced	0.325	0.164	0.189	0.169
Salesman	FIFSED	0.357	0.323	0.011	0.044
	Enhanced	0.183	0.166	0.014	0.046
Football	FIFSED	0.468	0.314	0.109	0.067
	Enhanced	0.256	0.166	0.149	0.059

Table 5.3: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 25 fps FIFSED and enhanced halftone Videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Pedestrian-area	FIFSED	0.388	0.323	0.051	0.028
	Enhanced	0.244	0.183	0.062	0.028
Rush-hour	FIFSED	0.383	0.329	0.027	0.024
	Enhanced	0.256	0.222	0.033	0.024
Sunflower	FIFSED	0.339	0.261	0.070	0.036
	Enhanced	0.232	0.178	0.081	0.036
Shields	FIFSED	0.32	0.211	0.152	0.094
	Enhanced	0.238	0.155	0.172	0.088
Blue-sky	FIFSED	0.301	0.191	0.112	0.109
	Enhanced	0.218	0.14	0.128	0.102
Station	FIFSED	0.381	0.302	0.055	0.025
	Enhanced	0.268	0.213	0.065	0.024
Tractor	FIFSED	0.417	0.261	0.127	0.045
	Enhanced	0.313	0.195	0.151	0.045

5.7 Summary

This chapter explores the problem of constrained enhancement of a binary halftone video. Flicker is reduced in medium frame rate binary halftone videos under the constraint that the amount of spatial quality degradation be controlled in the process of reducing flicker. An algorithm was designed that solved this problem. The developed algorithm is shown to be not feasible in a computational sense. An alternative more computationally efficient algorithm is developed to solve the problem of constrained halftone enhancement via flicker reduction.

Chapter 6

Video Halftone Enhancement via Reduction of DWE under a Spatial Quality Constraint

Enhancement of a halftone video by reducing flicker is discussed in Chapter 5. The kind of video halftone enhancement discussed in Chapter 5 is good for halftone videos that suffer from excessive flicker. There can, however, be instances where a halftone video does not suffer from excessive flicker, but still needs enhancement. An example of such a case is a halftone video that suffers from excessive dirty-window-effect (DWE). Recall from our discussion in Chapters 2, 3, and 4 that videos that suffer from excessive DWE usually do not, at the same time, suffer from excessive flicker. For example, in Chapter 2, it was shown (based on both objective and subjective evaluations), that FIOD video halftoning algorithm (using a 32x32 void-and-cluster dither array [54]) produced videos that had high DWE, and a relatively lower flicker as compared to the videos generated using the (flicker prone) FIFSED method.

The goal of this chapter is to develop methods for enhancing halftone videos that suffer from excessive DWE. This, as one might guess, is done at the expense of introducing flicker. Although the discussed methods for reducing DWE are constrained methods, the constraint, however, is not based on flicker. It is based on

the perceptual quality of (individual) frames of the halftone video. Consequently, the reduction of DWE is done while constraining the amount of spatial quality degradation that might get introduced as a result of reducing DWE.

In this chapter, the term enhancement means enhancement by reducing DWE in a binary halftone video. The problem that I solve in this chapter can be stated as follows. Given a medium frame rate binary halftone video produced from a continuous-tone grayscale video, it is desired to reduce DWE under the constraint that, as a result of DWE reduction, the introduction of any *additional* spatial perceptual errors in each enhanced halftone frame does not exceed a certain limit. This "limit" is controlled by a parameter that quantifies the perceptual degradation in the quality of a halftone frame. As will be explained later in this chapter, this parameter is a threshold that quantitatively represents the amount of (perceptually tolerable) *additional* degradation in the spatial quality of frames. Constraining the spatial quality of individual frames of the enhanced video ensures that the perceptual quality of each frame of the enhanced halftone video is acceptable when a frame is viewed as an image. Reduction of DWE, under the constraint of preserving spatial quality, is aimed to improve perceptual quality when the frames are viewed in a sequence (i.e. as a video).

This chapter utilizes the notation already introduced thus far. For clarity, some of it is repeated in this chapter. Any new terms will be defined as they are needed. The understanding of the techniques developed in this chapter requires a background in human visual system modeling. I model the HVS as a linear shift invariant system. HVS modeling has already been discussed in Chapters 1 and

5. I do not repeat the description of the HVS model used here. Instead, I refer the reader to Section 5.2 to refresh the understanding of the HVS model. The HVS model used in the algorithms described in this chapter is based on Nasanen's CSF [38], that exhibits *low-pass* characteristics of HVS.

In the discussion that follows, I first develop an algorithm for the reduction of DWE. I describe why the developed algorithm is not computationally feasible. I then develop a second algorithm that, in a relative sense, is computationally superior to the first algorithm. The development of these two algorithms is followed by a presentation and discussion of results of enhancing medium frame rate binary halftone videos. Finally, the chapter concludes with a summary of the contributions presented in this chapter.

6.1 Preliminaries

For clarity of presentation in this chapter, I repeat *some* notation from the previous chapters below. For any other notation that does not appear in this section, please refer to Sections 2.4.1 and 5.1.

- C_i: the i^{th} frame of the continuous-tone (original) video, V_c;
- $C_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the continuous-tone frame C_i;
- $\widetilde{C}_i$: the i^{th} perceived (by a human viewer) frame of the continuous-tone video, V_c;

- $\widetilde{C}_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the perceived continuous-tone frame $\widetilde{C}_i$;
- D_i: the i^{th} frame of the halftone video, V_d;
- $D_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the halftone frame D_i;
- $\widetilde{D}_i$: the i^{th} perceived (by a human viewer) frame of the halftone video, V_d;
- $\widetilde{D}_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the perceived halftone frame $\widetilde{D}_i$;
- $\widetilde{E}_{i,d,c,total}$: the perceived total squared error of D_i with respect to C_i;
- DE_i: the i^{th} frame of the enhanced halftone video, V_{de};
- $DE_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the enhanced halftone video DE_i;
- $\widetilde{DE}_i$: the i^{th} perceived (by a human viewer) frame of the enhanced halftone video, V_{de};
- $\widetilde{DE}_i\,(m,n)$: the pixel located at the m^{th} row and the n^{th} column of the perceived enhanced halftone frame $\widetilde{DE}_i$;
- $\Delta DE_{i,i-1}$: the absolute difference image for frames DE_i, DE_{i-1};
- $\widetilde{E}_{i,de,c,total}$: the perceived total squared error of DE_i with respect to C_i;

- V_c: the continuous-tone (contone) video;
- V_d: the corresponding halftone video;
- V_{de}: the enhanced halftone video produced by reducing artifacts in the halftone video, V_d;
- ψ_i: the *ordered* set of pixels that change in DE_i as a result of enhancement;
- U_i: the total number of pixels that get changed in DE_i as a result of enhancement;
- ζ_i: the *ordered* set of pixel locations corresponding to the pixels in ψ_i.

Recall from Chapter 2 that I represents the total number of frames in V_c, M represents the total number of pixel rows in each frame of V_c, and N represents the total number of pixel columns in each frame of V_c. Thus, $1 \leq i \leq I$, $1 \leq m \leq M$, and $1 \leq n \leq N$.

6.2 Halftone Video Enhancement

In this section, I develop an algorithm to enhance the perceptual quality of a medium frame rate binary halftone video, V_d, by reducing DWE. The algorithm requires no knowledge of the halftoning method used to generate the (input) halftone video V_d. To generate the (output) halftone video V_{de}, the algorithm requires the halftone video V_d, and the corresponding continuous-tone video V_c. These are the only two video data inputs to the halftone video enhancement algorithm.

Generally speaking, while reducing DWE, some pixels of the input binary halftone video V_d toggle to eventually produce the *final* output halftone video V_{de}. The pixel values of the *initial* V_{de} are set to equal those of the input halftone video, V_d. This *initial* V_{de} is then changed to form the *final* V_{de}, which is the algorithm output. ψ_i denotes the *ordered* set of pixels that change, as a result of enhancement, in the input halftone frame DE_i. The order in which elements appear in this set indicates the order in which the pixels get changed. U_i denotes the total number of pixels that get changed in the *initial* DE_i to produce the *final* DE_i. Let k index the elements of ψ_i. I also let the k^{th} pixel in the ordered set ψ_i be denoted by $u_{i,k}$. ζ_i represents the *ordered* set of pixel locations corresponding to the pixels in the set ψ_i. Note that the order of the elements of the set ζ_i depends on the order of the elements of the set ψ_i, and that k indexes the elements of ζ_i as well. The k^{th} element of ζ_i is denoted by $x_{i,k}$. Therefore,

$$U_i \leq M \cdot N, \tag{6.1}$$

$$\psi_i = \{u_{i,k} : 1 \leq k \leq U_i\}, \tag{6.2}$$

$$|\psi_i| = U_i, \tag{6.3}$$

$$\zeta_i = \{x_{i,k} : DE_i(x_{i,k}) \in \psi_i\}, \tag{6.4}$$

and

$$|\zeta_i| = |\psi_i|\,. \tag{6.5}$$

For $i > 1$, each pixel $\Delta DE_{i,i-1}(m,n)$ of the i^{th} absolute difference image, $\Delta DE_{i,i-1}$ is given by

$$\Delta DE_{i,i-1}(m,n) = |DE_i(m,n) - DE_{i-1}(m,n)|. \tag{6.6}$$

Note that $\Delta DE_{i,i-1}(m,n)$ is binary valued.

Figure 6.1 gives the formal details of the enhancement algorithm used to reduce DWE. In this figure, $\sim DE_i(w_{i,j})$ denotes the toggled value of $DE_i(w_{i,j})$. The algorithm begins by setting the *initial* enhanced halftone video V_{de} to be the same as the input halftone video V_d. Then, to produce the *final* enhanced halftone video V_{de}, the *initial* V_{de} is modified frame-by-frame starting from its second frame, DE_2, and sequentially processing the rest. The first output frame, DE_1, remains unchanged. For $i > 1$, to generate the i^{th} output frame DE_i, the output frame DE_i is traversed pixel-by-pixel at only those pixel locations where DE_i and DE_{i-1} are the same. Since these pixels have the same values between the successive frames DE_i and DE_{i-1}, these are the pixels whose values could *potentially* be the cause of any perceived DWE. Let ξ_i be the ordered set of pixel locations that have the same values between the two adjacent frames DE_i and DE_{i-1}. The elements of ξ_i are indexed by j. Let us denote the j^{th} element of ξ_i by $w_{i,j}$. Then, $w_{i,j}$ represents a pixel location vector. In other words, if (m,n) is the spatial location whose value stays the same between the adjacent frames DE_i and DE_{i-1}, then $w_{i,j} = (m,n)$

for some value of j. $DE_i(m,n)$ could more succinctly be written as $DE_i(w_{i,j})$. Therefore,

$$\xi_i = \{w_{i,j} : DE_i(w_{i,j}) = DE_{i-1}(w_{i,j})\}. \tag{6.7}$$

The element order of ξ_i will depend on how the pixels are traversed during a particular scan of a frame. The scan order could be raster for example. While processing the i^{th} enhanced halftone frame, at the start of the first scan, the binary pixel $DE_i(w_{i,1})$ (of frame DE_i) is toggled (i.e. its value is changed from either a "1" to a "0" or from a "0" to a "1"). I will call this change a trial change. If the trial change causes the difference in the total (squared) perceptual error between the enhanced halftone frame DE_i, and the continuous-tone frame C_i, and the total (squared) perceptual error between the original halftone frame D_i and the continuous-tone frame C_i to be lower than a certain threshold T_0, then the pixel toggle is accepted. Otherwise, the pixel value is changed back to its original value. This process is repeated at each pixel location in the set ξ_i until all the pixel locations in the set ξ_i have been processed. This completes the first full scan of the frame. One full scan of the frame refers to traversing the set of pixel locations once. At the end of a full scan, the elements of ξ_i that represent locations of pixels that were changed during the scan are removed from the set ξ_i.

