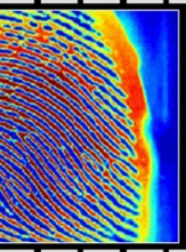
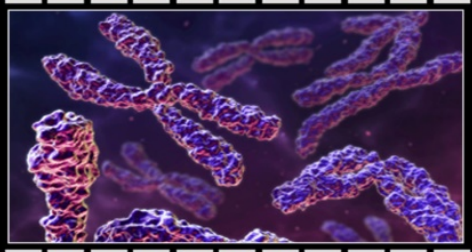
Volume 2, Issue 2
April 2014

AIVP

ADVANCES IN IMAGE AND VIDEO PROCESSING



Society for Science and Education
United Kingdom

ISSN: 2054-7412

TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD

DISCLAIMER

All the contributions are published in good faith and intentions to promote and encourage research activities around the globe. The contributions are property of their respective authors/owners and the journal is not responsible for any content that hurts someone's views or feelings etc.

AIVP ADVANCES IN IMAGE AND VIDEO PROCESSING
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 2
ISSN: 2054 - 7412

RGB Image Reconstruction Using Two-Separated Band Reject Filters

Muthana H. Hamd
Computer/ Faculty of Engineering, Al Mustansirya University
Baghdad, Iraq

ABSTRACT

Noises like impulse or Gaussian noise could be easily removed and recovered in the spatial domain by applying mean or median convolution. Structural noise, like periodic has a global degradation effect. This degradation is looking like bars that cover the image. This paper developed a new approach that detect the degradation; recover it; and hence reconstruct the original color image. The two dimensions discrete Fourier Transform is applied to isolate, block, and replace some particular frequencies (degradation bands) using auto detection and recovery procedure, it applies two separated band reject filters to avoid blocking the true bands like what is resulted in Ring filter. It tries locating the periodic noise (two spikes) and the four degraded lines using a powerful searching algorithm in the frequency domain. Unlike, Ring and Notch filter, the Two-Circle method is smart enough to predict the size of the spike, so a suitable band rejected filter should be selected to replace only degraded bands with approximation values instead of zeroes. The Peak Signal to Noise Ratio (PSNR) vs. periodic frequency relation is applied to find the quality of measurements of the reconstructed and cleaned colored image. So, the quality of three methods is calculated and compared. All testing results show that the two separated band reject filters method satisfied maximum dB results and minimum frequency change to the steady state values than other two methods.

Keywords: *Periodic Noise, BRF, Ring Filter, Notch Filter, DFT*

1. INTRODUCTION

Noise can be defined as any degradation or error in the image pixel value, which is caused by an external disturbance. When this disturbance becomes periodically rather than random disturbance, a corruption in image signal is obtained as bars over the image. The electromagnetic field of electric motor is an example of repeating nature that causes degradation over transmitting image. The two dimension discrete Fourier transform (2D DFT) is applied to process and decrease the degradation effect using the new transformed image features. The relatively high change of image frequency in spatial or data domain would be transformed into two spikes in image frequency domain. The location of two spikes reflects the

DOI: 10.14738/aivp.22.151
Publication Date:4th April 2014
URL: http://dx.doi.org/10.14738/aivp.22.151

range of periodic noise frequency (bars) in spatial domain. Two approaches were used to remove or decrease this degradation, first one is the Band Reject Filter (BRF) or Ring filter and the second one is the Notch filter. This paper develops two auto detection and recovery procedures that acts as two separated BRFs with a combined Notch filter [1,2,3].

They try removing and replacing only corrupted bands. The two spikes and their associated four lines are recovered by an approximation value other than zeroes. A comparison is made between these methods by finding the image qualification using Peak Signal to Noise Ratio (PSNR). This paper is organized as follows: Section 1 is an introduction, section 2, explains the main types of structured noise; section 3 illustrates the auto detection and recovery procedures that are used to detect and replace the unwanted bands with approximation values. The comparison between these methods is presented in section 4. The last section presents the conclusion.

2. STRUCTURED NOISE

The structured noise is classified into stationary with a fixed amplitude, frequency, and phase or non-stationary, where the noise parameters are varying across the image. When this image signal is subjected to a periodic, rather than a random disturbance, image will be corrupted by periodic noise [4,5]. So periodic noise may occur in the image because of:

- Image equipment
- Network equipment
- External disturbance of repeating nature

The general trigonometric formula that causes bars over image is [5].

$$\text{Noise (periodic)} = \text{sine}\,(\omega_1 x + \omega_2 y) + 1 \qquad (1)$$

Where:

x, y: are image plane pixel values

$\omega_{1,2}$: Periodic noise frequencies

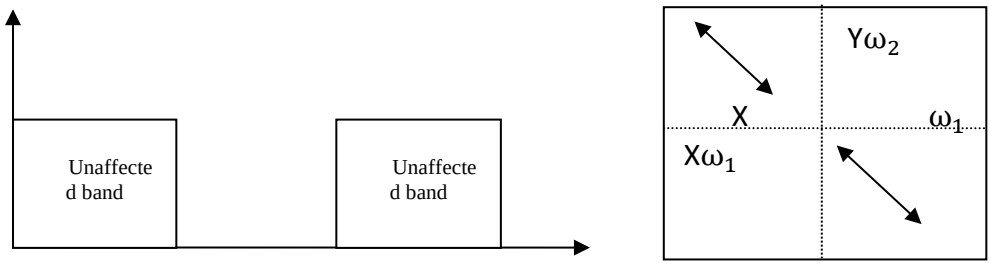
ω_1 is responsible for horizontal shift of spikes along x-axis, and ω_2 is responsible for vertical movement along y-axis. For (256 × 256) image, the movement scale is approximately from 0.1 which is closed to the image center (DC coefficients) towards outside at $\omega_{1,2} \approx 2.9$. If this degradation in equation 1 is injected in the RGB image planes, the process could be implemented for reach plane individually, by applying equation 2.

$$\text{RGB}_{\text{noisy}} = \text{Periodic noise} + \text{RGB}(r, g, b) \qquad (2)$$

3. FREQUENCY DOMAIN NOISE REMOVAL

Image transformation from spatial or data domain to the frequency domain could be implemented using the Two Dimension Discrete Fourier Transform (2D-DFT). Filtering or convolution in spatial domain is converted to the whole image and mask multiplication in

frequency domain. Transforms to the new domain can obtain many new features like: conjugate symmetry, mirror, DC coefficients…etc [6,7, 8]. The periodic noise frequency or bars would be represented as two shiny spikes in frequency domain. The two shiny spikes have maximum frequency complex value. If the spikes is detected and located, their associated two degraded lines will be detected and located also. If the frequency of periodic noise is low, the spikes would be closed to the DC coefficient (image center) and vice versa; they move out diagonally toward image plane edges when frequency of periodic noise is increased, see Fig. 1 and 2. For a given noise frequency ω_1 and ω_2, the searching procedure is started looking for the two maximum values (spikes) diagonally, if founds a recovery procedures is beginning Ring and Notch filters usually replace the degraded bands with zeros, while auto detection and recovery procedure try approximating the degraded bands with a mean band value for spike's neighbour, as follows;



(a) (b)

Figure 1: (a) spikes band reject filter, (b) periodic noise frequency movement

4. AUTO DETECTION AND RECOVERY PROCEDURE

The auto detection procedure starts finding the shiny spike which holds a maximum complex value in frequency band of quarter one. Once, it is located, the next step is to assign values instead of zeros. The new idea in this approach is the approximation value assignment. Instead of zeros substitution as in Ring and Notch filter, the Auto detection approach finds an approximation value as follows:

- $app_s = \frac{spike\ value(max)}{\sqrt{spike\ value}}$ (4)
- Associated spike lines

$$app_{rw} = \frac{\sum_{i=1}^{M} ROW_i}{M} \quad (5)$$

$$app_{cl} = \frac{\sum_{i=1}^{N} COL_i}{N} \quad (6)$$

Where,

app_s : is the approximation value

app_{rw} : is the approximation value of row vector

app_{cl} : is the approximation value of column vector

5. QUALITY MEASURMENT AND COMPARISION

Four (256 × 256) colored image are considered and tested by three periodic noise removal methods. First, images are injected with variant frequencies of periodic noise, ranging from $\omega_{1,2}$ = 0.125 to $\omega_{1,2}$ = 2.9. Then auto detection and recovery procedures results are tabulated in table 1, 2, and 3. These results belong to the Green.jpg image and it finds a relation between periodic noise frequency ($\omega_{1,2}$) and image quality, (PSNR). The Two-Circle BRF results (PSNR vs. $\omega_{1,2}$) for all three tables are better and more stable than other two detection methods. That's because of smartness of the Two-Circle procedure where, it detects the size of spikes for each periodic noise value and a suitable band reject filter is chosen to block the degraded bands then change it with an approximation value as computed in equation 4. The PSNR qualifications for all three methods are represented in figure 3. The Two-Circle BRF method satisfied rapid change and reached to the maximum dB and steady state value with $\omega_{1,2} = 0.3$, i.e.; after 4 change of $\omega_{1,2}$. The Two-circle method is very suitable to remove small or large value of degradation, where, the area of the Two-circle BRFs is changed according the spike location. The closest to the DC coefficients, the smallest area of the Two-Circle filters, see Figure 2. Ring filter curve starts with 10.29 dB but it reaches to the maximum PSNR at $\omega_{1,2}$= 1.0 as shown in figure 2, it means that Ring method satisfy the steady state (10.42 dB) after 8 changes of $\omega_{1,2}$. Like Ring filter, Notch filter starts with initial value (10.20 dB) and it reaches to the steady state at $\omega_{1,2} = 0.75$, i.e.; after 7 change of $\omega_{1,2}$. The four reconstructed and cleaned images are figured in Fig.4

Table 1. Ring Method

	Ring Method, PSNR (dB)											
$\omega_{1,2}$	0.125	0.15	0.175	0.2	0.3	0.4	0.5	0.75	1.0	1.75	2.5	2.9
Lenna	7.00	7.01	7.02	7.02	7.03	7.03	7.03	7.04	7.04	7.04	7.04	7.04
Football	10.98	10.98	10.99	10.99	10.99	10.99	10.99	11.00	11.00	11.00	11.00	11.00
Greens	10.29	10.33	10.34	10.36	10.36	10.37	10.38	10.41	10.42	10.42	10.42	10.42
Peppers	8.65	8.67	8.68	8.69	8.69	8.70	8.70	8.71	8.71	8.71	8.71	8.71