After a full scan of the frame DE_i, the possibility of another scan of the same frame, DE_i, is determined. This is done by checking if a convergence criterion is met. Convergence criterion could, for example, be based on the number of pixel

changes in the (last) completed scan. If the convergence criterion is not met, the scan is repeated on the enhanced frame DE_i. In the next scan, if the order of pixel traversal is changed, then the ordering of elements of the set ξ_i is accordingly changed before beginning the scan. On the other hand, if the convergence criterion is satisfied, then the algorithm moves on to enhance the next frame DE_{i+1}. This process is continued until all frames have been processed. When this happens, the enhanced video V_{de} is the halftone video with reduced DWE.

To clarify the algorithm of Figure 6.1 further, note that during a scan of the i^{th} frame DE_i, if $w_{i,j}$ ($\in \xi_i$) denotes the spatial coordinates of the pixel whose value is under a trial change, then $DE_i^{w_{i,j}}$ denotes the enhanced frame DE_i after the pixel at location $w_{i,j}$ is toggled for trial. The associated perceived error frame is given by

$$\widetilde{E}_{i,de,c}^{w_{i,j}} = (C_i - DE_i^{w_{i,j}}) \circledast p_{HVS}. \tag{6.8}$$

Recall that $\circledast$ denotes the two-dimensional convolution operation and that p_{HVS} represents the point spread function representing the linear shift invariant model of the HVS. The perceived total squared error of $DE_i^{w_{i,j}}$ (with respect to C_i), $\widetilde{E}_{i,de,c,total}^{w_{i,j}}$ is given by

$$\widetilde{E}_{i,de,c,total}^{w_{i,j}} = \sum_m \sum_n \left| \widetilde{E}_{i,de,c}^{w_{i,j}}(m,n) \right|^2. \tag{6.9}$$

Please note that $DE_i^{w_{i,j}}$ represents the current state of DE_i. That is, $DE_i^{w_{i,j}}$ is DE_i with all accepted pixel changes as well as the current trial change at the pixel location $w_{i,j}$ ($\in \xi_i$).

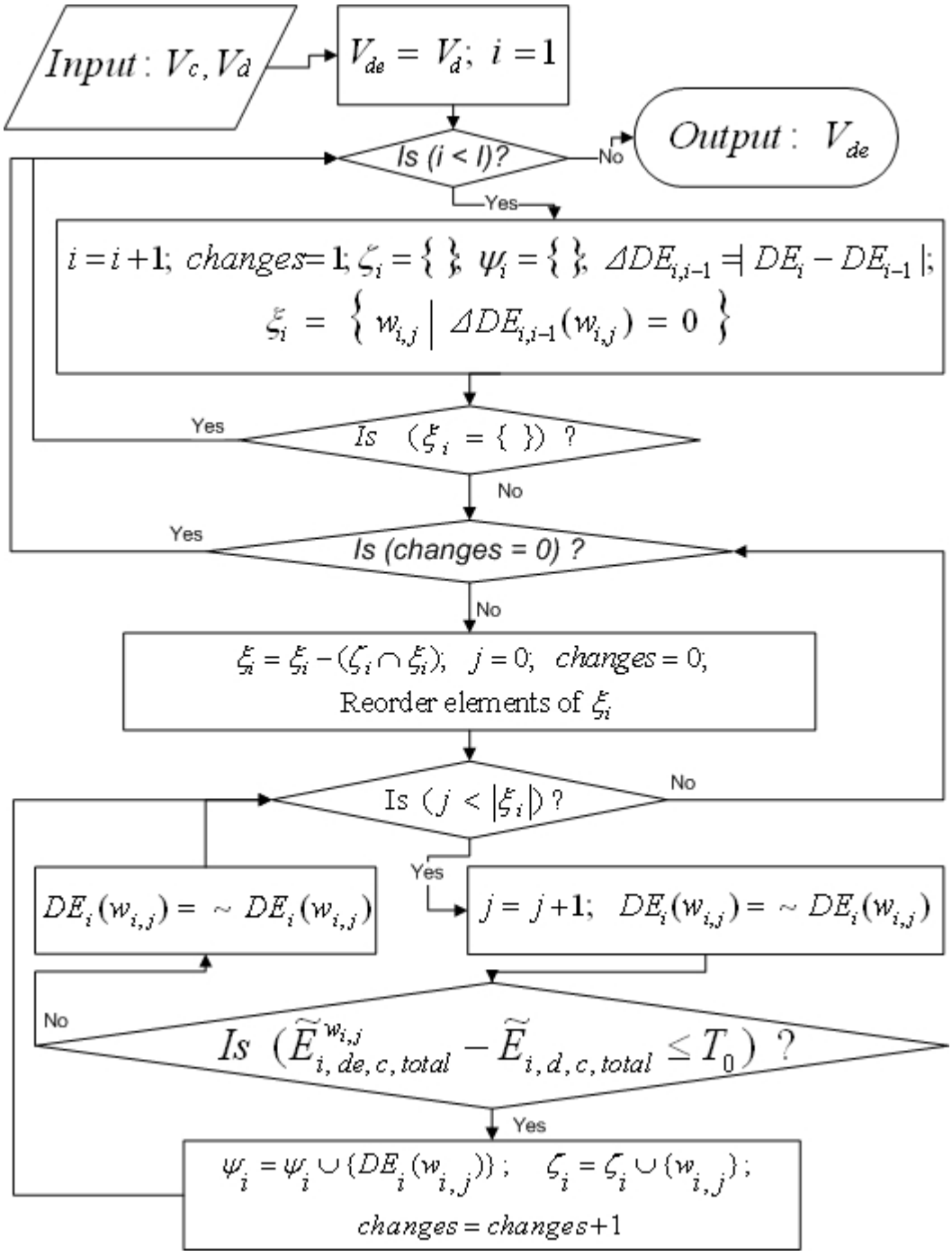


Figure 6.1: Binary halftone video enhancement via DWE reduction.

6.2.1 Impact of Threshold T_0

The threshold value T_0 determines the amount of DWE reduction in the enhanced halftone video V_{de}. It also constrains the spatial artifacts that might get introduced in the frame DE_i as a result of DWE reduction. Generally speaking, a lower value of T_0 will constrain spatial quality degradation more, but will not reduce DWE as effectively. Too low a value of T_0 will not change the input video by much! A higher threshold value will reduce DWE more, but could result in perceptual quality degradation of individual frames. With T_0 set to zero, based on the error metric of (6.9), the enhanced (output) frames should *theoretically* have at least as good a perceptual spatial quality as the original (input) halftone frames. The frames could differ in how the binary pixels are spatially distributed though. Using a higher value of T_0 could alleviate DWE more, but possibly at the expense of individual frame quality.

6.3 Computation Issues

The DWE reduction algorithm developed in this section, like the flicker reduction algorithm of Section 5.3, is not computationally efficient. Refer to Figure 6.1 and note that the evaluation of $(\widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total} \leq T_0)$ is done each time a pixel of the frame DE_i has the same value as the value of the same spatial location pixel in the previous frame DE_{i-1} . For the evaluation of this expression, both $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ and $\widetilde{E}_{i,d,c,total}$ need to be computed. Of these two, $\widetilde{E}_{i,d,c,total}$ needs to be computed only once per frame. On the other hand, $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ needs to be computed every time a pixel trial change in made in the (enhanced) frame DE_i.

Note from (6.8) and (6.9), that the evaluation of $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ requires a convolution operation between the error image $(C_i - DE_i^{w_{i,j}})$ and the HVS filter p_{HVS}. These two-dimensional convolution operations are very expensive, particularly when the spatial resolution of DE_i is high. Even if a complete image convolution is not used to update the value of $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$, its evaluation would still involve several pixels of the error image $(C_i - DE_i^{w_{i,j}})$.

6.4 Computationally Efficient Enhancement of Halftone Videos

Since the algorithm of Section 6.2 is not very efficient, in this section, I modify it to improve its computational performance. The resulting modified algorithm enhances the perceptual quality of a binary halftone video by reducing DWE in a relatively efficient manner. Let $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ represent the *change* in perceptual error due to trial-changing the pixel at location $w_{i,j}$ in the frame DE_i. Also, let $\widetilde{E}^{\delta}_{i,de,c,total}$ be the perceptual error between DE_i and C_i prior to making the trial change at pixel location $w_{i,j}$. Then, $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ is given by

$$\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} = \begin{cases} \widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}^{\delta}_{i,de,c,total} & \text{for } 2 \leq j \leq M \cdot N, \\ \widetilde{E}^{w_{i,j}}_{i,de,c,total} - \widetilde{E}_{i,d,c,total} & \text{for } j = 1. \end{cases} \tag{6.10}$$

Recall from the development in Chapter 5 that the evaluation of $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ is computationally inexpensive compared to the evaluation of $\widetilde{E}^{w_{i,j}}_{i,de,c,total}$. Evaluation of $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$ may be done by utilizing the simpler computations described by (5.22) and (5.23) of Section 5.4. The definition of $a_i(w_{i,j})$ for the algorithm of this section

is different than the definition of (5.22). For the DWE reduction algorithm of this section, I define $a_i(w_{i,j})$ by

$$a_i(w_{i,j}) = NOT(DE_{i-1}(w_{i,j})) - D_i(w_{i,j}). \tag{6.11}$$

The effect of the logical NOT operation is to toggle the (binary) value of $DE_{i-1}(w_{i,j})$.

Figure 6.2 depicts the algorithm that achieves DWE reduction in a computationally efficient manner. In this figure, $\sim DE_i(w_{i,j})$ denotes the toggled value of $DE_i(w_{i,j})$. As the processing of a frame begins, $\widetilde{E}_{i,d,c,total}$ is evaluated and this value is assigned to $\bar{E}_{i,de,c,total}$. To evaluate the effect of a trial change (in the value of the pixel located at $w_{i,j}$ in the enhanced frame DE_i), $(\bar{E}_{i,de,c,total} + \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total})$ is compared against $(\widetilde{E}_{i,d,c,total} + T_0)$. If $(\bar{E}_{i,de,c,total} + \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total})$ is less than or equal to $(\widetilde{E}_{i,d,c,total} + T_0)$, then, relative to the perceptual error of D_i, the trial change does not increase the perceptual error of DE_i by more than T_0. If this is indeed the case, the trial change is accepted. Otherwise, it is rejected. Thus T_0 controls the amount of (spatial) perceptual error introduced in frame DE_i during the process of DWE reduction of the input binary halftone video. If the trial change is accepted, then $\bar{E}_{i,de,c,total}$ is updated using

$$\bar{E}_{i,de,c,total} = \bar{E}_{i,de,c,total} + \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}. \tag{6.12}$$

Note that the *change* (whether positive or negative) in the perceptual error, $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$, due to modifying the value of the pixel located at $w_{i,j}$ in the i^{th} enhanced frame, DE_i, is added to $\bar{E}_{i,de,c,total}$. This update operation causes $\bar{E}_{i,de,c,total}$

to have, at any moment, the most up-to-date value of perceptual error of the enhanced frame DE_i. That is, $\bar{E}_{i,de,c,total}$ reflects the total perceptual error after all pixel changes, in DE_i, up to the current moment have taken place. Thus, $\bar{E}_{i,de,c,total}$ tracks the total perceptual error of the enhanced frame DE_i during the enhancement process. By doing so, for any trial pixel value change, the algorithm is able to make a decision as to whether to accept the trial change or not.

6.4.1 Comparison with Threshold Modulation

Observe that unlike the efficient algorithm of Figure 5.2, the efficient algorithm described by the flowchart of Figure 6.2 does not use threshold modulation to evaluate the effect of a trial change! One might wonder, though, as to how do the two efficient algorithms compare in terms of their relative efficiency.

Of the two efficient video halftone enhancement algorithms described by Figures 5.2 and 6.2, the one described by Figure 5.2 is relatively more efficient. The algorithm of this section, described by Figure 6.2, requires an extra addition operation ($\bar{E}_{i,de,c,total} + \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total}$) to evaluate the effect of a trial change. On the other hand, the algorithm of Figure 5.2 requires no such addition operation to evaluate the effect of a trial change. As has been explained in Chapter 4, computational resources are a commodity on a light, portable handheld device. If the number of pixel value trial changes is significant (which is a possibility), avoiding an addition operation could be desirable. Furthermore, observe from Figure 6.2, that the algorithm of this section requires an initial evaluation of $\widetilde{E}_{i,d,c,total}$ for each enhanced frame DE_i, whereas this evaluation is not needed by the algorithm described in Figure

5.2. Thus, I have established that, in evaluating the effect of a trial change in the value of a pixel, the threshold modulation technique is computationally superior!

Regardless of the relative computational superiority of threshold modulation algorithm of Figure 6.2, the algorithm described in this section is, as explained earlier in this section, far more desirable than the algorithm of Figure 6.1, described in Section 6.2. Furthermore, not only does it provide an additional insight into solving the problem of video halftone enhancement, it also gives an efficient solution as a possible alternative to the use of threshold modulation. Nevertheless, threshold modulation can also be utilized to design an efficient algorithm to reduce DWE, in the same manner as it was utilized to efficiently reduce flicker in Section 5.4.

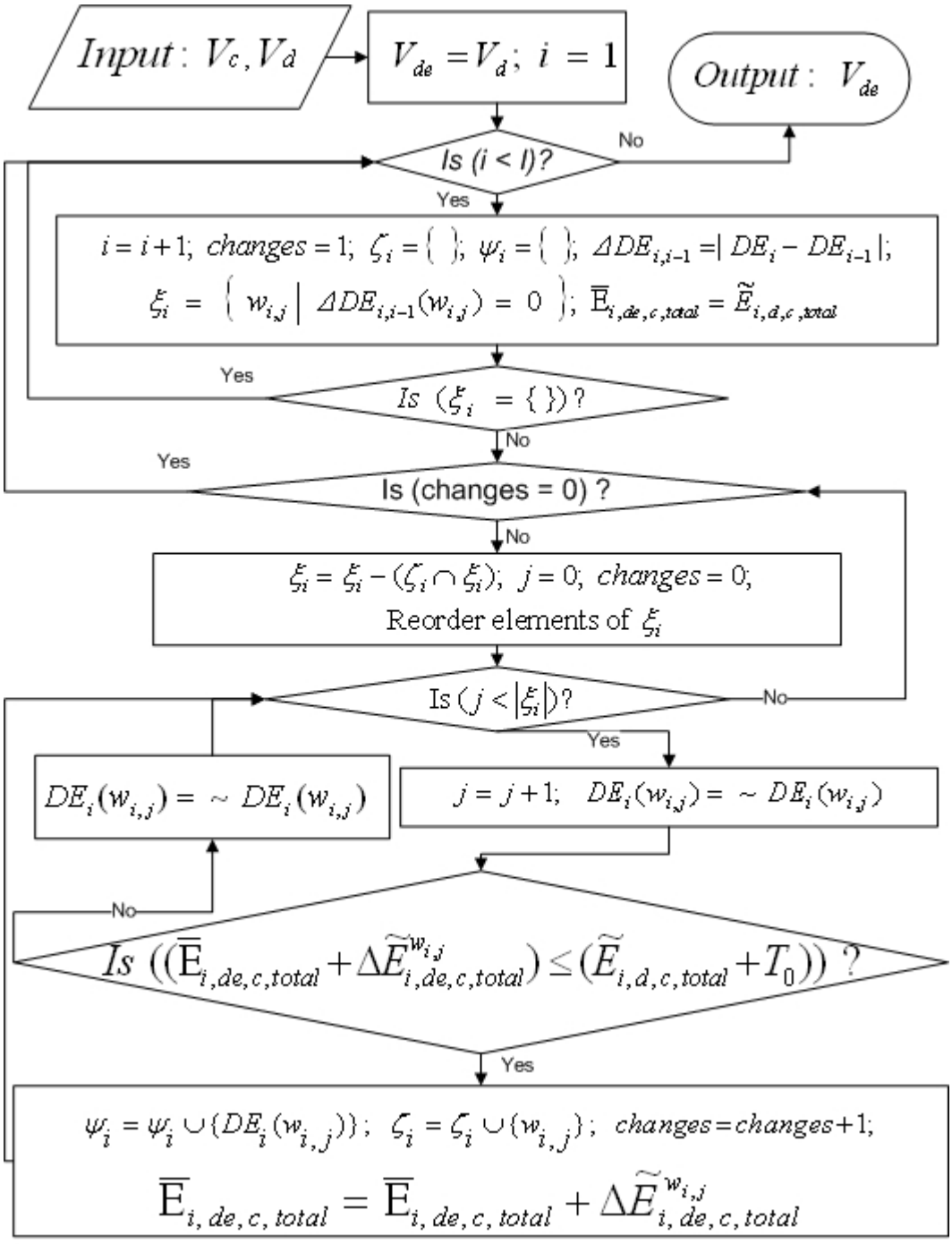


Figure 6.2: Binary halftone video enhancement via reduction of DWE in an efficient manner.

6.5 Implementation and Results

In this section, using the concepts of the algorithms described in this chapter, I present the results of reducing DWE in medium frame rate binary halftone videos. The implementation used to generate the results for enhanced videos is *based* on the efficient enhancement algorithm of Section 6.4. The implementation utilizes the general concepts of this chapter and is *not* an exact reflection of the flow charts of Figures 6.1 and 6.2. The results presented here are based on one particular implementation/instantiation of the *theoretical* concepts described in Section 6.4. Different implementations of the same algorithm could possibly produce different results.

There can be different possibilities for convergence criterion used to determine whether the processing of a frame was complete. For generating the enhancement results of this section, convergence criterion was checked differently than it was in the originally proposed methods of this chapter. For the implementation used to generate the halftone video enhancement results, the convergence criterion was checked after *two* full scans of the frame (as opposed to the suggestions of Figures 6.1 and 6.2). For the initial threshold, a value of $T_0 = 0$ was used. A trial change in the value of the pixel at location $w_{i,j}$ was accepted if $(\bar{E}_{i,de,c,total} + \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total})$ was less than $(\widetilde{E}_{i,d,c,total} + T_0)$. The error bounds predicted by the algorithms discussed in this chapter are *theoretical*. Since any implementation is also constrained by practical limitations (such as those sometimes encountered in handling the pixels at the boundaries of a frame), actual value of *introduced* additional perceptual error might be different than the theoretical prediction. Furthermore, recall from (5.9)

and (5.10) (Chapter 5) that the error metric, used to constrain the degradation of spatial quality of a frame, is dependent on the HVS filter implementation. Any filter used to represent the HVS is typically tuned to a particular application [20]. The tuning might require modification of the filter parameters to suit the needs of the display designer. It is up to the designer to choose a filter that produces the "best" results for his or her application. Here, I have used an HVS filter *based* on Nasanen's model [38] already discussed in Section 5.2. The filter was tuned for better performance on my LCD screen. For the design of my filter, the parameter values (see Section 5.2) are $a = 131.6$, $b = 0.3188$, $c = 0.525$, $d = 3.91$, and $L = 400$. I used a filter support of 11x11 pixels and assumed a screen resolution of 94 pixels per inch, as well as a viewing distance of 12 inches.

In the implementation of the efficient enhancement algorithm used to generate the results presented in this section, before attempting DWE reduction, some preprocessing was performed on the input halftone and continuous-tone videos. The continuous-tone video, V_c was preprocessed by performing an edge sharpening operation on each of its frames. The first frame of the halftone video, V_d, was improved using the DBS algorithm [64].

Since the FIOD method produces videos with excessive DWE (see Chapters 2, 3, and 4), for halftone video enhancement, I chose the videos generated using FIOD. Tables 6.1, 6.2, and 6.3 compare the performance of the videos generated using the FIOD algorithm with the enhanced videos. The DWE performance is evaluated using the DWE Index, DWE, of Chapter 2. The flicker performance is evaluated using the Flicker Index, F, of Chapter 2. The power performance is

assessed using the Power Index, P, of Chapter 4. The spatial quality of the halftone videos is evaluated using the spatial quality index, S, discussed in Section 5.5. For F, DWE, and P, a *lower* value indicates better performance. On the other hand, a *lower* value of S indicates worse performance.