Table 2. Notch Method

	Notch Method, PSNR (dB)											
$\omega_{1,2}$	0.125	0.15	0.175	0.2	0.3	0.4	0.5	0.75	1.0	1.75	2.5	2.9
Lenna	6.98	6.99	7.00	7.01	7.03	7.03	7.04	7.04	7.04	7.04	7.04	7.04
Football	10.95	10.97	10.97	10.98	10.99	11.00	11.00	11.00	11.00	11.00	11.00	11.00
Greens	10.20	10.25	10.30	10.32	10.38	10.40	10.41	10.42	10.42	10.42	10.42	10.42
Peppers	8.60	8.64	8.66	8.67	8.70	8.70	8.71	8.71	8.71	8.71	8.71	8.71

Table 3. Two-Circle Method

	Two Circle Method, PSNR (dB)											
$\omega_{1,2}$	0.125	0.15	0.175	0.2	0.3	0.4	0.5	0.75	1.0	1.75	2.5	2.9
Lenna	7.02	7.03	7.03	7.03	7.04	7.04	7.04	7.04	7.04	7.04	7.04	7.04
football	10.98	10.99	10.99	11.00	11.00	11.00	11.00	11.00	11.00	11.00	11.00	11.00
Greens	10.36	10.39	10.40	10.41	10.42	10.42	10.42	10.42	10.42	10.42	10.42	10.42
Peppers	8.69	8.70	8.70	8.70	8.71	8.71	8.71	8.71	8.71	8.71	8.71	8.71

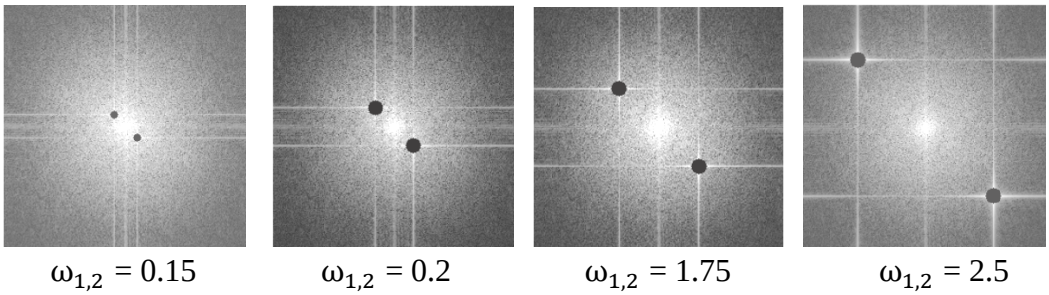


Figure 2. Auto detection and periodic noise removal procedure

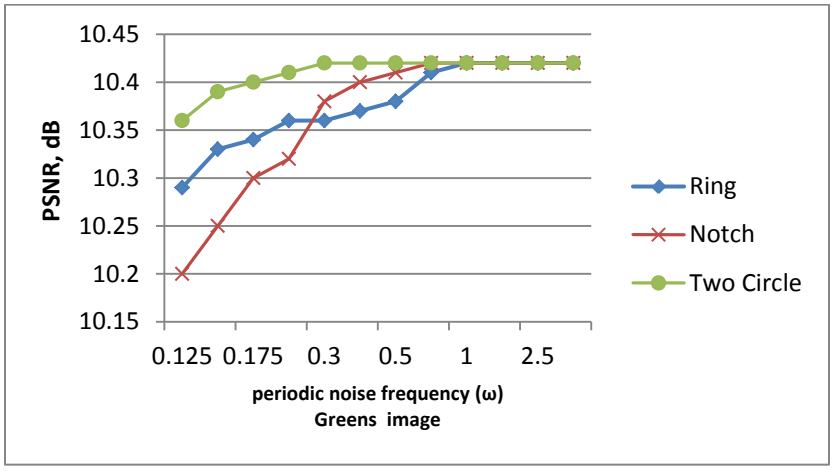


Figure 3. PSNR curves for three filter types

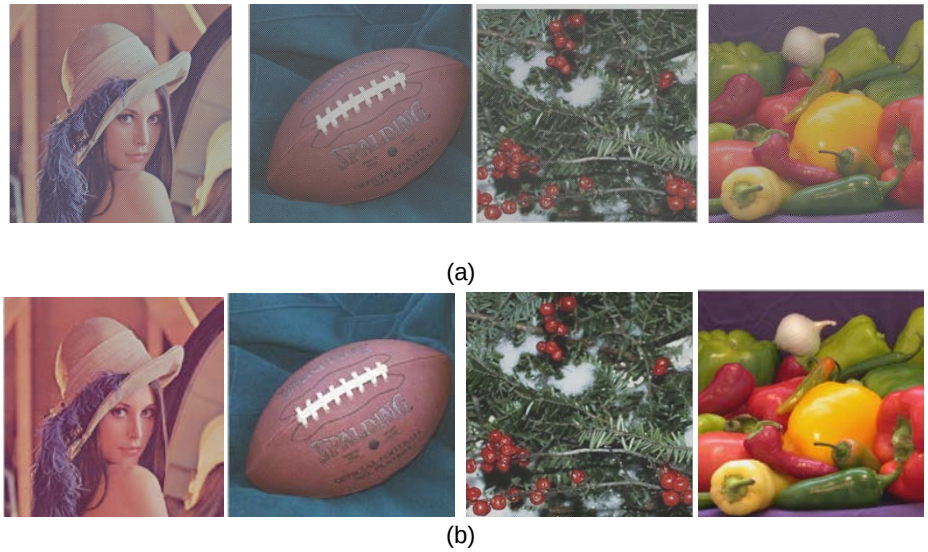


Figure 4. (a) noisy images, (b) reconstructed and cleaned images

6. CONCLUSION

Three auto detection and periodic noise removal methods have been developed and implemented to remove or decrease the degradations that could be resulted from an external disturbance like periodic noise. The two separated band reject filters are designed to block and replace the degradation frequency bands by approximation values after transforming noisy image into the 2D discrete Fourier transform. The two common methods: the Ring and Notch filters are modified to be auto detected and recovered procedure. All methods have been implemented and tested on four colored image of 256×256 size as tabulated in table 1, 2 and 3. The maximum PSNR dB values are satisfied with Two-Circle method. Also, it satisfied minimum frequency change to the steady and stable state of PSNR.

REFERENCES

[1]. K. H. Talukder and K. Harada, "Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image", IAENG International Journal of Applied Mathematics, 36-1, IJAM_36_1_9, (2007).

[2]. M. Saraswat , A. K. Wadhwani and M. Dubey, "Compression of Breast Cancer Images by Principal Component Analysis", International journal of Advanced Biological and Biomedical Research, 1, 767-776, (2013).

[3]. J. S. Walker. Wavelet-based Image Compression (2nd Edition), University of Wisconsin, Eau Claire (1999).

[4]. S.S.Palewar and Ranjana Shende, "Watermarking Robustness Evaluation Using Enhanced Performance Metrics.", International Journal of Engineering Research & Technology (IJERT), 2, 2278-0181, (2013).

[5]. N. D.Venkata, T. D. Kite, B. L. Evans, and A. C. Bovik, "Image Quality Assessment Based on a Degradation Model", IEEE Trans. Image Proc. 9, 636-650, (2000).

[6]. Sungkwang Mun and J. E. Fowler, "BLOCK COMPRESSED SENSING OF IMAGES USING DIRECTIONAL TRANSFORMS", International Conference on Image Processing, Cairo, Egypt, 3021-3024, (2009).

[7]. Yusra A. Y. Al-Najjar and Der Chen Soong, "Comparison of Image Quality Assessment: PSNR, HVS, SSIM, UIQI", International Journal of Scientific & Engineering Research, 3, 8, 2229-5518, (2012).

[8]. A. Saffor and Abdul Rahman Ramli, "A COMPARATIVE STUDY OF IMAGE COMPRESSION BETWEEN JPEG AND WAVELET", Malaysian Journal of Computer Science, 14, 1, 39-45, (2001).

[9]. N. Salamati and Z. Sadeghipoor, "Compression of Multispectral Images: Color (RGB) plus Near-Infrared (NIR)", (2011).

[10]. N. Dey, A. B. Roy, and S. Dey, "A Novel Approach of Color Image Hiding using RGB Color planes and DWT", International Journal of Computer Applications, 36, 5, 0975 – 8887, (2011).

AIVP ADVANCES IN IMAGE AND VIDEO PROCESSING
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 2
ISSN: 2054 - 7412

Novel Evaluation of Digital Halftone Image Qualities by Psychological Analysis

Tadahiko Kimoto[1], Chieko Kato[2]
[1]*Faculty of Science and Engineering, Toyo University, Kawagoe, Japan;*
[2]*Faculty of Information Sciences and Arts,Toyo University, Kawagoe, Japan;*
[1]kimoto@toyo.jp, [2]kato-c@toyo.jp

ABSTRACT

Monochrome binary halftone images express multiple levels with halftones produced by distributing black and white pixels. In this paper, for the subjective evaluation of halftone qualities, binary halftone images are considered as pointillistic images of visible black-and-white pixels. Visual impressions given by the pixel distribution are measured by the semantic differential (SD) method in a psychological manner.

An experiment for measuring subjective impressions of subjects by using selected 30 SD items was conducted on several binary halftone images that were produced by three kinds of the typical halftoning methods: two error-diffusion methods and a kind of random dither method. The sample data are analyzed statistically by factor analysis. As a result, two common factors are found out and interpreted as the *sensitivity* factor and the *activity* factor, respectively. Also, the result of variance analysis demonstrates that the two common factors are able to describe the difference among the three halftoning methods.

Keywords: Digital halftone, Visual impression, Semantic differential method, Factor analysis, Variance analysis.

1.INTRODUCTION

Digital halftoning is one of data compression techniques for displaying digital images. It reduces pixel bits while compensating the missing signal levels with visually pseudo-continuous levels, called halftones [1]. Halftoning to convert multi-level images to bi-level ones is important particularly in printing today. There are a variety of halftoning methods for the binarization [2].

In generating a monochrome binary halftone image, halftoning is achieved by distributing black and white pixels. As a result, the resulting image looks like a picture made in a *pointillistic* manner, which is a style of painting for making up pictures by using small colored dots. Appearances of the halftone image, which are essentially subjective qualities, are generally

DOI: 10.14738/aivp.22.151
Publication Date:4th April 2014
URL: http://dx.doi.org/10.14738/aivp.22.151

assessed by observing it at an image resolution such that each pixel is hardly distinguishable. However, it is just the pixel distribution that characterizes the halftones.

In this paper, we consider binary halftone images as the pointillistic images composed of visible black and white pixels rather than the images approximated with halftone levels. The purpose is to characterize halftoning methods from the viewpoint of visual impressions given by the black-and-white distribution. To measure the subjective impressions, we have carried out an experiment using a semantic differential (SD) method, which is often used for psychological analysis [3]. The sample data are investigated by statistical analysis to search for significant factors that can describe the halftone qualities.

2. VARIETY OF BINARY HALFTONES

2.1 Principle of halftoning in binary

The principle of converting a multi-level image to a bi-level or binary halftone image is expressed below: Let $f(m, n)$ denote a pixel value at the coordinates of (m, n) in an image f. Also, source images are assumed to have 8-bit, that is, 256 levels from 0 (black) to 255 (white). Converting a source image f to a binary image f' is generally expressed as

$$f'(m,n)=\begin{cases} 1, & f(m,n)+a(m,n)\geq 128 \\ 0, & f(m,n)+a(m,n)<128 \end{cases} \quad (1)$$

where $a(m, n)$ is a signal that determines a threshold level for $f(m, n)$ as a result. Varying $a(m, n)$ for every pixel produces the variation of $f'(m, n)$. Accordingly, the properties of the quantization error $e(m, n)$, which is defined by

$$e(m,n)=255f'(m,n)-f(m,n), \quad (2)$$

depend on the properties of $a(m, n)$. Different properties of the quantization error yield different distribution of black and white pixels, and accordingly, different appearances in pointillism.

2.2 Halftoning methods being focused

In this paper, we consider the following three typical ones of the conventional halftoning methods.

2.2.1 Error diffusion halftoning

The principle of error diffusion halftoning is to diffuse the quantization errors by transforming them to high-frequency signals that could be hardly perceptible to human eyes. The transformation is implemented with a digital filter, which gives the values of $a(m, n)$ of Equation (1). Thus, the frequency response characteristics of the filter determine the distribution of binary pixels and consequently, affect the halftone qualities.

Various digital filters for error-diffusion halftoning have been known that are capable of producing halftones of visual quality. We have chosen two such filters in this paper: the filter proposed by Floyd and Steinberg [4] and that by Jarvis et al. [5]. We refer to the error-diffusion methods with these two filters as EDF-FS and EDF-JJN, respectively. According to the filter characteristics, EDF-FS diffuses the quantization errors in the high-frequency range, and EDF-JJN does in the middle-frequency range. This reflects the difference in halftone quality for the two methods.

2.2.2 Improved grayscale quantization

Improved grayscale (IGS) quantization [6] performs halftoning like a random dither method. In halftoning an 8-bit source image by IGS, pixel values are accumulated through a scanning path in the image until at a pixel, say, the coordinates (m, n), and the seven low-order bits of the result are used as $a(m, n)$. Consequently, $a(m, n)$ behaves as a random variable with local characteristics of the image.

In the experiment below, IGS is carried out with a level transformation for preserving the average level of a source image through halftoning in terms of 8-bit levels [7]. Also, a Hilbert scanning path is used to implement IGS in a source image so that artifacts such as geometric patterns can be prevented from occurring in the resulting image. We refer to the whole scheme as HT-IGS below.

3. EXPERIMENTAL METHOD

3.1 Preparing test images

We use monochrome images of 8-bit brightness levels from 0 to 255 as the original images, and convert them to binary halftone images for the experiment. The procedure of the conversion is composed of the following three processes:

(1) Each original image is modified by the histogram equalization transformation [8] so that all source images to which the halftoning methods are applied can have both the same contrast and the same average level. The resulting image, f, has pixel levels ranging in the whole of 8-bit levels, and all the levels occur equally often.

(2) A binary image f' is generated from f by each method of EDF-FS, EDF-JJN and HT-IGS.

(3) The pixel values 0 and 1 of f' are transformed to the darkest level 0 and the brightest level 255, respectively.

Thus, the source images are normalized in terms of overall contrast and averaged brightness level. Hence, the halftone patterns rather than the halftone levels are expected to mainly affect impressions of the test images.

3.2 Displaying test images

We decide a method for presenting the test images to subjects by specifying image resolution and viewing conditions. The image resolution is related both to image size on a monitor screen and to a distance between the monitor and a viewer. The viewing conditions are involved in implementing the SD method.

3.2.1 Image resolution

We use a 17-inch, 24-bit-color TFT LCD monitor of 1,280 x 1,024 dots to display the test images with. The region of 512 x 400 pixels of a test image is shown in the middle of the screen by using two by two dots for every pixel. Thus, the displayed image has the size of about 270 mm in width and about 211 mm in height, in other words, the resolution of about 48 pixels per inch on the screen. A subject looks at the monitor about three times as long as the monitor's height away. The black and white pixels look generally distinguishable from the distance.

3.2.2 Viewing conditions

There are two conditions to consider for viewing the test images on a monitor screen.

(a) Displaying time; the duration of time a subject looks at a test image must be the same both for any subject and for any test image. It should be long enough for the subject to gain some visual impressions from the image but not so long as to make the subject tired of looking at it.

(b) Background color on the screen; a test image is displayed in the middle of the monitor screen. Hence, a color of the area surrounding the image, that is, the background influences the easiness in looking at the image.

To examine these conditions, we have conducted a preliminary experiment. The conditions have been determined from the result: The displaying time is set four seconds, and the background color is set black (See Appendix A).

3.3 Semantic differential method

An SD item is a pair of adjective words that are contradictory to each other, for example, *fine-coarse*. It should represent a visual impression of pictures in the present study. The SD items that seem to apply to the halftone images are determined in the following manner (See Appendix B for more details): First, a total of 75 items are chosen from among each of seven categories, which are generally used in psychological analysis. Then, we have carried out another preliminary experiment where subjects look at several halftone images and select those items that seem to describe visual impressions of the images. From the result, 30 of the most selected items have been determined for the use in this study. The items will be shown later (in Table 1).

As shown in Appendix B.2, for each of the selected items, a scale of seven gradations from 1 to 7 is defined to measure the impression that the item describes. For example, for the item

dark-bright, the gradation of 1 represents *dark*, and 7 represents *bright*, while 4 represents *not impressed either way* (See Figure A.7). All the items each expressed by the corresponding scale are listed in a questionnaire sheet to present to subjects.

3.4 Conducting the experiment

3.4.1 Test images used for SD method

Figure 1 shows five source images from which the test halftone images are generated. We have selected these images considering the following points:

(a) The images hardly include anything such that some ones like it personally and some ones dislike it, such as some kinds of vegetables.

(b) There are included a variety of contents in the image set: a living thing, a nonliving thing, a plant, a human, a close-range view, a long-distant view and so on.

A total of 15 binary halftone images have been produced from these original images by using the three halftoning methods. Examples of the halftone image produced by each method are shown in Figure 2. These images are to be used only for the SD method, which have never been used in the preliminary experiments described in Appendices.

3.4.2 Procedure of SD method

Thirty-nine subjects, who are all Japanese in their early twenties and unfamiliar with halftoning techniques, have participated in the experiment. The 15 test images are displayed on the monitor one by one in a random order to a subject. The SD method proceeds for every subject as follows:

(1) The subject is asked to gaze at the image while it is being displayed on the monitor screen.

(2) As soon as the image is erased from the screen, the subject is asked to evaluate each SD item with the corresponding scale.

(3) After evaluating all the SD items, the subject goes on to the next image by operating the experimental system himself/herself.

The experiment has been carried out for every subject in the same indoor environment of, for example, the lighting.

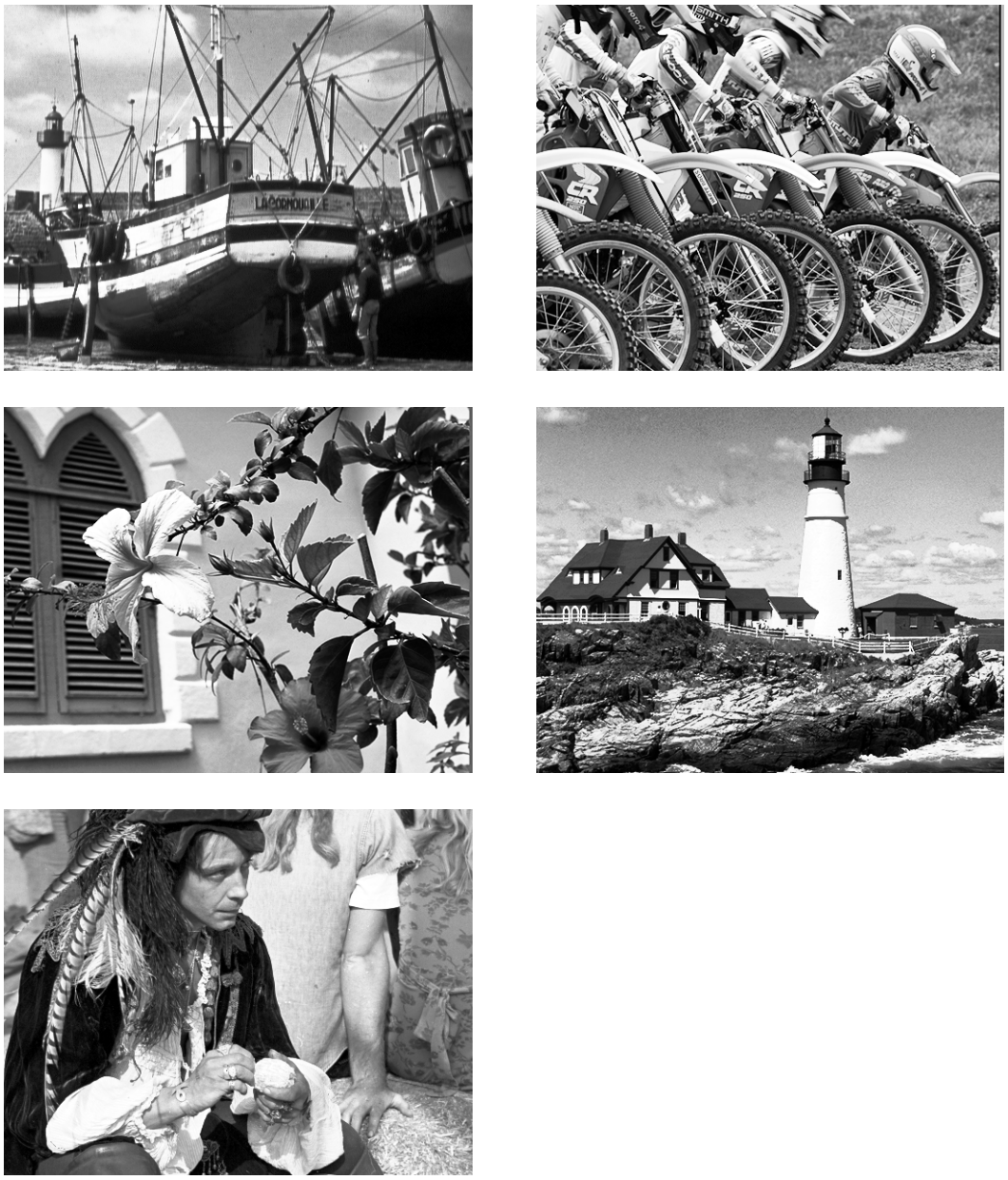


Figure 1: Monochrome 8-bit images used as source images to produce halftone images from, which are to be used in the main experiment of the SD method. Each figure shows the 512 x 400 region of the image that is actually displayed on the monitor while the whole image is of 512 x 512 pixels. These images have been processed in the manner described in Section 3.1. The original images are from the WEB sites [9-11].