Observe the results reported in Tables 6.1, 6.2, and 6.3. According to these tables, there is considerable improvement in DWE performance, as shown by a lower value of the DWE Index, DWE. However, considerable flicker has also been introduced as shown by an increase in the Flicker Index, F. The value of spatial quality measure S for the original and the enhanced halftone videos is generally fairly close indicating that the spatial quality of the halftone videos is not reduced by much, if at all, using my implementation of the enhancement algorithm. Table 6.3 displays the results for videos that have relatively higher spatial resolution (See Table 2.3 for a description of the resolution of these videos). It can be observed from Table 6.3 that for several 25 fps videos, the enhancement resulted in reduction of DWE as well as a slight improvement in the spatial quality of the frames.

Table 6.1: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 30 fps FIOD and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIOD	0.035	0.024	0.156	0.033
	Enhanced	0.499	0.385	0.077	0.034
Tempete	FIOD	0.037	0.025	0.062	0.107
	Enhanced	0.493	0.381	0.032	0.089
Miss America	FIOD	0.013	0.011	0.065	0.014
	Enhanced	0.286	0.25	0.046	0.014
Susie	FIOD	0.02	0.015	0.077	0.015
	Enhanced	0.529	0.463	0.037	0.015
Tennis	FIOD	0.035	0.019	0.115	0.075
	Enhanced	0.634	0.502	0.047	0.061
Trevor	FIOD	0.015	0.012	0.044	0.024
	Enhanced	0.382	0.323	0.026	0.024
Garden	FIOD	0.104	0.048	0.198	0.146
	Enhanced	0.571	0.318	0.087	0.127
Salesman	FIOD	0.013	0.011	0.04	0.035
	Enhanced	0.411	0.363	0.025	0.035
Football	FIOD	0.061	0.032	0.143	0.056
	Enhanced	0.659	0.478	0.058	0.052

Table 6.2: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 15 fps FIOD and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIOD	0.048	0.028	0.225	0.033
	Enhanced	0.505	0.346	0.109	0.035
Tempete	FIOD	0.056	0.033	0.118	0.107
	Enhanced	0.493	0.346	0.06	0.093
Miss America	FIOD	0.013	0.011	0.052	0.014
	Enhanced	0.286	0.254	0.036	0.014
Susie	FIOD	0.03	0.021	0.11	0.015
	Enhanced	0.532	0.446	0.053	0.015
Tennis	FIOD	0.046	0.023	0.138	0.075
	Enhanced	0.634	0.475	0.057	0.062
Trevor	FIOD	0.02	0.014	0.069	0.024
	Enhanced	0.383	0.311	0.04	0.025
Garden	FIOD	0.133	0.054	0.244	0.146
	Enhanced	0.591	0.289	0.106	0.132
Salesman	FIOD	0.008	0.006	0.018	0.035
	Enhanced	0.41	0.369	0.01	0.035
Football	FIOD	0.08	0.041	0.181	0.055
	Enhanced	0.668	0.451	0.07	0.053

Table 6.3: The Power Index, P, the Flicker Index, F, the DWE Index, DWE and, the Spatial Quality Index, S for the 25 fps FIOD and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Pedestrian-area	FIOD	0.033	0.021	0.077	0.025
	Enhanced	0.415	0.343	0.049	0.025
Rush-hour	FIOD	0.019	0.014	0.044	0.023
	Enhanced	0.378	0.323	0.028	0.023
Sunflower	FIOD	0.037	0.025	0.102	0.034
	Enhanced	0.331	0.253	0.071	0.034
Shields	FIOD	0.067	0.037	0.214	0.074
	Enhanced	0.382	0.25	0.135	0.075
Blue-sky	FIOD	0.071	0.031	0.148	0.093
	Enhanced	0.34	0.202	0.1	0.096
Station	FIOD	0.02	0.014	0.086	0.021
	Enhanced	0.396	0.312	0.054	0.021
Tractor	FIOD	0.06	0.034	0.214	0.041
	Enhanced	0.496	0.308	0.106	0.041

6.6 Summary

This chapter develops algorithms to enhance medium frame rate halftone videos. The algorithms described in this chapter enhance a binary halftone video by reducing DWE, which is a temporal artifact. The algorithms reduce DWE while controlling the amount of (spatial) degradation of individual frames. The control of the degradation of individual frame perceptual quality is achieved by using a tunable parameter. The relationship of this parameter to the amount of DWE reduction and the perceptual quality of constituent frames of the binary halftone video is described. The two algorithms described in this chapter differ in their computational requirements. The relative computational inefficiency of one of the algorithms was explained. The computationally more efficient algorithm of this chapter is compared against the computationally efficient algorithm of Chapter 5.

Chapter 7

Video Halftone Enhancement via Reduction of Temporal Artifacts under Spatial and Temporal Quality Constraints

In Chapters 5 and 6, I have designed algorithms that enhance a medium frame rate binary halftone video by reducing temporal artifacts while constraining the incurred "cost" of the halftone video's constituent frames' degradation in spatial perceptual quality. Chapters 5 and 6 have each solved the problem of reducing one temporal artifact of the halftone video while attempting to constrain the resulting degradation in the (spatial) quality of the frames of the video. Chapter 5 developed methods that can enhance a halftone video by reducing flicker. Chapter 6 developed methods that can reduce DWE in a halftone video. In each of Chapters 5 and 6, there was one main parameter, the threshold determining whether a pixel should change value, that established the main trade-off between temporal artifact (flicker or DWE) reduction and the resulting degradation in the perceptual quality of the individual frames of the enhanced halftone video. This chapter solves the relatively broader problem of reducing temporal artifacts under both spatial and temporal quality constraints.

In this chapter, I design algorithms that provide more control over the perceptual performance of the enhanced halftone video by introducing an additional

control parameter. Doing so introduces an additional constraint on the degradation of temporal quality of the halftone video. Spatial quality constraint is *still* enforced through the use of threshold T_0, as discussed in Chapters 5 and 6. The new algorithms impose temporal quality constraint by utilizing the temporal artifact assessment framework designed in Chapter 2. Recall that the enhancement algorithms of Chapters 5 and 6 are independent methods that do not depend on and, hence, do not utilize the framework developed in Chapter 2. The goal of this chapter is to design video halftone enhancement algorithms that build upon the contributions of Chapters 2, 5, and 6.

In each of the two preceding chapters, a threshold, denoted by T_0, determined any *additional* spatial perceptual error introduced in each frame of the enhanced halftone video. The trade-off between the degree to which a temporal artifact could be reduced and the resulting increase in spatial perceptual error was discussed in Chapters 5 and 6. Higher value of T_0 means higher reduction of the temporal artifact (flicker of DWE). Higher T_0 also, however, implies relatively more perceptual degradation of each frame of the halftone video. Since, a higher value of the threshold T_0 can result in more pixel value changes accepted in each frame, it implies more computation as well!

Since, flicker and DWE are related, T_0 has an additional effect as well! For the flicker reduction algorithms, a higher T_0 can result in more reduction of flicker and, hence, potentially a higher DWE in the enhanced halftone video. For the DWE reduction algorithms, a higher T_0 can potentially increase flicker while reducing DWE. While using any of the algorithms developed in Chapters 5 and 6, it is up to

the display device designer to choose a value of T_0 that best meets his or her display device's constraints including perceptual performance.

Using T_0 to control the trade-off between flicker and DWE is not the best way to go because modifying T_0 impacts the spatial quality of the frames. It would be nice to have some other parameter establish a balance between flicker and DWE. This way T_0 can be left alone to do its job, that is to constrain the degradation in perceptual quality of the frames. The algorithms designed in this chapter introduce additional parameters, besides the T_0 of the preceding two chapters, to provide additional control over the resulting perceptual performance of the enhanced halftone video. The introduction of new parameters is achieved by incorporating the artifact assessment framework of Chapter 2. The algorithms developed in this chapter are "selective" in the sense that the locations of pixels considered for changes are selected based on the artifact assessment criteria of Chapter 2.

This chapter begins by describing a modification to the algorithms of Chapter 5 to selectively reduce flicker. Video halftone enhancement results obtained by incorporating the proposed changes (i.e. the introduction of an additional control parameter in the algorithms) are discussed. This is followed by a discussion of a modification to the algorithms of Chapter 6. Results of applying this modification to the concepts of Chapter 6 are presented. The chapter concludes with a summary of the presented developments.

7.1 Reduction of Flicker under Spatial and Temporal Quality Constraints

The goal of this section is to design algorithms that reduce flicker while constraining the *additional* degradation (resulting due to flicker reduction) in the spatial quality of each frame as well as in the DWE performance of the entire frame sequence. The proposed algorithms are designed by modifying the enhancement algorithms of Chapter 5.

The reader is encouraged to refer to Section 2.4.1 to remind himself or herself of the notation introduced in Chapter 2. Recall that D_i is the i^{th} frame of the halftone video, V_d. Recall from Chapter 5 that DE_i is the i^{th} frame of the *enhanced* halftone video, V_{de}. Based on the notation used in Section 6.2, for $i > 1$, each pixel $\Delta DE_{i,i-1}(m,n)$ of the i^{th} absolute difference image, $\Delta DE_{i,i-1}$ is given by:

$$\Delta DE_{i,i-1}(m,n) = |DE_i(m,n) - DE_{i-1}(m,n)| \tag{7.1}$$

Since this dissertation relates to binary halftones, $\Delta DE_{i,i-1}(m,n) \in \{0,1\}$. Consider the case when $\Delta DE_{i,i-1}(m,n) = 1$. This indicates that the binary pixels at spatial location (m,n) in the adjacent halftone frames DE_i, and DE_{i-1} have *different* values . Based on the discussion in Chapter 2, note that this can contribute to the perception of flicker in a medium frame rate binary halftone video. Consequently, in Chapter 5, halftone video enhancement was achieved by changing $DE_i(m,n)$ such that its value was equal to $DE_{i-1}(m,n)$. This was done under the constraint that the resulting total perceptual error of DE_i, as defined in Chapter 5, did not exceed

the total perceptual error of D_i by more than the threshold T_0.

Recall that in Chapter 5, ξ_i is defined to be the ordered set of pixel locations that have different values between the two adjacent frames DE_i and DE_{i-1}. The elements of ξ_i are indexed by j and the j^{th} element of ξ_i is denoted by $w_{i,j}$. ξ_i is formally defined by (5.16). In the algorithms of Chapter 5, all pixel locations belonging to ξ_i are candidates for a trial change. Whether to accept the change or not, in the process of reducing flicker, is determined by a comparison of a perceptual error measure, described in Chapter 5, with a threshold T_0. The discussion and development of temporal artifact assessment framework in Chapter 2, however, reveals that not all pixel locations belonging to ξ_i need to be considered for a trial change.