(a) (b)

(c)

Figure 2: Examples of 512 x 400 halftone images, (a) by EDF-FS, (b) by EDF-JJN, and (c) by HT-IGS

4. ANALYSES OF EXPERIMENTAL RESULT

4.1 Factor analysis

4.1.1 Deriving common factors

Thirty-nine sample data of 30 SD-item values have been obtained from the experiment for each of five source images and for each of three methods. Considering each SD item as a variable that takes a value in 1, 2, ..., 7 of the item's scale, we carry out a factor analysis of the 30-variable data. In the analysis, the maximum likelihood method is used for the calculation of factors, and the axis rotation is carried out by the varimax rotation method. The analysis proceeds as follows.

First, all the 585 sample data are analyzed to derive common factors from the variables. The result has shown that there are five factors with the eigenvalue greater than one. Taking account of the contribution of the factors, we have set the number of common factors to be extracted to two, and then, carry out the factor analysis again to obtain them.

Table 1: Result of factor analysis

No.	SD item word of value 1-word of value 7	Factor loading	
		Factor 1	Factor 2
1	*unlikeable-likeable*	0.902	0.074
2	*bad-good*	0.897	0.135
3	*messy-tidy*	0.871	0.028
4	*unfriendly-friendly*	0.863	0.019
5	*ugly-beautiful*	0.844	-0.045
6	*low quality-high quality*	0.841	0.155
7	*dislike-like*	0.828	0.096
8	*unclear-clear*	0.792	0.237
9	*dirty-clean*	0.770	-0.178
10	*unfocused-easy to recognize*	0.768	0.264
11	*dull-vivid*	0.767	0.072
12	*unnatural-natural*	0.757	-0.024
13	*muddy-transparent*	0.753	-0.075
14	*ambiguous-distinctive*	0.692	0.304
15	*gloomy-shiny*	0.660	0.126
16	*delicate-harsh*	-0.650	0.258
17	*dark-bright*	0.645	-0.078
18	*old-new*	0.603	0.205
19	*fine-coarse*	-0.596	-0.029
20	*uniform-uneven*	-0.488	0.036
21	*cold-warm*	0.428	0.241
22	*weak-powerful*	0.263	0.707
23	*inconspicuous-conspicuous*	0.237	0.653
24	*light-rich*	0.039	0.638
25	*static-dynamic*	0.001	0.635
26	*quiet-lively*	0.047	0.572
27	*faint-deep*	-0.051	0.562
28	*showy-dull*	0.156	-0.472
Eigenvalue		11.8	3.12
Contribution ratio (%)		42.2	11.1
Cumulative contribution ratio (%)		42.2	53.3

As regards the factor loadings of SD variables for each common factor, we assume that factor loadings of an absolute magnitude greater than 0.4 are effective for the factor. According to the result of the second analysis, then, two SD items both of whose factor loadings are found ineffective have been omitted. Then, another factor analysis is carried out for the remaining 28 SD variables.

The result of the third analysis is shown in Table 1, which includes the factor loadings of the rotated values. In this table, the SD variables are divided into two groups: one that is composed of those variables whose loading of the primary common factor, referred to as Factor 1, is

larger than the loading of the secondary common factor, referred to as Factor 2, in absolute magnitude, and the other that is composed of those variables whose Factor 2-loading is larger than Factor 1-loading. The variables of the first group are listed in descending order of the absolute Factor 1-loadings, and then, those of the second group are listed in descending order of the absolute Factor 2-loadings. These two main common factors account for 53.3% of the total variance. Factor 1 is affected mainly by 21 variables and Factor 2 by 7 variables. Note that the SD items were actually written in Japanese in the experiment.

Figure 3 shows two dimensional plots of the pairs of factor loadings for 28 variables with the axes rotated. From this figure, we observe that the variables cluster together around the two axes that represent the respective main common factors.

4.1.2 Interpreting common factors

An interpretation of the meaning of each common factor extracted above is discussed below. As shown in Table 1, Factor 1 accounts for 42.2% of the total variance. Also, those SD items of a large factor loading are *unlikeable-likeable*, *bad-good*, *dirty-clean* and so on. These adjectives are considered to describe people's feelings and preferences. Accordingly, we designate this factor as a *sensitivity* factor.

Factor 2 explains 11.1% of the total variance. Those SD items of a large factor loading are *weak-powerful*, *light-rich*, *static-dynamic* and so on. These adjectives are considered to describe how much active, energetic or dynamic an image looks. Accordingly, we label this factor as *activity*.

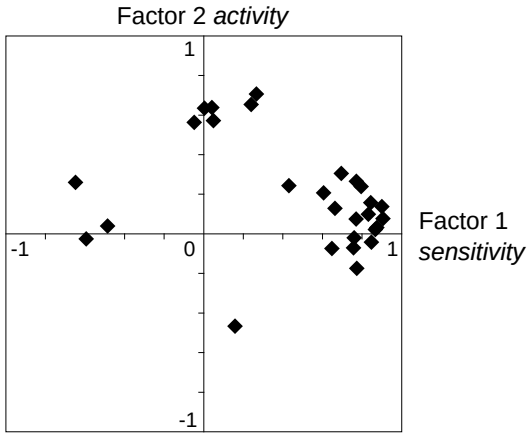


Figure 3: Two-dimensional plot of pairs of factor loadings after rotated

4.2 Variance analysis

By the above factor analysis, two factor scores of the respective common factors are calculated for each SD sample. Using the factor scores, we discuss the characterizing of the halftoning methods below.

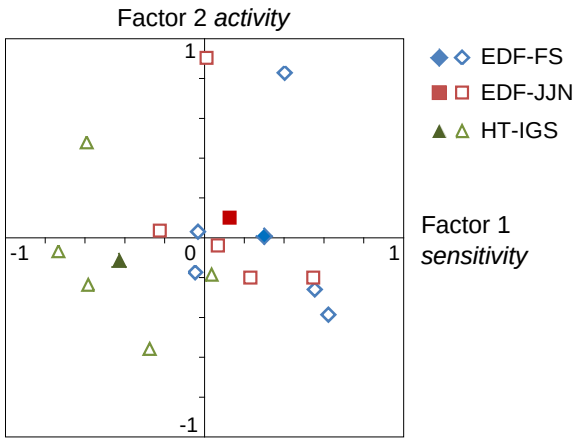


Figure 4: Two-dimensional plot of two factor scores of the respective common factors: An open marker shows the factor scores averaged for each of the five source images. A solid marker shows the average point of all the samples for each method.

First, for each method, we have carried out a variance analysis on the factor scores of SD samples for the five kinds of image. The result shows that both the factor scores of Factor 1 and those of Factor 2 differ among the five kinds of image at the significant level of 0.05. Figure 4 shows two-dimensional plots of the averaged factor score of Factor 1 and that of Factor 2 for each image.

Next, another variance analysis has been carried out on the factor scores of all the SD samples for the three methods. The averaged factor score is also obtained for each method as

depicted in preceding Figure 4. The result reveals that the factor scores of Factor 1 differ among the methods at the significant level of 0.05. Thus, Factor 1 can describe the difference in *sensitivity* impression among the three methods. The factor scores of Factor 2, on the contrary, show little difference among the methods.

5. CONCLUSION

By carrying out statistical analyses on the sample data obtained by the SD method, we have summarized visual impressions given by binary halftone images into two common factors, *sensitivity* and *activity*. The analyses have also demonstrated that EDF-FS method can produce those binary halftone images which give a stronger impression of *sensitivity* than other two methods, and HT-IGS method produces the least *sensitivity* in the black-and-white distribution. As for the *activity* impression, the three methods give little difference to the subjects.

The fact that one of the factors is able to differentiate among the halftoning methods validates this approach. Our future study is to investigate the impression factors in more detail so as to choose an appropriate halftoning method that can produce visual impressions as required.

REFERENCES

[1]. Ulichney, R., *Digital Halftoning*, MIT Press, 1993.

[2]. Venkata, N.D., Evans, B.L., and Monga, V., *Color error-diffusion halftoning*, IEEE Signal Pocessing Magazine, 2003, **20**(4): p. 51-58.

[3]. Heise, D.R., *Surveying Cultures: Discovering Shared Conceptions and Sentiments*, Wiley, 2010.

[4]. Floyd, R.W. and Steinberg, L., *An adaptive algorithm for spatial grayscale*, Proceeding of the Society for Information Display, **17**(2), 1976: p. 75-77.

[5]. Jarvis, J.F. and Judice, C.N. and Ninke, W.H., *A survey of techniques for the display of continuous-tone pictures on bilevel displays*, Computer Graphics and Image Processing, **5**(1), 1976: p. 13-40.

[6]. Gonzalez, R.C., Woods, R.E., *Digital Image Processing*, Addison-Wesley, 1993.

[7]. Kimoto, T., *Multi-level halftoning by IGS quantization*, Journal of Signal and Information Processing, **4**(4), 2013: p. 351-358.

[8]. Rosenfeld, A., Kak, A.C., *Digital Picture Processing*, Academic Press, 1982.

[9]. (Image files) http://r0k.us/graphics/kodak/

[10]. (Image files) http://sipi.usc.edu/database/

[11]. (Image files) http://homepages.cae.wisc.edu/~ece533/images/

Appendices

A. DETERMINING VIEWING CONDITIONS

A.1 Experimental method

Both for the purposes of presenting halftone images to subjects appropriately for the SD method and for the purpose of performing the presentation under the same viewing conditions, the following two variables are determined:

(a) A color of the background area surrounding a halftone image in the monitor screen
(b) A duration of time a halftone image is displayed to a viewer on the screen

The variables are evaluated by simulating the view in a preliminary experiment. The method of the experiment is described in detail below.

As regards the background color, monochrome color, that is, grayscale is here assumed to be proper because halftone images are to be displayed in black and white. Then, a halftone image is displayed in the background of an 8-bit brightness level *B* for just *T* seconds where *B* takes one of five values 0, 64, 128, 192 and 255 and *T* takes one of four values 2, 3, 4 and 5. Thereby, 20 different displays are presented to every subject in a random order. For each display, a subject is asked to evaluate both the whole screen view from the viewpoint of how much easy to see it is on the ratings listed in Table A.1, and the duration of time from the viewpoint of how much suitable it is for obtaining a certain impression from the image on the ratings listed in Table A.2.

Table A.1: Ratings used to evaluate the whole screen view

Value	Rating
5	Very easy to see
4	Easy to see
3	Ordinary
2	Hard to see
1	Very hard to see

Table A.2: Ratings used to evaluate the duration of displaying an image

Value	Rating
5	Too long
4	Long
3	Suitable
2	Short
1	Too short

Figure A.1 shows a halftone image of 256 x 256 pixels produced by EDF-JJN. We use it through this preliminary experiment and display it on a 12.1" color LCD monitor of 1024 x 768 dots by using a 2 x 2 dot area for every pixel. Thus, the displayed image is 123 mm square centered in the 246 mm x 185 mm monitor screen. The hardware settings of the monitor such as luminous intensity and contrast are fixed during the experiment. Nine subjects have taken

part in the experiment, who are all male Japanese in their early twenties and unfamiliar with digital halftoning techniques.

A.2 Analyses of experimental results

From the above experiment, we have obtained both nine scores regarding the screen view and nine scores regarding the displaying duration for each viewing condition (*B*, *T*), for the background level *B*=0, 64, 128, 192, 255 and the duration time *T*=2, 3, 4, 5. The analyses of these results evaluate the proper values of *B* and *T* below.

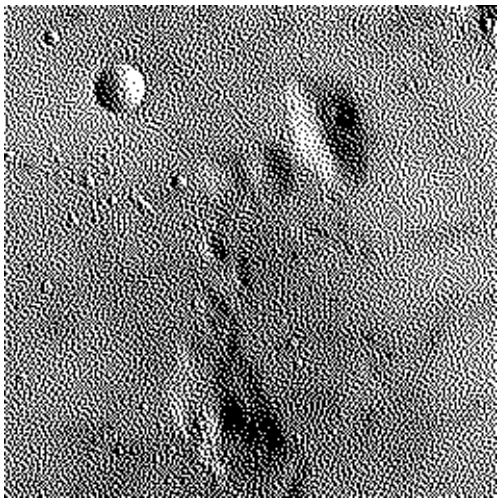


Figure A.1: The halftone image used for evaluating the viewing conditions in the preliminary experiment

A.2.1. Background level

A total of 36 scores on the ratings defined in Table A.1 have been collected for each of five values of the background level *B*. The frequency distributions of rating values in the scores are shown as a function of *B* in Figure A.2. The figure also includes the gravity center μ of the frequencies for each *B*, which is defined as a ratio of moment by

$$\mu = \frac{\sum_{v=1}^{5} v \cdot N(v)}{\sum_{v=1}^{5} N(v)} \qquad \text{(A.1)}$$

where $N(v)$ is the frequency of a rating value v for v=1, 2, ..., 5.

From Figure A.2, we observe that the total proportion of the rating values 3, 4 and 5 which correspond to the rating *ordinary*, *easy to see* and *very easy to see*, respectively, increases as B decreases to 0, and the gravity center increases accordingly. Consequently we select the background level B to be 0.

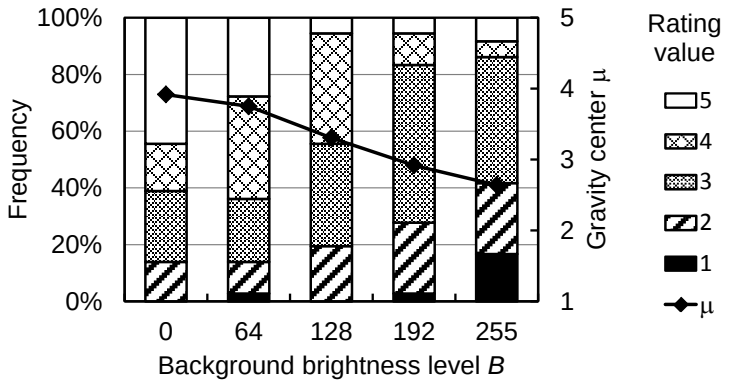


Figure A.2: Frequency distributions of the ratings on screen view for background brightness levels

A.2.2. Displaying duration

A total of 45 scores on the ratings defined in Table A.2 have been collected for each of four values of the displaying time T. The frequency distributions of rating values in the scores are shown for each T separately in Figure A.3. Using a calculation similar to that defined in Equation (A.1), we also obtain gravity centers of the scores. Figure A.4 shows them as a function of T.

From Figure A.4, it is obvious that the displaying time of four seconds, T=4, achieves the gravity center closest to the rating value of 3 which corresponds to the rating *suitable* in these four experimental values of T, and Figure A.3(c) supports it. Consequently we select the displaying time T to be four seconds.

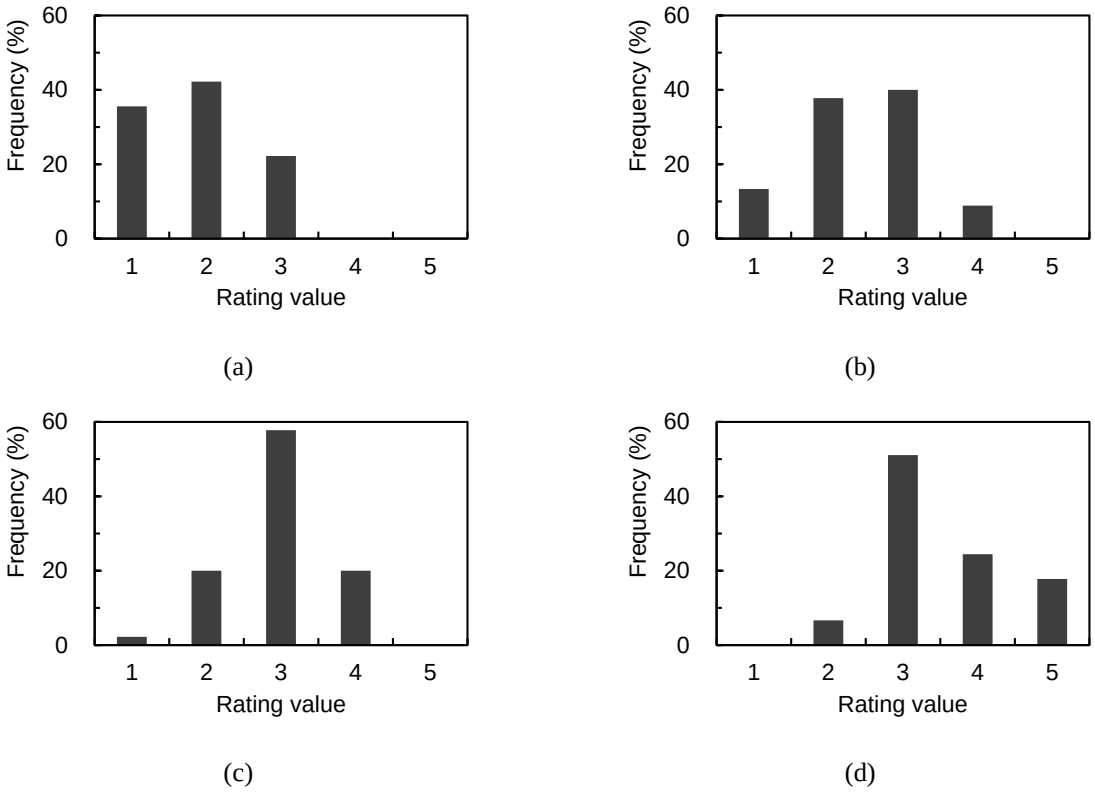


Figure A.3: Frequency distributions of the ratings on displaying duration for the time (a) *T*=2, (b) *T*=3, (c) *T*=4 and (d) *T*=5 seconds

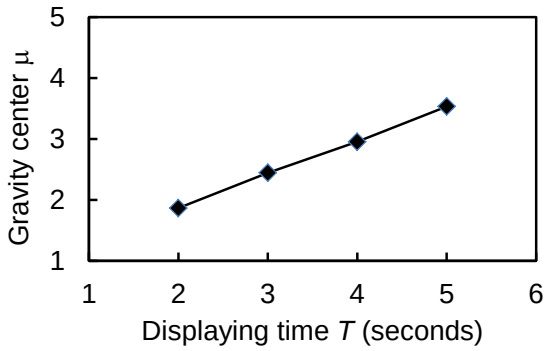


Figure A.4: Gravity center of the rating scores shown in Figure A.3

B. SELECTING SD ITEMS

B.1 Experimental method

The SD items to apply to the evaluation of halftone impressions are chosen from a source set of SD items through another preliminary experiment with subjects and test images. The source set has been composed of items of each of the following seven categories: *appearance*, *sensibility*, *activity*, *mass and texture*, *conditions*, *sense of value* and *attitude and character*,

which are generally used in psychological analysis. Thus, a total of 75 SD items have been gathered in the source set. Note that these items were actually written in Japanese in the experiment.

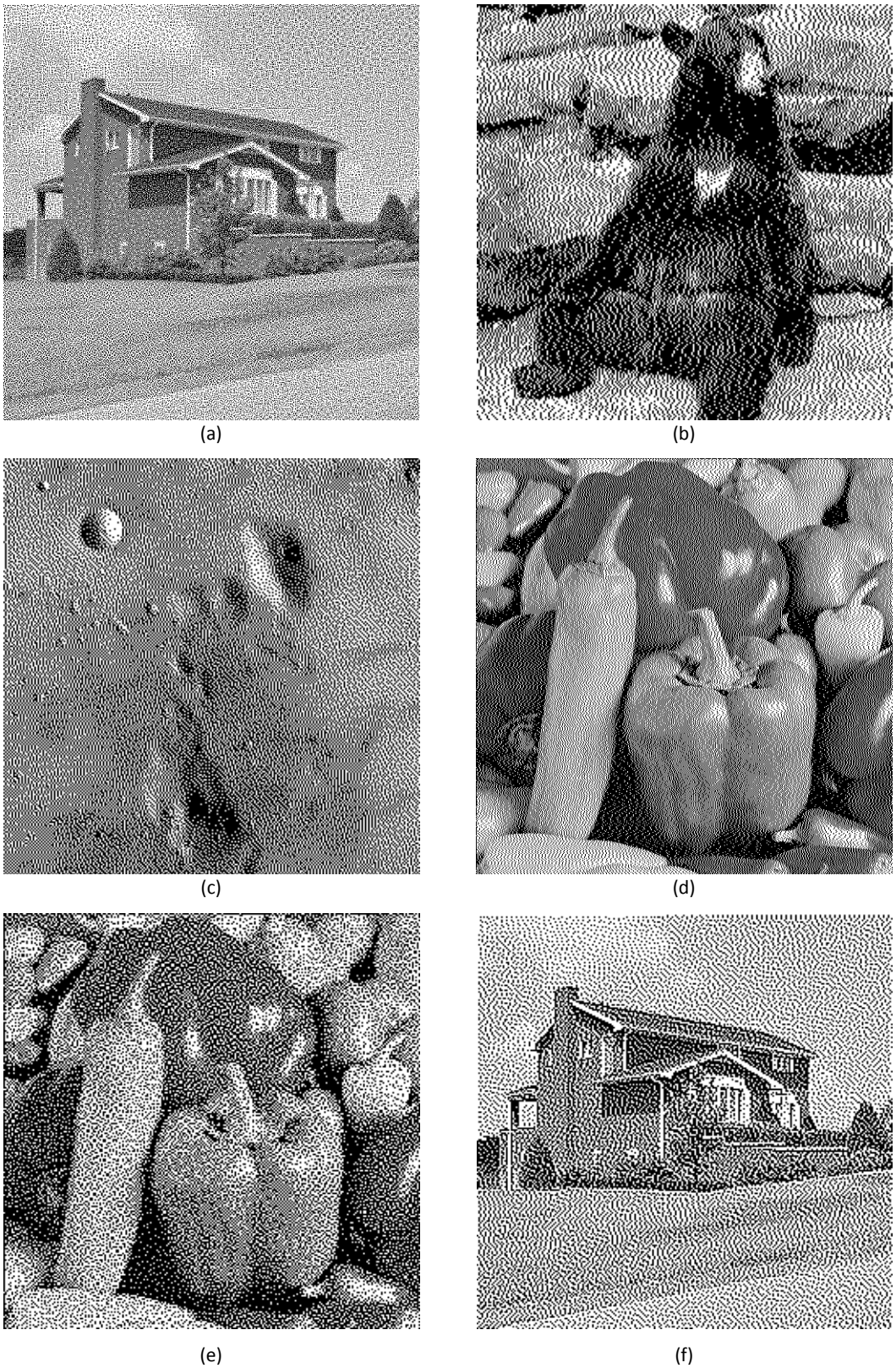
Figure A.5 shows a set of six halftone images that was used in this preliminary experiment. Two of the images are of size 512 x 512 pixels, and the other images are of size 256 x 256 pixels. Each image was produced by one of the halftoning methods EDF-FS, EDF-JJN and IGS. The image set was printed on a single A4-size sheet of photographic paper with a full-color photo printer, which has a printing resolution of 400 dots per inch. In printing, each image was printed at the pixel resolution of either 200 pixels per inch (ppi) or 400 ppi so that any of them was shown in the same size of 2.56 inch square. Thus, the test sheet represents both a variety of halftone patterns and two kinds of halftone resolutions.