In this section, I modify the algorithms of Chapter 5 by changing the definition of ξ_i. The definition is changed based on the artifact assessment framework of Chapter 2. Recall from Section 2.4.4 that a higher value of the product $SSIM\{C_i, C_{i-1}\}(m,n)\cdot(1-W_i(m,n))$ means that any pixel toggle at location (m,n) potentially results in a (correspondingly) higher value of perceived flicker. Therefore, the locations of pixels whose values should be trial changed in the enhanced frame DE_i should be determined by evaluating $SSIM\{C_i, C_{i-1}\}(m,n)\cdot(1-W_i(m,n))$. I define ξ_i as

$$\xi_i = \{w_{i,j} : (DE_i(w_{i,j}) \neq DE_{i-1}(w_{i,j}))\} \cap \{w_{i,j} : SSIM\{C_i, C_{i-1}\}(w_{i,j})\cdot(1-W_i(w_{i,j})) > \tau_f\}, \quad (7.2)$$

where τ_f is a threshold that, besides the threshold T_0 discussed in Chapter 5, controls the degree by which flicker gets reduced. Thus, τ_f, and T_0 are the two parameters of the algorithms of this section. The value of T_0, as discussed at the beginning of this chapter, impacts the flicker (and hence also DWE), and the individual frame quality of the enhanced video V_{de}. The primary purpose of using T_0 is to constrain any spatial degradation of the frame as a result of post-processing. τ_f establishes a direct trade-off between flicker and DWE of the enhanced video. A lower value of τ_f means that possibly more pixels will be trial changed. This could result in a lower flicker, but also a higher DWE. Regardless of the value of τ_f, any degradation in the perceptual quality of individual frames of V_{de} is still controlled by T_0.

7.1.1 Results

I now present the results of reducing flicker in medium frame rate binary halftone videos. The enhancement algorithm implementation used for producing these results is *based* on the flicker reduction concepts discussed in this section (Section 7.1). The general flow of algorithm implementation is *based* on Figure 5.2 but with ξ_i defined by (7.2). The implementation utilizes the general concepts of this chapter and Chapter 5, but it is *not* an exact reflection of the flow charts of Figures 5.1 and 5.2. The results presented here are based on one particular implementation/instantiation of the *theoretical* concepts described this section and in Section 5.4. Different implementations of the same algorithm could potentially result in variation of results.

As was done for the implementation discussed in Section 5.6, in the imple-

mentation used to generate the halftone video enhancement results presented in this section, the convergence criterion is checked after *two* full scans of the frame. The two successive scans, that are completed before checking convergence criterion, comprise of a horizontal raster scan and a vertical raster scan. For the initial threshold, a value of $T_0 = 0$ is used. Also, a trial change at a pixel location $w_{i,j}$ is accepted if $\Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total} < T_{w_{i,j}}$. Furthermore, $\tau_f = 0.7$ and and for the design of my filter, the parameter values are identical to those used for generating the results discussed in Section 5.6.

Before running the video enhancement algorithm, some preprocessing is performed on the input halftone and continuous-tone videos. The continuous-tone video, V_c is preprocessed by performing an edge sharpening operation on each of its frames. The first frame of the halftone video, V_d, is improved using DBS algorithm [64]. For halftone video enhancement, I chose the videos generated using FIFSED.

Tables 7.1 and 7.2 compare the performance of the input (FIFSED) and the enhanced halftone videos. The flicker performance is evaluated using the Flicker Index, F, of Chapter 2. The DWE performance is evaluated using the DWE Index, DWE, of Chapter 2. The power performance is assessed using the Power Index, P, of Chapter 4. The spatial quality of the halftone videos is evaluated using the spatial quality index, S, discussed in Section 5.5. For F, DWE, and P, a *lower* value indicates better performance. On the other hand, a higher value of S indicates better performance.

As can be seen from the data in Tables 7.1 and 7.2, the spatial quality of the

two (input and enhanced) videos is fairly close for the tested sequences. Compare these results with the results reported in Chapter 5 (Tables 5.1 and 5.2). Tables 7.1 and 7.2 show a relatively less reduction in flicker in halftones enhanced using the modified algorithm of this chapter. At the same time, the increase in DWE is smaller than the increase observed in Tables 5.1 and 5.2 of Chapter 5. The additional (control) parameter τ_f is used to balance flicker and DWE. It has done its job!

7.2 Reduction of DWE under Spatial and Temporal Quality Constraints

The goal of this section is to propose algorithms that reduce DWE while constraining the *resulting* degradation in the spatial quality of each frame as well as in the flicker performance of the entire frame sequence. The proposed algorithms are designed by modifying the enhancement algorithms of Chapter 6.

Recall from the discussion in the previous section that $\Delta DE_{i,i-1}(m,n) \in \{0,1\}$. When $\Delta D_{i,i-1}(m,n) = 0$, the binary pixels at the spatial location (m,n) in the adjacent halftone frames DE_i, and DE_{i-1} have the *same* value. Based on the discussion in Chapter 2, note that this can contribute to the perception of DWE in a medium frame rate binary halftone video. Accordingly, in Chapter 6, halftone video enhancement was achieved by changing $DE_i(m,n)$ such that its value was not equal to $DE_{i-1}(m,n)$. This was done under the constraint that any resulting increase in the perceptual error did not exceed the threshold T_0. This process is detailed in Figures 6.1 and 6.2.

In Chapter 6, ξ_i is defined to be the ordered set of pixel locations that have

Table 7.1: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 30 fps FIFSED and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIFSED	0.427	0.333	0.092	0.041
	Enhanced	0.345	0.265	0.103	0.039
Tempete	FIFSED	0.34	0.266	0.042	0.141
	Enhanced	0.258	0.199	0.047	0.14
Miss America	FIFSED	0.3	0.262	0.044	0.015
	Enhanced	0.235	0.206	0.049	0.015
Susie	FIFSED	0.456	0.4	0.043	0.017
	Enhanced	0.31	0.271	0.055	0.016
Tennis	FIFSED	0.436	0.344	0.066	0.096
	Enhanced	0.181	0.135	0.095	0.084
Trevor	FIFSED	0.366	0.310	0.027	0.029
	Enhanced	0.216	0.183	0.034	0.029
Garden	FIFSED	0.408	0.232	0.127	0.19
	Enhanced	0.387	0.217	0.131	0.188
Salesman	FIFSED	0.361	0.319	0.026	0.044
	Enhanced	0.201	0.178	0.032	0.045
Football	FIFSED	0.457	0.329	0.087	0.068
	Enhanced	0.434	0.312	0.09	0.067

Table 7.2: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 15 fps FIFSED and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIFSED	0.429	0.3	0.134	0.041
	Enhanced	0.374	0.256	0.143	0.04
Tempete	FIFSED	0.358	0.254	0.079	0.141
	Enhanced	0.288	0.2	0.086	0.138
Miss America	FIFSED	0.299	0.267	0.036	0.016
	Enhanced	0.22	0.196	0.04	0.015
Susie	FIFSED	0.458	0.385	0.063	0.017
	Enhanced	0.35	0.291	0.073	0.016
Tennis	FIFSED	0.444	0.330	0.080	0.096
	Enhanced	0.217	0.151	0.111	0.084
Trevor	FIFSED	0.367	0.301	0.042	0.029
	Enhanced	0.247	0.2	0.05	0.029
Garden	FIFSED	0.421	0.211	0.16	0.19
	Enhanced	0.406	0.201	0.163	0.189
Salesman	FIFSED	0.357	0.323	0.011	0.044
	Enhanced	0.208	0.188	0.014	0.046
Football	FIFSED	0.468	0.314	0.109	0.067
	Enhanced	0.461	0.309	0.111	0.067

the same values between the two adjacent frames DE_i and DE_{i-1}. The elements of ξ_i are indexed by j and the j^{th} element of ξ_i is denoted by $w_{i,j}$. For the algorithms of Chapter 6, ξ_i is defined by (6.7). For these algorithms (Figures 6.1 and 6.2), all pixel locations belonging to ξ_i are candidates for a trial change. Whether to accept the change or not, in the process of reducing DWE, is determined by a comparison of a perceptual error measure, described in Chapter 5, with a threshold T_0. The discussion and development of the DWE assessment framework in Chapter 2, however, suggests that not all pixel locations belonging to ξ_i contribute to the perception of DWE equally. Therefore, not every pixel location belonging to ξ_i (as defined by (6.7)) need be considered for a trial change.

In this section, I modify the algorithms depicted in Figures 6.1 and 6.2 by changing the definition of ξ_i. The definition is changed based on the DWE assessment framework of Chapter 2. Recall from Section 2.4.2 that a higher value of the product $(1 - SSIM\{C_i, C_{i-1}\}(m,n)) \cdot (1 - W_i(m,n))$ means that any pixels that have the same value at location (m,n) of successive halftone frames contribute more to the perception of DWE. Hence, the pixel locations whose values should be trial changed in the enhanced frame DE_i can be determined by evaluating the product $(1 - SSIM\{C_i, C_{i-1}\}(m,n)) \cdot (1 - W_i(m,n))$. I redefine ξ_i as

$$\begin{aligned}\xi_i = &\{w_{i,j} : (DE_i(w_{i,j}) = DE_{i-1}(w_{i,j}))\} \\ &\cap \{w_{i,j} : (1 - SSIM\{C_i, C_{i-1}\}(w_{i,j})) \cdot (1 - W_i(w_{i,j})) > \tau_{dwe}\}, \quad (7.3)\end{aligned}$$

where τ_{dwe} is a threshold that, besides the threshold T_0 discussed in Chapter 6,

controls the degree by which DWE gets reduced. τ_{dwe}, and T_0 are the two parameters of the algorithms of this section. The value of T_0, as discussed at the beginning of this chapter, mainly impacts the reduction of DWE, and the individual frame quality of the enhanced video V_{de}. τ_{dwe} controls reduction of DWE, and hence the introduction of flicker. A lower value of τ_{dwe} means that possibly more pixels will be trial changed. This is so because the number of elements in ξ_i is (also based on the definition of (7.3)) dependent on τ_{dwe}. Lower τ_{dwe} could possibly result in a lower DWE, if the trial changes are accepted. This could also result in higher flicker. Regardless of the value of τ_{dwe}, any degradation in the perceptual quality of individual frames of V_{de} is still controlled by T_0. Thus, T_0 is used to constrain the degradation in spatial quality and τ_{dwe} is used to constrain the degradation in flicker performance.

7.2.1 Results

I now present the results of reducing DWE in medium frame rate binary halftone videos. The enhancement algorithm implementation used for producing these results is *based* on the DWE reduction concepts discussed in this section (Section 7.2). The general flow of algorithm implementation is *based* on Figure 6.2 but with ξ_i defined by (7.3). The implementation utilizes the general concepts of this chapter and Chapter 6, but it is *not* an exact reflection of the flow charts of Figures 6.1 or 6.2. The results presented here are based on one particular implementation/instantiation of the *theoretical* concepts described in this section and in Section 6.4. Different implementations of the same algorithm could potentially produce different results.