Looking at the test sheet, every subject is asked to judge for each SD item of the source set whether the item can describe an impression that the subject himself/herself gets from the images. Forty-four subjects took part in the experiment.

B.2 Analysis of experimental result

From the experimental result, we obtain the approving rate of each SD item, that is, the ratio of subjects who have approved the capability of the item of describing halftone image impressions. Figure A.6 shows the approving rates of the 75 SD items of the source set in descending order. For example, the item *unclear-clear* is one of the SD items of the largest approving rate of 0.7 which means that 31 of 44 subjects have approved it in this experiment. In accordance with the result, we have determined 30 SD items in the descending order of the approving rates. Table A.3 summarizes the selection of SD items, including the examples of items in each category. Again note that there might be language gaps between Japanese and English in translating in this paper.

In subjective evaluation, an SD item is to be evaluated by measuring semantic degree between the two contradictorily adjective words. We use a scale of seven gradations for the measurement; gradations 1 and 7 correspond to the respective adjective words; gradation 4 represents being intermediate, or more specifically, *not impressed either way*. An example of the scale is illustrated in Figure A.7. The selected SD items are to be presented to subjects with each associated scale on a questionnaire sheet in the main experiment.



(a) (b) (c) (d) (e) (f)

Figure A.5: Digital halftone images used in the preliminary experiment. All the images are shown here with the same size as those actually printed. The printed pixel density of (a) and (d) is 200 pixels per inch (ppi), and the others 100 ppi. The images (c) and (f) were made by the error-diffusion methods, the others by the IGS method.

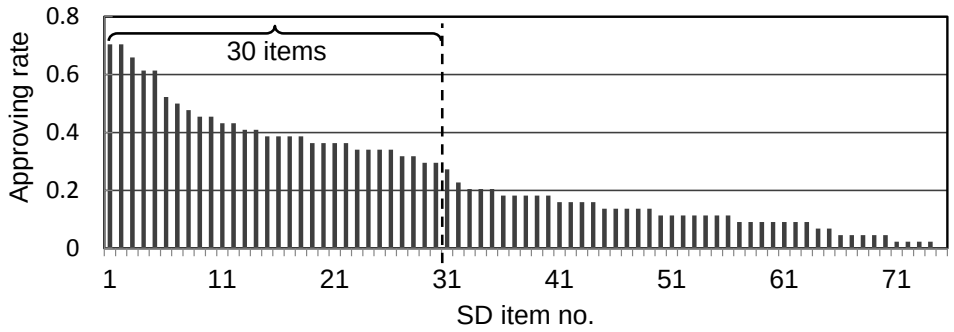


Figure A.6: Experimental result of selecting SD items

Table A.3: Selecting SD items

Category	Number of source items	Number of selected items	Examples
Appearance	11	9	*dark-bright* *ambiguous-distinctive*
Sensibility	7	5	*ugly-beautiful* *cold-warm*
Activity	6	3	*static-dynamic* *weak-powerful*
Mass and texture	7	4	*fine-coarse* *gloomy-shiny*
Conditions	14	4	*unnatural-natural* *old-new*
Sense of value	4	1	*low quality-high quality* *inexpensive-expensive*
Attitude and character	26	4	*delicate-harsh* *unfriendly-friendly*
Total	75	30	-

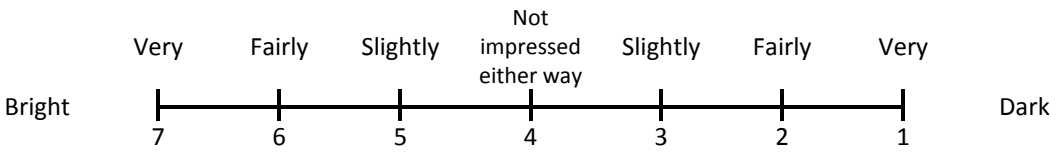


Figure A.7: Example of scale for replying to an SD item *dark-bright*

AIVP ADVANCES IN IMAGE AND VIDEO PROCESSING

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM

VOLUME 2, ISSUE 2
ISSN: 2054 - 7412

Hexagonal pixel-array for efficient spatial computation for motion-detection pre-processing of visual scenes

Nicoladie D. Tam
Department of Biological Sciences, University of North Texas, Denton, TX 76023, USA
nicoladie.tam@unt.edu

ABSTRACT

Motion-detection, edge-detection, and orientation-detection often require spatial computation of the light intensity difference between neighboring pixel cells. Pre-processing the image at the retinal level can improve the computational efficiency when the parallel processing can be achieved naturally by the spatial arrangement of the pixel-array. Pixel-arrays that require spatial pre-processing computation have different geometric constraints than plain pixel-arrays that do not require such local computation. Our analysis shows that geometric optimization of pixel-arrays can be achieved by using hexagonal arrays over rectilinear arrays. Hexagonal arrays improve the packing density, and reduce the complexity of the spatial computation compared to rectilinear square and octagonal arrays. They also provide geometric symmetry for efficient computation not only at the contiguous neighboring cell level, but also at the higher-order neighboring cell level. The light intensity difference at the higher-order cell level is used to compute the first-order and second-order time-derivatives for velocity and acceleration detections of the visual scene, respectively. Thus, hexagonal arrays increase the computational efficiency by using a symmetric configuration that allows pre-processing of spatial information of the visual scene using hardware implementations that are repeatable in all higher-order neighboring pixels.

Keywords: Hexagonal pixel-array, rectilinear pixels, geometric optimization, computational efficiency, spatial computation, parallel pre-processing.

1. INTRODUCTION

Rectilinear array is commonly used in spatial computation because of the orthogonal nature of the orthogonal axes and the ease of its implementation in the Cartesian coordinate system. Typical examples of rectilinear arrays are implemented in the pixel elements of computer display screens and camera image sensors, where the pixel elements are configured according to the orthogonal *x*- and *y*-axes. Yet, square array does not necessarily provide the most symmetrical geometric configuration for efficient computation in the spatial dimension, when

DOI: 10.14738/aivp.22.153
Publication Date: 4th April 2014
URL: http://dx.doi.org/10.14738/aivp.22.153

computation between the adjacent neighboring cells is required. Efficient spatial computation can be optimized by the spatial relationship and the distance between adjacent cells.

Due to these optimization constraints, rectilinear array may not be an optimal system for spatial computation when the computation requires minimization of distance among neighboring cells in a densely packed area. Since most pixel-arrays in a computer monitor or camera do not require any local computations involving neighboring cells, it is not important to optimize the geometric configuration of the pixel-array in terms of computational efficiency. The choice of rectilinear array or hexagonal array is optimized for the number of colors used in the display pixels. For example, rectilinear array is optimal for 2 or 4 color pixels, such as B/W (black-and-white) pixels, RGBW (red, green, blue, white), and CYGM (cyan, yellow, green, magenta) color pixels, while hexagonal arrays are optimal for 3 color pixels, such as RGB (red, green, blue) color pixels. These choices of the geometric array are not necessarily optimized for spatial computational efficiency, because no local computation is required for displaying the pixels or processing the imagine sensor pixels. But if spatial pre-processing of the visual scene is required, then geometric optimization of the pixel-arrays is important to improve the computational efficiency at the local pixels level. In order to take advantage of the geometric arrangements of the pixel, pre-processing of the visual scene analysis can be done locally at the photo-detector level before transmitting the information to the CPU (central processing unit) for post-processing. The pre-processing can include edge-detection, motion-detection and orientation-detection of the visual scene.

1.1 . Biologically-Inspired Hexagonal Array for Photo-Detectors

Hexagonal array, on the other hand, provides a more compact geometric configuration for local computation that requires a spatial relationship between adjacent cells, such as the retina in the lens eyes and ommatidia [1] in the compound eyes (see Fig. 1). The difference between camera image sensors and retinal cells is that camera image pixels detect light only, but retinal cells not only detect light, but also perform local computation to detect edge, orientation, and motion of the visual scene. The retinal cells pre-process the image signals so that the retina not only detects the intensity of light, but also performs scene analysis for edge-detection, spatial orientation-detection, and motion-detection on the detected image. The pre-processed patterns of spatial and motion orientation information on the visual scene are transmitted from the retina to the brain, in addition to the light intensity information of the image. Therefore, spatial processing of image pattern requires efficient computation at the local pixel level.

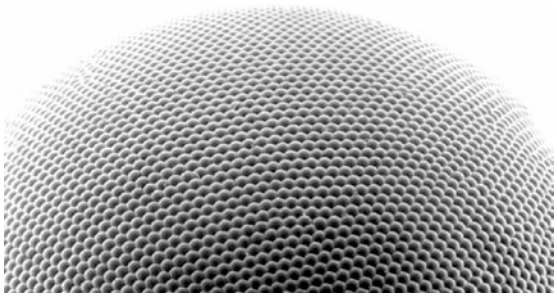


Figure 1: Image of a compound eye showing the hexagonal array of ommatidia (light sensors) of an insect.

Thus, it is important to optimize the geometric pattern of the retinal array of light-sensing pixels to perform efficient local computation of spatial information in relation to the neighboring cells. In this paper, we explore the computational efficiency using hexagonal array compared to rectilinear array. We will compare and contrast the geometric configurations of hexagonal array and rectilinear array in terms of spatial computation so that an optimal geometric design can be determined to improve its computational efficiency.

1.2 Spatial Computation by Local Pixels

Most camera image sensors detect light intensity information and transmit the image intensity information to the central processors for post-processing. In contrast, most retinal photo-detector cells in animals not only detect light intensity, but also process the image locally for edge-detection and motion-detection, before transmitting such orientation-detection information to the brain. Thus, the pre-processing visual image embeds not only a static image, but also a dynamic image that contains both motion and spatial information of the visual scene. Transmitting such orientation-detection dynamic image provides efficiency by compacting the image with both static and dynamic information content of the visual scene.

The first step in image acquisition is the detection of light intensity information by the camera image sensors. In animals, the image is acquired by the photo-detector cells of the retina or the ommatidia of the compound eye. Once the image is acquired, higher-order information concerning the visual scene can be extracted for edge-detection, spatial orientation-detection, and motion-detection [2]. Traditionally, this post-processing of image is performed by the CPUs of the computer instead of locally at the image-detecting array. Yet, the higher-order information can be computed much more efficiently by neighboring pixel cells locally prior to transmitting the image to the CPUs. This is because efficient processing of the local pixel information is highly parallelizable since the neighboring pixels are already geometrically arranged optimally for spatial processing. Instead of post-processing the image pixel-by-pixel sequentially at the computer, parallel pre-processing can be done locally at the image detector array level before transmitting the processed image to the CPU. It is more efficient to process the image by taking advantage of the geometric configurations of the pixels that are already arranged in parallel. This specialized computing architecture of the pixel-array

can lend itself to efficient computation for scene analysis, if the geometric arrangement of the pixels is optimized.

Optimization of the geometric pattern depends highly on the computations required for processing the image. The information contained in each pixel is often related to its neighboring pixels in most real-world visual scenes. For instance, the adjacent pixels are related for detection of a line in a visual scene. If an entire scene is moving in one direction, the adjacent pixels are also related by changing progressively as the scene propagates from one pixel to another. Such changes can be detected by the difference in intensity of adjacent pixels, since the velocity is computed by the distance of the corresponding pixels moved laterally divided by the time of motion. The direction of motion is detected by the direction in which the propagation occurs. Thus, the geometric arrangement of the pixels can provide the direction of motion and/or the orientation of the detected edges automatically prior to post-processing at the CPU.

2. GEOMETRIC OPTIMIZATION FOR SQUARE AND OCTAGONAL PIXEL-ARRAYS

Square array is a good candidate geometric array for computing the local differences in light intensity between neighboring cells. Because the array is orthogonal to each other, the direction (angle) of direct motion-detection is limited to 90^{o}. In a square array, each cell shares 4 contiguous neighbors. Detection of 90^{o}-motion can be detected optimally by the 4 contiguous neighboring cells. If the motion is shifted by a 45^{o} angle, computation between corner cells is required, but these corner cells are non-contiguous (see Fig. 2). Furthermore, the corner cells are not equidistance to the adjacent cells (that share contiguous contact between them).

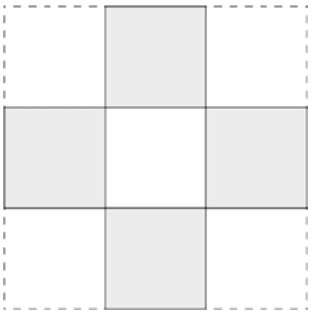


Figure 2: Diagram of a rectilinear square showing the center cell (in white) surrounded by the contagous neighboring cells (in shaded color).

To resolve this corner-cell problem, an interlaced octagon-surround, square-center tile array (see Fig. 3) can be used instead. This square-center arrangement surrounded by octagons will eliminate the non-contiguous corner cells. But this solution introduces an additional dilemma because of the difference in size and shape of the squares and octagons. This means each cell will detect a different amount of photons, which will introduce biases with this asymmetrical geometry, causing erroneous motion-detection. Thus, geometric symmetry of

adjacent cells and equal surface area for all cells are important considerations in spatial computation for proper motion-detection.

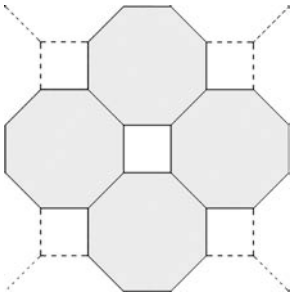


Figure 3: Diagram of a octagon-surround, square-center tile array showing the center cell (in white) surrounded by the contagous neighboring cells (in shaded color).

An alternative is to use a square-surround, octagon-centered arrangement surrounded by alternating squares and octagons (see Fig. 4). This geometric arrangement has all 8 contiguous neighboring cells, which will allow detection of both 45^o and 90^o motions without the corner-cell problem. This arrangement still has the same geometric asymmetry and different surface area as before. In fact, the geometric tile arrangement of Fig. 4 is exactly the same as the tile arrangement of Fig. 3, except that it is rotated by a 45^o angle (by comparing Figs. 3 and 4). Although this configuration resolves the corner-cell problem, it breaks the symmetry of having two classes of cells – those with 8 contiguous neighbors and 4 contiguous neighbors. Thus, it requires two sets of algorithms for computing the local light intensity differences that correspond to the direction of motion angle.

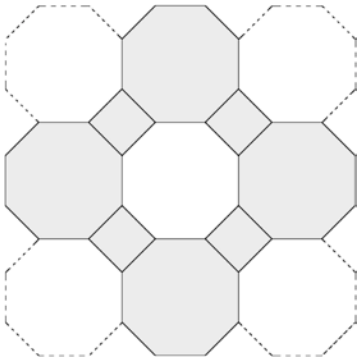


Figure 4: Diagram of a square-surround, octagon-center tile array showing the center cell (in white) surrounded by the contagous neighboring cells (in shaded color).

3. GEOMETRIC OPTIMIZATION FOR HEXAGONAL PIXEL-ARRAY

Hexagonal arrays are often found in compound eyes of insects and invertebrates, forming an array of ommatidia (photo-detectors) for light detection. Such hexagonal arrays (Fig. 1) are more commonly found in animals and plants than any other geometric arrays, such as rectilinear orthogonal arrays This is because motion-detection in compound eyes can be computed based on the local difference of light intensity between adjacent neighboring cells [3-5] of the ommatidia using the process called lateral inhibition [3, 6]. Lateral inhibition is a

contrast-enhancement computation used by an animal's neural circuitry to exaggerate the light intensity differences of the neighboring cells for edge-detection or two-point discrimination. An edge (or a point) is enhanced by suppressing the light intensity of its neighbors using the lateral inhibition spatial circuitry.

In contrast, hexagonal array provides the best candidate for contiguous neighboring-cell computation of local light intensity difference between adjacent cells without the complexity of the square or octagonal rectilinear array discussed above [7]. The first-order computation of light intensity differences can be computed using the contiguous neighboring cells at 60^o angles (see Fig. 5). With this hexagonal configuration, finer angular differences, such as 30^o angles, can be obtained using the second-order adjacent cells. Thus, this provides a means for symmetric computation using one set of computational algorithms to compute the light intensity difference of exactly 6 contiguous neighbors. Each successive higher-order neighbor will compute the light intensity difference with the angular direction rotated by 30^o successively.

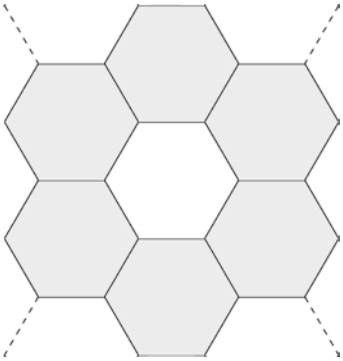


Figure 5: Diagram of a hexagonal array array showing the center cell (in white) surrounded by the contagous neighboring cells (in shaded color).

Edge-detection can be implemented locally at the pixel level by comparing the light intensity gradient of neighboring cells. Lines are detected by similar luminance in adjacent pixels, while edges are detected by sudden change in light intensity along the detected lines. Numerous studies have shown that hexagonal pixel-arrays are more efficient for edge-detection instead of rectilinear arrays by using spatial computation [2, 8-13]. There is a 40% computational efficiency using a hexagonal edge-detection operator [2, 11].

We propose that the use of hexagonal arrays to perform spatial computation can also improve the efficiency of motion-detection, not just edge-detection. The significance of this symmetrical geometry arrangement can be appreciated when the computation of light intensity difference between second-order and higher-order adjacent cells is required, such as the computation needed for finer angular increments, velocity and acceleration computation (second-order and third-order time-derivatives). Velocity of the image is computed by the first-order derivative of the light intensity difference between adjacent cells. Acceleration is computed by the second-order derivative of the light intensity difference between adjacent cells.

Insects (such as flies) often use this motion-detection of shadows to trigger the escape reflex circuitry to flee from predators when the velocity of the shadow exceeds the "critical velocity" threshold. Such escape requires the detection of both motion and direction of the predator's shadow, which can be detected instantaneously at the ommatidia level of the compound eyes using spatial computation. Motion is detected by the velocity and acceleration of the image based on the changes in local light intensity difference between adjacent photo-detector cells. The direction of motion is detected by the propagation of such local changes in these neighboring cells. This spatial computation can be done by a simple circuitry at the photo-detector level of the ommatidia to produce the escape reflex in insects that do not even have the complex circuitry of a brain.

4. GEOMETRIC OPTIMIZATION FOR PACKING DENSITY OF PIXEL CELLS

Hexagonal arrays also provide more optimal packing density than rectilinear square arrays. In order to prove that hexagonal arrays can achieve better packing density than square arrays, let us use circular pixel cells for the array, since a circle is symmetrical in all directions, and a circle will always form contacts with its neighbor, independent of whether the array is rectilinear or hexagonal.

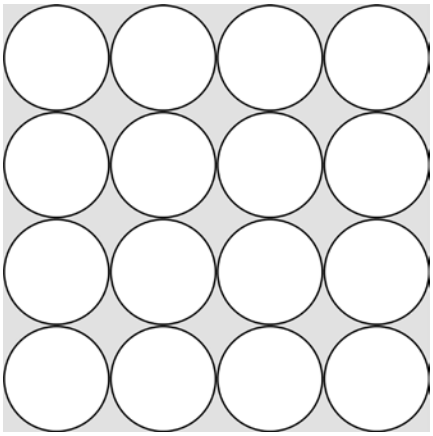


Figure 6: Diagram of a square array of circles to illustrate the packing density of rectilinear square array.