As was done for the implementation discussed in Section 6.5, in the implementation used to generate the halftone video enhancement results presented in this section, the convergence criterion is checked after *two* full scans of the frame. The two successive scans, that are completed before checking convergence criterion, comprise of a horizontal raster scan and a vertical raster scan. For the initial threshold, a value of $T_0 = 0$ is used. Also, a trial change in the value of the pixel at location $w_{i,j}$ is accepted if $(\bar{E}_{i,de,c,total} + \Delta\widetilde{E}^{w_{i,j}}_{i,de,c,total})$ is less than $(\widetilde{E}_{i,d,c,total} + T_0)$. Furthermore, $\tau_{dwe} = 0.08$ and for the design of my filter, the parameter values are identical to those used for generating the results discussed in Section 6.5.

Before running the video enhancement algorithm, some preprocessing is performed on the input halftone and continuous-tone videos. The continuous-tone video, V_c is preprocessed by performing an edge sharpening operation on each of its frames. The first frame of the halftone video, V_d, is improved using DBS algorithm [64]. For halftone video enhancement, I chose the videos generated using the FIOD algorithm.

Tables 7.3 and 7.4 compare the performance of the input (FIOD) and the enhanced halftone videos. DWE performance is evaluated using the DWE Index, DWE, of Chapter 2. Flicker performance is evaluated using the Flicker Index, F, of Chapter 2. Power performance is assessed using the Power Index, P, of Chapter 4. The spatial quality of the halftone videos is evaluated using the spatial quality index, S, discussed in Section 5.5. For F, DWE, and P, a *lower* value indicates better performance. On the other hand, a higher value of S indicates better performance.

As can be seen from the data in Tables 7.3 and 7.4, the spatial quality of the

two (input and enhanced) videos is fairly close for the tested sequences. Compare these results with the results reported in Chapter 6 (Tables 6.1 and 6.2). Tables 7.3 and 7.4 show a relatively less reduction in DWE in halftones enhanced using the modified algorithm of this chapter. At the same time, the increase in flicker is smaller than the increase observed in Tables 6.1 and 6.2 of Chapter 6. This is what was expected of the enhancement algorithm modification proposed in this section. The additional (control) parameter τ_{dwe} is used to provide a balance between DWE and flicker performance of the enhanced video.

7.3 Summary

In the process of enhancing a medium frame rate binary halftone video, reducing one temporal artifact can result in an increase of the other artifact. For example, reduction of flicker can potentially introduce dirty-window-effect in the enhanced halftone video. To enable better control over achieving a balance between the "reduced" and the "introduced" artifacts, this chapter proposes modifications to the algorithms of Chapters 5 and 6. Additional parameters are introduced in the process of modifying these algorithms. This results in new algorithms that can enhance medium frame rate binary halftone videos under both spatial and temporal quality constraints.

Table 7.3: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 30 fps FIOD and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIOD	0.035	0.024	0.156	0.033
	Enhanced	0.333	0.248	0.098	0.034
Tempete	FIOD	0.037	0.025	0.062	0.107
	Enhanced	0.113	0.081	0.055	0.103
Miss America	FIOD	0.013	0.011	0.065	0.014
	Enhanced	0.071	0.058	0.057	0.014
Susie	FIOD	0.02	0.015	0.077	0.015
	Enhanced	0.057	0.045	0.073	0.015
Tennis	FIOD	0.035	0.019	0.115	0.075
	Enhanced	0.08	0.054	0.109	0.074
Trevor	FIOD	0.015	0.012	0.044	0.024
	Enhanced	0.063	0.051	0.04	0.024
Garden	FIOD	0.104	0.048	0.198	0.146
	Enhanced	0.491	0.26	0.102	0.128
Salesman	FIOD	0.013	0.011	0.04	0.035
	Enhanced	0.045	0.038	0.037	0.035
Football	FIOD	0.061	0.032	0.143	0.056
	Enhanced	0.639	0.462	0.059	0.053

Table 7.4: The Power Index, P, the Flicker Index, F, the DWE Index, DWE, and the Spatial Quality Index, S for the 15 fps FIOD and enhanced halftone videos. *Lower* values of F, DWE, and P indicate *better* performance. A *lower* value of S indicates *worse* performance.

Video	**Halftone**	P	F	DWE	S
Caltrain	FIOD	0.048	0.028	0.225	0.033
	Enhanced	0.392	0.257	0.128	0.035
Tempete	FIOD	0.056	0.033	0.118	0.107
	Enhanced	0.254	0.164	0.085	0.099
Miss America	FIOD	0.013	0.011	0.052	0.014
	Enhanced	0.045	0.037	0.049	0.014
Susie	FIOD	0.03	0.021	0.11	0.015
	Enhanced	0.209	0.161	0.08	0.015
Tennis	FIOD	0.046	0.023	0.138	0.075
	Enhanced	0.111	0.069	0.128	0.072
Trevor	FIOD	0.02	0.014	0.069	0.024
	Enhanced	0.109	0.084	0.058	0.024
Garden	FIOD	0.133	0.054	0.244	0.146
	Enhanced	0.53	0.246	0.119	0.132
Salesman	FIOD	0.008	0.006	0.018	0.035
	Enhanced	0.015	0.013	0.018	0.035
Football	FIOD	0.08	0.041	0.181	0.055
	Enhanced	0.66	0.446	0.07	0.053

Chapter 8

Conclusion

This dissertation presents several contributions in the areas of video halftone artifact assessment, generation, and enhancement. This dissertation also studies the relationship between the halftones generated using several algorithms and the associated power consumption on a bistable display device.

In Chapter 1, the dissertation begins by providing a general introduction to the problem of displaying image or video data on limited bit-depth display devices. Halftoning is discussed as a solution to this problem. Properties of the human visual system crucial to the success of halftoning are presented. Lessons learned from the use of human visual system models in previous contributions in the areas of image and video halftoning are discussed.

In Chapter 2, typical quantization artifacts that result due to bit-depth reduction are introduced to the reader. The chapter then develops a generalized framework for the assessment of two key temporal artifacts, flicker and dirty-window-effect, typical to binary video halftones produced from grayscale continuous-tone videos and displayed at frame rates ranging between 15 to 30 frames per second. A visual inspection study is designed. The performance of the temporal artifact assessment framework is evaluated by comparing the objective artifact assessment

results with the results of the visual inspection study.

Following the presentation of the development of the temporal artifact assessment framework, in Chapter 3, the dissertation presents the design of two video halftone generation algorithms, each aimed at reducing one temporal artifact. An iterative video halftoning algorithm is designed to generate medium frame rate binary video halftones with reduced dirty-window-effect. A neighborhood based video halftoning algorithm is designed to generate medium frame rate binary video halftones with reduced flicker. Performance of the algorithms is evaluated both objectively and subjectively.

In Chapter 4, the dissertation presents an analysis of consumption of power by the display component of a bistable display multimedia device. Bistable display technology is compared with the prevalent conventional display technology used in most currently used handheld multimedia devices. A comparison of the performance of five different video halftoning algorithms in terms of the power requirements, and the degree of temporal artifacts present in the halftone videos generated by these algorithms is presented.

In Chapter 5, the dissertation proposes methods for reducing flicker in medium frame rate binary halftone videos. The proposed methods reduce flicker under the constraint that, in the process of flicker reduction, the degradation in spatial quality of the halftone frames is controlled. To enhance a halftone video, the methods discussed in Chapter 5 do not utilize the flicker assessment framework developed in this dissertation. Spatial and temporal performance of videos enhanced through an implementation of the main concepts of the chapter is evaluated using objective

quality measures.

In Chapter 6, the dissertation develops halftone post-processing algorithms to reduce DWE in a medium frame rate binary halftone video while attempting to preserve the spatial perceptual quality of the video's frames. To enhance a halftone video by reducing DWE, the proposed algorithms do not utilize the DWE assessment framework developed in this dissertation. Spatial and temporal performance of videos enhanced through an implementation of the introduced ideas is evaluated using objective quality measures.

Finally, in Chapter 7, this dissertation proposes modifications to the video halftone enhancement techniques introduced in Chapters 5 and 6. The proposed modifications result in video halftone enhancement algorithms that enable additional control on how much a temporal artifact gets reduced during enhancement. This additional control is gained by incorporating the temporal artifact assessment framework developed in Chapter 2.

As the technology advances, power efficient reflective bistable devices will become more capable in terms of bit-depth and supported frame rates for video display. A future direction for research would be to explore quality assessment techniques that can be applied to multilevel halftones. The quality assessment techniques presented in this dissertation have not been tested on multilevel or color halftone videos. Color video halftoning also seems to be a promising area of research. It is my hope that the research presented in this dissertation sets up the ground work for exploring either of these avenues of future research.

Bibliography

[1] C. Hsu, C. Lu, and S. Pei, "Power-scalable multi-layer halftone video display for electronic paper," in *Proc. IEEE International Conference on Multimedia and Expo*, 2008, pp. 1445–1448.

[2] J. Robson, "Spatial and temporal contrast-sensitivity functions of the visual system," *Journal of the Optical Society of America*, vol. 56, no. 8, pp. 1141–1142, 1966.

[3] A. B. Watson, *Handbook of Perception and Human Performance*. Wiley, 1986, ch. Temporal Sensitivity.

[4] J. Allebach, "Human vision and image rendering: is the story over, or is it just beginning?" in *Proceedings of SPIE*, vol. 3299, 1998, p. 26.

[5] B. Jahne, *Digital Image Processing*. Springer-Verlag, 2005.

[6] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Prentice Hall, 2002.

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

[8] R. Stevenson and G. Arce, "Binary display of hexagonally sampled continuous-tone images," *Journal of the Optical Society of America A*, vol. 2, no. 7, pp. 1009–1013, 1985.

[9] J. G. Webster, *Electrical Measurement, Signal Processing, and Displays.* CRC Press, 2003.

[10] A. K. Bhowmik, Z. Li, and P. J. Bos, *Mobile Displays: Technology and Applications.* John Wiley & Sons Ltd., 2008.

[11] H. J. Trussell and M. J. Vrhel, *Fundamentals of Digital Imaging.* Cambridge University Press, 2008.

[12] J. C. Russ, *The Image Processing Handbook.* CRC Press, 1999.

[13] R. Hattori, S. Yamada, Y. Masuda, and N. Nihei, "A novel bistable reflective display using quick-response liquid powder," *Journal of the SID*, vol. 20, no. 1, pp. 75–80, 2004.

[14] Z. XiE and S. Hoi, "Reflective bistable twisted nematic liquid crystal display," *Jpn. J. Appi. Phys. Vol*, vol. 37, pp. 2572–2575, 1998.

[15] R. Barberi, M. Giocondo, J. Li, R. Bartolino, I. Dozov, and G. Durand, "Fast bistable nematic display with grey scale," *Applied Physics Letters*, vol. 71, p. 3495, 1997.

[16] D. Berreman and W. Heffner, "New bistable cholesteric liquid-crystal display," *Applied Physics Letters*, vol. 37, p. 109, 1980.

[17] J. B. Mulligan, "Methods for spatiotemporal dithering," in *Proc. SID Int. Symp., Dig. Tech. Papers*, Seattle, WA, 1993, pp. 155–158.

[18] D. L. Lau and G. R. Arce, *Modern Digital Halftoning.* CRC Press, 2008.

[19] T. Pappas and D. Neuhoff, "Least-squares model-based halftoning," *IEEE Transactions on Image Processing*, vol. 8, no. 8, pp. 1102–1116, 1999.

[20] T. Pappas, J. Allebach, and D. Neuhoff, "Model-based digital halftoning," *IEEE Signal Processing Magazine*, vol. 20, no. 4, pp. 14–27, July 2003.

[21] B. A. Wandell, *Foundations of Vision*. Sinauer, 1995.

[22] S. Westen, R. Lagendijk, and J. Biemond, "Perceptual image quality based on a multiple channel HVS model," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, 1995, pp. 2351–2351.

[23] D. Heeger and P. Teo, "A model of perceptual image fidelity," in *International Conference on Image Processing*, vol. 2, 1995.

[24] S. Daly, *Digital Images and Human Vision*. MIT Press, 1993, ch. The visible difference predictor: An algorithm for the assess- ment of image fidelity, pp. 179–205.

[25] A. Watson, "The cortex transform: rapid computation of simulated neural images," *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 311–327, 1987.

[26] J. Lubin, *Vision Models for Target Detection and Recognition*. World Scientific, 1995, ch. A visual discrimination model for imaging systems design and evaluation, pp. 245–283.

[27] ——, *Digital images and human vision.* Cambridge, MA: MIT Press, 1993, ch. The use of psychophysical data and models in the analysis of display system performance, pp. 163–178.

[28] D. Lieberman and J. Allebach, "A dual interpretation for direct binary search and its implicationsfor tone reproduction and texture quality," *IEEE Transactions on Image Processing*, vol. 9, no. 11, pp. 1950–1963, 2000.

[29] H. Kang, *Digital color halftoning.* SPIE/IEEE, 1999.

[30] J. Sullivan, L. Ray, and R. Miller, "Design of minimum visual modulation halftone patterns," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 21, no. 1, pp. 33 –38, jan/feb 1991.

[31] B. Kolpatzik and C. Bouman, "Optimized error diffusion for high quality image display," *Journal of Electronic Imaging*, vol. 1, no. 3, pp. 277–292, 1992.

[32] D. Neuhoff, T. Pappas, and N. Seshadri, "One-dimensional least-squares model-based halftoning," *Journal of the Optical Society of America*, vol. 14, no. 8, pp. 1707–1723, 1997.

[33] J. Sullivan, R. Miller, and G. Pios, "Image halftoning using a visual model in error diffusion," *Journal of the Optical Society of America*, vol. 10, no. 8, pp. 1714–1723, 1993.

[34] T. Mitsa and K. Varkur, "Evaluation of contrast sensitivity functions for the formulation of quality measures incorporated in halftoning algorithms," in

IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 5, 27-30 1993, pp. 301 –304.

[35] A. Agar and J. Allebach, "Model-based color halftoning using direct binary search," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 1945 –1959, dec. 2005.

[36] F. W. Campbell, R. H. S. Carpenter, and J. Z. Levinson, "Visibility of aperiodic patterns compared with that of sinusoidal gratings," *The Journal of Physiology*, vol. 204, no. 2, p. 283, 1969.

[37] J. Mannos and D. Sakrison, "The effects of a visual fidelity criterion of the encoding of images," *IEEE Transactions on Information Theory*, vol. 20, no. 4, pp. 525–536, 1974.

[38] R. Nasanen, "Visibility of halftone dot textures," *IEEE transactions on systems, man, and cybernetics*, vol. 14, no. 6, pp. 920–924, 1984.

[39] S. Daly, "Subroutine for the generation of a two dimensional human visual contrast sensitivity function," Eastman Kodak, Tech. Rep. Tech. Rep. 233203y, 1987.

[40] H. K. Sang and J. Allebach, "Impact of HVS models on model-based halftoning," *IEEE Transactions on Image Processing*, vol. 11, no. 3, pp. 258–269, 2002.

[41] R. Sekuler and R. Blake, *Perception*. McGraw-Hill, 2006.

[42] F. Campbell, J. Kulikowski, and J. Levinson, "The effect of orientation on the visual resolution of gratings," *J. Physiology London*, vol. 187, pp. 427–436, 1966.

[43] A. J. González, J. B. Rodríguez, G. R. Arce, and D. L. Lau, "Alpha stable modeling of human visual systems for digital halftoning in rectangular and hexagonal grids," *Journal of Electronic Imaging*, vol. 17, no. 1, p. 013004, 2008.

[44] D. Kelly, "Motion and vision. II. Stabilized spatio-temporal threshold surface," *Journal of the Optical Society of America*, vol. 69, no. 10, pp. 1340–1349, 1979.

[45] C. Atkins, T. Flohr, D. Hilgenberg, C. Bouman, and J. Allebach, "Model-based color image sequence quantization," *Proc. SPIE: Human Vision, Visual Processing, and Digital Display V*, pp. 310–309.

[46] D. P. Hilgenberg, T. J. Flohr, C. B. Atkins, J. P. Allebach, and C. A. Bouman, "Least-squares model-based video halftoning," *Human Vision, Visual Processing, and Digital Display V*, vol. 2179, no. 1, pp. 207–217, 1994.

[47] R. Floyd and L. Steinberg, "An adaptive algorithm for spatial grayscale," in *Proc. SID Int. Symp., Dig. Tech. Papers*, 1976, p. 3637.

[48] Z. Sun, "Video halftoning," *IEEE Trans. on Image Processing*, vol. 15, no. 3, pp. 678–686, 2006.

[49] C. Gotsman, "Halftoning of image sequences," *The Visual Computer*, vol. 9, no. 5, pp. 255–266, 1993.

[50] R. A. Ulichney, "Review of halftoning techniques," in *Proc. SPIE Conf. on Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts*, vol. 3963, no. 1, 1999, pp. 378–391.

[51] ——, "Dithering with blue noise," in *Proc. IEEE*, vol. 76, Jan. 1988, pp. 56–79.

[52] T. Mitsa and K. Parker, "Digital halftoning technique using a blue-noise-mask," *Journal of Optical Society of America*, vol. 9, pp. 1920–1929, 1992.

[53] K. Spaulding, R. Miller, and J. Schildkraut, "Methods for generating blue-noise dither matrices for digital halftoning," *J. Electronic Imaging*, vol. 6, no. 2, pp. 208–230, 1997.

[54] R. A. Ulichney, "Void-and-cluster method for dither array generation," in *IS&T/SPIE Symposium on Electronic Imaging Science & Technology*, vol. 1913, no. 1. SPIE, 1993, pp. 332–343.

[55] Z. Fan and R. Eschbach, "Limit cycle behavior of error diffusion," in *IEEE International Conference on Image Processing*, vol. 2, 1994.

[56] Z. Fan, "Stability analysis of error diffusion," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, 1993.

[57] T. Pappas and D. Neuhoff, "Printer models and error diffusion," *IEEE Transactions on Image Processing*, vol. 4, no. 1, pp. 66–80, 1995.

[58] T. D. Kite, B. L. Evans, and A. C. Bovik, "Modeling and quality assessment of halftoning by error diffusion," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 909–922, 2000.

[59] K. T. Knox, "Evolution of error diffusion," *Journal of Electronic Imaging*, vol. 8, no. 4, pp. 422–429, 1999.

[60] V. Ostromoukhov, "A simple and efficient error-diffusion algorithm," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, p. 572.

[61] R. Eschbach, Z. Fan, K. T. Knox, and G. Marcu, "Threshold modulation and stability in error diffusion," *IEEE Signal Processing Magazine*, vol. 20, no. 4, pp. 39–50, July 2003.

[62] B. L. Evans, V. Monga, and N. Damera-Venkata, "Variations on error diffusion: retrospectives and future trends," in *Color Imaging VIII: Processing, Hardcopy, and Applications*, vol. 5008, no. 1. SPIE, 2003, pp. 371–389.

[63] M. Analoui and J. Allebach, "Model-based halftoning using direct binary search," in *Proceedings of SPIE*, vol. 1666, 1992, p. 96.

[64] D. Lieberman and J. Allebach, "Efficient model based halftoning using direct binary search," in *Proceedings of the IEEE International Conference on Image Processing*, 1997, pp. 775–778.

[65] H. Hild and M. Pins, "A 3-d error diffusion dither algorithm for half-tone animation on bitmap screens," in *State-of-the-Art in Computer Animation*. Berlin, Germany: Springer-Verlag, 1989, p. 181190.

[66] C. Hsu, C. Lu, and S. Pei, "Video halftoning preserving temporal consistency," in *Proc. IEEE Int. Conf. on Multimedia and Expo*, 2007, pp. 1938–1941.

[67] K. Seshadrinathan, "Video quality assessment based on motion models," Ph. D., Department of Electrical and Computer Engineering, The University of Texas at Austin, USA, 2008.

[68] S. Winkler, *Digital Video Quality: vision models and metrics*. Chichester, West Sussex, England: John Wiley & Sons Ltd., 2005.

[69] H. R. Wu and K. R. Rao, Eds., *Digital Video Image Quality and Perceptual Coding*. Boca Raton, FL: CRC Press, 2006.

[70] M. Pedersen, F. Albregtsen, and J. Y. Hardeberg, "Detection of worms in error diffusion halftoning," in *Image Quality and System Performance VI*, vol. 7242, no. 1. SPIE, 2009, p. 72420L.

[71] S. Hocevar and G. Niger, "Reinstating Floyd-Steinberg: Improved Metrics for Quality Assessment of Error Diffusion Algorithms," in *Lecture Notes in Computer Science*, vol. 5099. Springer, 2008, pp. 38–45.

[72] F. Cittadini, M. Remita, J. Pervillé, S. Berche, M. Chouikha, H. Brettel, and G. Alquié, "Contribution to quality assessment of digital halftoning algo-

rithms," in *Color Imaging XII: Processing, Hardcopy, and Applications*, R. Eschbach and G. G. Marcu, Eds., vol. 6493, no. 1. SPIE, 2007, p. 64931D.

[73] X. Wan, D. Xie, and J. Xu, "Quality evaluation of the halftone by halftoning algorithm-based methods and adaptive method," in *Image Quality and System Performance IV*, vol. 6494, no. 1. SPIE, 2007, p. 64940U.

[74] T. chiun Chang and J. Allebach, "A new framework for characterization of halftone textures," *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1285–1299, May 2006.