Fig. 6 shows the rectilinear array of circular pixels. The packing density is derived from the percentage of space occupied by the circular pixels compared to the empty space not occupied by the pixel:

$$d_{rect} = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{2} = 78.54\% \quad (1)$$

where d_{rect} denotes the package density for the rectilinear array, and r is the radius of the circle.

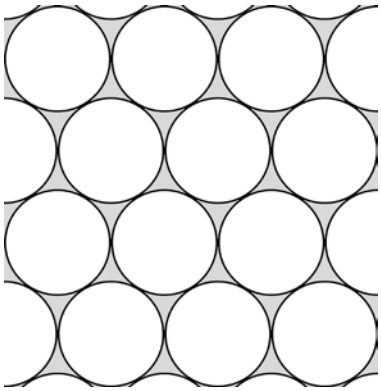


Figure 7: Diagram of a hexogonal array of circles to illustrate the packing density of rectilinear square array.

On the other hand, Fig. 7 shows the hexagonal array of circular pixels. The packing density for hexagonal array is given by:

$$d_{hex} = \frac{7\pi r^2}{3(3r)^2 \sin(\pi/3)} = \frac{7\pi}{27\sin(\pi/3)} = 94.05\% \quad (2)$$

where d_{hex} denotes the package density for the hexagonal array. This shows that hexagonal arrays have a drastic improvement in packing density over rectilinear arrays – 94.1% vs. 78.5%. Optimizing the pixel-array geometrically with hexagonal cells will have a significant improvement on packing pixels in motion-detecting cameras and in the compound eyes that require spatial computation with adjacent neighboring cells.

5. EDGE-DETECTION BY SPATIAL PROCESSING

The edge in a visual scene can be detected by the orientation of a line of sharp change in light intensity. Instead of merely detecting the light intensity of the pixels in the photo-detector array, edge-detection can be processed locally within the pixel-array. The abrupt change in light intensity of the neighboring pixels can be processed spatially within the photo-detector array by taking the light intensity difference of the neighboring pixels.

The orientation of the line can be detected by the similar light intensity in the longitudinal direction of the neighboring pixels and abrupt change in light intensity of the neighboring pixels in the orthogonal direction. Thus, intensity thresholds can be used to detect edges along the longitudinal and orthogonal directions.

Let p be the light intensity detected by the photo-detector. The difference between the light intensity of the neighboring cells, Δp_x, is given by:

$$\Delta p_{x12} = p_{x1} - p_{x2} \quad (3)$$

where p_{x1} and p_{x2} are the adjacent pixels in the x-direction (along the x-axis). Similarly,

$$\Delta p_{y12} = p_{y1} - p_{y2} \quad (4)$$

Δp_x represents the light intensity difference in the *y*-direction. The thresholds for edge-detection in the *x*-direction is given by:

$$\begin{cases} \Delta p_x < \varepsilon \\ \Delta p_y > \theta \end{cases} \quad \text{for } \varepsilon << \theta \tag{5}$$

The edge-detection in any arbitrary direction can be obtained by the rotational transformation of the *x*-axis. Since the geometrical arrangement of the pixel-array lends itself naturally to the orthogonal edge-detection directions for rectilinear pixel-arrays, the principle directions for optimal edge-detection are the horizontal (0^o) and vertical (90^o) directions, if the edges are aligned horizontally or vertically. This can be accomplished by setting the thresholds for the nearest-adjacent neighbors in the hardware implementation for the spatial processing. For diagonally oriented (45^o) edges, the next-adjacent neighbors can be used for setting the threshold.

For hexagonal pixel-arrays, edge-detection can be optimally computed by setting the thresholds for nearest-adjacent neighbors aligned in the 0^o, $60^o$and 120^o directions. Similarly, edges aligned along the 30^o, $90^o$and 150^o directions can be detected by the next-adjacent pixel neighbors. Because the distance between the nearest- and next-adjacent pixels are multiples of each other in the hexagonal array, the hardware implementations for edge-detection spatial computation are symmetrical in all directions (see Fig. 5). In contrast, since the distances between nearest- and next-adjacent neighbors are not multiples of each other for rectilinear pixel-arrays, the complexity of the hardware implementations increases because of the asymmetry in edge-detection for different directions (see Figs. 2-4). Thus, hexagonal pixel-arrays are ideal for edge-detection compared to rectilinear arrays.

6. IMAGE SMOOTHING BY SPATIAL PROCESSING

Since video images are often noisy, the static noise can introduce errors in the edge-detection algorithm when the light intensity exceeds the threshold by accidental noise. To eliminate the noise-effect, spatial smoothing can be done by averaging the neighboring pixels. Instead of computing the difference between neighboring pixels (as in edge-detection), the sum of the neighboring pixels can be used in the smoothing algorithm.

$$p_{x12} = p_{x1} + p_{x2} \tag{6}$$

Thus, the edge-detection can be processed after the image has been smoothed to reduce static noise. In other words, the hardware implementations requires only an adder (and a subtractor, which can be implemented by an adder using 2's complement arithmetic) for each neighboring pixels symmetrically located spatially in the pixel-array.

Note that different thresholds can be used to detect sharp vs. soft edges. Furthermore, since most edges don't necessarily span just the nearest neighboring pixels, multiple higher-

order next-neighbors can be used in both spatial smoothing and edge-detection in combination. The sharpness of the edge can be determined by how many higher-order next-neighbors the leading-edge spans. The length of the edge can be determined by how many higher-order next-neighbors the light intensity remains constant.

7. MOTION-DETECTION BY SPATIAL PROCESSING

Motion-detection can be computed by the rate of change of the local intensity difference of the neighboring cells (photo arrays). An adaptive threshold light intensity difference of the neighboring cells can be used to detect motion direction. Similar to edge-detection, it can detect motions in the 6 different directions at 60^o to each other for a hexagonal array, whereas for rectilinear array, it can only detect motions in 4 different directions at 90^o to each other along the principle axes. Motion-detection can be achieved efficiently using spatial processing algorithm implemented in the hardware.

Since the distance between contagious neighboring cells are known (fixed in the photo-detector array), the rate of change of light intensity can be used to compute the velocity of motion. Let x be the distance traveled in the visual scene, then the velocity of motion, v, is given by:

$$v = \frac{x}{\Delta t} \tag{7}$$

where Δt is the time-increment between each video frame (frame rate). In order to detect motions across different pixels, the corresponding (similar) light intensity would traverse across those pixels. One of the efficient methodologies for detecting motion is to incorporate the edge-detection computed earlier for motion-detection. If an edge is detected progressively along the neighboring pixels, then motion can effectively be detected. The thresholds for sharp edges and soft edges can be adjusted adaptively to accommodate for various visual scenes.

The velocity of the motion can be computed by the propagation of the edge along the neighboring pixels based on the center-to-center distance (Δx) between neighboring pixels. The advantage of hexagonal pixel-array is that the distances between nearest- and next-adjacent neighbors are multiples of each other; the motion-detection circuitry can be symmetrical in all directions, unlike rectilinear array. This symmetry allows for simple hardware implementation for motion-detection without the extra complexity in the asymmetrical nearest- and next-adjacent neighbors for rectilinear array.

8. CONCLUSION

Geometric optimization of pixel-arrays that require local spatial computation can be achieved by using hexagonal arrays, instead of using the traditional rectilinear arrays. Hexagonal arrays improve the packing density, as well as reducing the complexity of spatial computation, compared to rectilinear square or octagonal arrays. Hexagonal arrays also

provide geometric symmetry with the contiguous neighboring cells, as well as higher-order neighboring cells. Hexagonal array increases the computational efficiency by using a symmetric configuration that is repeatable in all higher-order cells. This simplifies the computational hardware implementation for motion-detection, edge-detection, and orientation-detection. The efficiency can also be improved by implementing the spatial computation at the photo-detector level using hexagonal arrays for pre-processing. It will increase the information content of the visual scene transmitted to the CPU (or the brain) by embedding the dynamic information of the pre-processed image that includes motion, edge and orientation-detection, in addition to the static information of a plain motionless image.

9.ACKNOWLEDGEMENT

I greatly appreciate Ms. Krista Smith for the helpful suggestions, and for proofreading the manuscript.

REFERENCES

[1] Blum, M. and T. Labhart, *Photoreceptor visual fields, ommatidial array, and receptor axon projections in the polarisation-sensitive dorsal rim area of the cricket compound eye.* J Comp Physiol A, 2000. 186(2): p. 119-28.

[2] Staunton, R.C., *The design of hexagonal sampling structures for image digitization and their use with local operators.* Image and Vision Computing, 1989. 7(3): p. 162-166.

[3] Coleman, B.D. and G.H. Renninger, *Theory of delayed lateral inhibition in the compound eye of limulus.* Proc Natl Acad Sci U S A, 1974. 71(7): p. 2887-91.

[4] Schultz, A.M. and M.J. Wilcox, *Parallel image segmentation using the L4 network.* Biomed Sci Instrum, 1999. 35: p. 117-21.

[5] Srinivasan, M.V. and G.D. Bernard, *The effect of motion on visual acuity of the compound eye: a theoretical analysis.* Vision Res, 1975. 15(4): p. 515-25.

[6] Laughlin, S., *A simple coding procedure enhances a neuron's information capacity.* Z Naturforsch C, 1981. 36(9-10): p. 910-2.

[7] Tam, D.C., *Computational efficiency circuitry for detecting shadows in hexagonal array of compound eye.* Neurocomputing, 2004. 58–60(0): p. 1073-1078.

[8] Davies, E.R., *Optimising computation of hexagonal differential gradient edge detector.* Electronics Letters, 1991. 27(17): p. 1526 – 1527.

[9] He, X., et al., *Bilateral Edge-detection on a Virtual Hexagonal Structure*, in *Advances in Visual Computing*, G. Bebis, et al., Editors. 2006, Springer Berlin Heidelberg. p. 176-185.

[10] Abu-Baker, S. and R.J. Green. *Detection of edges based on hexagonal pixel formats*. in *Signal Processing, 1996., 3rd International Conference on*. 1996.

[11] Staunton, R.C. *Hexagonal image sampling: A practical proposition*. in *Proc. SPIE 1008, Expert Robots for Industrial Use*. 1989.

[12] Gardiner, B., S. Coleman, and B. Scotney. *Multi-scale Feature Extraction in a Sub-pixel Virtual Hexagonal Environment*. in *Machine Vision and Image Processing Conference, 2008. IMVIP '08. International*. 2008.

[13] Davies, E.R., *Low-level vision requirements.* Electronics & Communication Engineering Journal, 2000. 12(5): p. 197 – 210.