[75] H. Fang and X. Lu, "Characterization of error diffusion halftone images based on subband decompostion," in *IEEE Workshop on Multimedia Signal Processing*, Nov. 2005, pp. 1–4.

[76] P.-E. Axelson, "Quality measures of halftoned images (a review)," Master's thesis, Department of Science and Technology, Link$\acute{}$oping University, Sweden, 2003.

[77] D. L. Lau, R. Ulichney, and G. R. Arce, "Blue- and green-noise halftoning models," *IEEE Signal Processing Magazine*, vol. 20, no. 4, pp. 28–38, July 2003.

[78] F. Nilsson, "Objective quality measures for halftoned images," *Journal of Optical Society of America*, vol. 16, no. 9, pp. 2151–2162, 1999.

[79] Q. Lin, "Halftone image quality analysis based on a human vision model," in *Human Vision, Visual Processing, and Digital Display IV*, vol. 1913, no. 1. SPIE, 1993, pp. 378–389.

[80] T. Mitsa, "Evaluation of halftone techniques using psychovisual testing and quantitative quality measures," in *Human Vision, Visual Processing, and Digital Display III*, vol. 1666, no. 1. SPIE, 1992, pp. 177–187.

[81] H. Rehman and B. L. Evans, "Flicker assessment of low-to-medium frame-rate binary video halftones," in *Proc. IEEE Southwest Symposium on Image Analysis and Interpretation*, May 2010.

[82] ——, "A framework for the assessment of temporal artifacts in medium frame-rate binary video halftones," *EURASIP Journal on Image and Video Processing*, vol. 2010, Oct. 2010.

[83] Z. Fan and F. Li, "Edge behavior of error diffusion," in *IEEE International Conference on Image Processing*, vol. 3, 1995.

[84] F. Campbell, "The human eye as an optical filter," in *Proc. IEEE*, vol. 56, June 1968, pp. 1009–1014.

[85] C. Hsu, C. Lu, and S. Pei, "Compression of halftone video for electronic paper," in *IEEE International Conference on Image Processing*, 2008, pp. 1600–1603.

[86] W. J. Tam, L. B. Stelmach, L. Wang, D. Lauzon, and P. Gray, "Visual masking at video scene cuts," in *Proc. SPIE Conf. on Human Vision, Visual Processing, and Digital Display*, vol. 2411, no. 1, 1995, pp. 111–119.

[87] B. Breitmeyer and H. Ogmen, "Recent models and findings in visual backward masking: A comparison, review, and update," *Perception and Psychophysics*, vol. 62, no. 8, pp. 1572–1595, 2000.

[88] R. P. I. Center for Image Processing Research. [Online]. Available: http://www.cipr.rpi.edu/resource/sequences/index.html

[89] V. Monga, N. Damera-Venkata, H. Rehman, and B. L. Evans. Halftoning toolbox for matlab. [Online]. Available: http://users.ece.utexas.edu/~bevans/projects/halftoning/index.html

[90] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "Study of subjective and objective quality assessment of video," *in press, IEEE Transactions on Image Processing*, 2010.

[91] K. Seshadrinathan, R. Soundararajan, A. Bovik, and L. Cormack, "A Subjective Study to Evaluate Video Quality Assessment Algorithms," *to appear, SPIE Proceedings Human Vision and Electronic Imaging*, 2010.

[92] [Online]. Available: http://live.ece.utexas.edu/research/quality/live_video.html

[93] B. Girod, "The information theoretical significance of spatial and temporal masking in video signals," in *Proc. Conf. on SPIE: Human Vision, Visual processing, and Digital Display*, vol. 1077, 1989, pp. 178–187.

[94] A. Seyler and Z. Budrikis, "Detail perception after scene changes in television image presentations," *IEEE Transactions on Information Theory*, vol. 11, no. 1, pp. 31–43, 1965.

[95] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[96] J. Jarvis, C. Judice, and W. Ninke, "A survey of techniques for the display of continuous tone pictures on bilevel displays," *Computer Graphics and Image Processing*, vol. 5, no. 1, pp. 13–40, 1976.

[97] P. Stucki, "MECCA-a multiple-error correcting computation algorithm for bilevel image hardcopy reproduction," *Research Report RZ1060, IBM Research Laboratory, Zurich, Switzerland*, 1981.

[98] J. Shiau and Z. Fan, "Set of easily implementable coefficients in error diffusion with reduced worm artifacts," in *Proceedings of SPIE*, vol. 2658, 1996, p. 222.

[99] J. Guo and J. Chen, "High Efficiency Digital Halftoning with Two-Element Error Kernel," in *IEEE International Conference on Image Processing*, 2006, pp. 1501–1504.

[100] J. Pingshan Li Allebach, "Tone-dependent error diffusion," *IEEE Transactions on Image Processing*, vol. 13, no. 2, 2004.

[101] T. Chang and J. Allebach, "Memory efficient error diffusion," *IEEE Transactions on image processing*, vol. 12, no. 11, pp. 1352–1366, 2003.

[102] R. Eschbach, "Reduction of artifacts in error diffusion by means of input-dependent weights," *Journal of Electronic Imaging*, vol. 2, pp. 352–358, 1993.

[103] V. Monga, N. Damera-Venkata, and B. L. Evans, "Design of tone-dependent color-error diffusion halftoning systems," *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 198–211, 2007.

[104] R. Eschbach and K. Knox, "Error-diffusion algorithm with edge enhancement," *Journal of the Optical Society of America*, vol. 8, no. 12, pp. 1844–1850, 1991.

[105] K. T. Knox, "Threshold modulation in error diffusion on nonstandard rasters," *Human Vision, Visual Processing, and Digital Display V*, vol. 2179, no. 1, pp. 159–169, 1994.

[106] N. Damera-Venkata and B. L. Evans, "Adaptive threshold modulation for error diffusion halftoning," *IEEE Transactions on Image Processing*, vol. 10, no. 1, pp. 104–116, JANUARY 2001.

[107] B. Zhou and X. Fang, "Improving mid-tone quality of variable-coefficient error diffusion using threshold modulation," *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 437–444, 2003.

[108] T.-C. Chang and J. P. Allebach, "Quantization of accumulated diffused errors in error diffusion," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 1960–1976, December 2005.

[109] I. H.Witten and R. M. Neal, "Using peano curves for bilevel display of continuous-tone images," *IEEE Comput. Graph. Applicat.*, vol. 2, pp. 47–52, May 1982.

[110] D. E. Knuth, "Digital halftones by dot diffusion," *ACM Trans. Graph.*, vol. 6, no. 4, pp. 245–273, Oct. 1987.

[111] A. K. Moorthy and A. C. Bovik, "A motion compensated approach to video quality assessment," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, 2009, pp. 872 –875.

[112] C.-J. Lian, S.-Y. Chien, C. ping Lin, P.-C. Tseng, and L.-G. Chen, "Power-aware multimedia: Concepts and design perspectives," *IEEE Circuits and Systems Magazine*, vol. 7, no. 2, pp. 26 –34, 2007.

[113] K. McGoldrick, "Mobile friendly rollable displays," in *Proceedings of the 32nd European Solid-State Circuits Conference (ESSCIRC)*, sept. 2006, pp. 1 –2.

[114] I. Choi, H. Shim, and N. Chang, "Low-power color tft lcd display for hand-held embedded systems," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED)*, August 2002, pp. 112–117.

[115] T. Srivastava and A. Narayan, "Dynamic fuzzy controller based multilayered architecture for extended battery life in mobile handheld devices," in *IEEE*

International Conference on Systems, Man and Cybernetics, 11-14 2009, pp. 3035 –3040.

[116] J. Kimmel, J. Hautanen, and T. Levola, "Display technologies for portable communication devices," *Proceedings of the IEEE*, vol. 90, no. 4, pp. 581–590, 2002.

[117] S. Pasricha, S. Mohapatra, M. Luthra, N. Dutt, and N. Venkatasubramanian, "Reducing backlight power consumption for streaming video applications on mobile handheld devices," in *Proc. First Workshop Embedded Systems for Real-Time Multimedia*, 2003, pp. 11–17.

[118] H. Shim, N. Chang, and M. Pedram, "A compressed frame buffer to reduce display power consumption in mobile systems," in *Proceedings of the Asia and South Pacific Design Automation Conference*. IEEE Press, 2004, pp. 818–823.

[119] H. Shim, Y. Cho, and N. Chang, "Frame buffer compression using a limited-size code book for low-power display systems," in *Workshop on Embedded Systems for Real-Time Multimedia*, 2005, pp. 7–12.

[120] K. Patel, E. Macii, and M. Poncino, "Frame buffer energy optimization by pixel prediction," in *IEEE International Conference on Computer Design: VLSI in Computers and Processors*, 2-5 2005, pp. 98 – 101.

[121] H. Cao, Y. Liu, L. Xu, Z. Deng, B. Wang, and T. Xue, "A bluetooth-based display system using bi-stable technology for rural people," in *Mobility '09:*

Proceedings of the 6th International Conference on Mobile Technology, Applications, and Systems. New York, NY, USA: ACM, 2009, pp. 1–4.

[122] H. Zang, J. Hwang, H. Gu, J. Hou, X. Weng, Y. Chen, and R. Liang, "Threshold and grayscale stability of Microcup electronic paper," in *Proceedings of SPIE*, vol. 5289, 2004, p. 102.

[123] B. Fitzhenry-Ritz, "Optical properties of electrophoretic image displays," *IEEE Transactions on Electron Devices*, vol. 28, no. 6, pp. 726–735, 1981.

[124] A. Dalisa, "Electrophoretic display technology," *IEEE Transactions on Electron Devices*, vol. 24, no. 7, 1977.

[125] S. Inoue, H. Kawai, S. Kanbe, T. Saeki, and T. Shimoda, "High-resolution microencapsulated electrophoretic display (epd) driven by poly-si tfts with four-level grayscale," *IEEE Transactions on Electron Devices*, vol. 49, no. 9, pp. 1532 – 1539, sep 2002.

[126] S. Inoue, K. Sadao, T. Ozawa, Y. Kobashi, H. Kawai, T. Kitagawa, and T. Shimoda, "Low temperature poly-si tft-electrophoretic displays (tft-epds) with four level gray scale," in *International Electron Devices Meeting, IEDM Technical Digest*, 2000, pp. 197 –200.

[127] A. Khan, T. Schneider, N. Miller, D. Marhefka, T. Ernst, F. Nicholson, and J. Doane, "Recent advances and product enhancements in reflective cholesteric displays," in *Proceedings of SPIE*, vol. 5741, 2005, p. 1.

[128] H. Rehman, M. Mignard, C. Chui, and J. Gille, "An efficient method to reduce temporal artifacts in video halftones," *submitted June 2010, EURASIP Journal on Image and Video Processing.*