

COMPRESSION OF HALFTONE VIDEO FOR ELECTRONIC PAPER

Chao-Yung Hsu,^{1,2} Chun-Shien Lu,^{1,*} and Soo-Chang Pei²

¹Institute of Information Science, Academia Sinica, Taipei, Taiwan, ROC

²Graduate Institute of Communication Eng., National Taiwan University, Taipei, Taiwan, ROC

ABSTRACT

Video halftoning is a key technology for use in the innovative display - electronic paper (e-paper). Since e-paper is power-limited, halftone video compression becomes an emerging issue but is still relatively unexplored. In this paper, this issue is addressed and a novel halftone video compression scheme is proposed. Our scheme is mainly composed of three components: block decomposition, block-based halftone quantization, and source coding. We evaluate the proposed method via lossless halftone video compression comparison with the famous standard, JBIG2. In addition, we demonstrate the rate-distortion performance of the proposed lossy halftone video compression method.

Keywords: Electronic paper, Halftone video compression, Quantization, Video halftoning

1. INTRODUCTION

Monitor industry is actively studying new display technologies to make the monitor lighter, thinner, and more portable. The electronic paper (e-paper) or smart paper is exactly the advanced and emerging display technology. Compared with traditional monitors, e-paper also consumes little power. Since digital halftoning format is the only format that many electronic devices can read, the development of digital image/video halftoning techniques is the key to meet the requirement of e-paper.

Digital halftoning refers to the physical process of converting a continuous tone images to black and white dots. In particular, video halftoning aims to transfer a general video sequence into a format that can be displayed on devices with limited intensity resolutions and color palettes. In the literature, only few video halftoning techniques [2, 8] were discussed. Recently, we have proposed a new video halftoning method [3], which is composed of spatial error diffusion and interframe reference error diffusion, and possesses the advantage of preserving temporal consistency.

An important problem in the applications of processing halftone videos is how to store the halftone video sequences effectively. Like traditional gray-scale/color video

sequences, which are compressed before transmission, halftone video compression definitely becomes an emerging research topic in e-paper. Data compression is undoubtedly a solution to this problem. JBIG [4, 5] is a lossless bi-level image compression scheme, which has been established in the Joint Bi-level Image Experts Group, standardized as ISO/IEC standard 11544 and as ITU-T recommendation T.82. In JBIG, arithmetic coding is used to achieve the goal of entropy coding. However, this scheme cannot be directly applied to halftone video compression because in some cases, we need to control the bit-rate even if the video quality is degraded and JBIG doesn't provide lossy compression. In addition, JBIG is only designed for compression of bi-level images.

In this paper, the relatively unexplored issue, halftone video lossy compression, is investigated, which is mainly composed of three components: block decomposition, block-based halftone quantization, and source coding. Both lossless and lossy halftone video compression are considered in this paper. For block decomposition, our scheme is proposed to compress halftone video frame in a block-based manner, like traditional block-based video coding. For halftone quantization, block replacement is used to achieve the goal of lossy compression. For source coding, a kind of block-based entropy coding is presented for lossless compression of the bitstream resulted from the quantization process (for lossy compression) or the block decomposition process (for lossless compression). We do not adopt arithmetic coding as source coding because it consumes much power, which is prohibitive for resource-limited mobile devices such as e-paper. The effectiveness of our method is demonstrated via experimental results.

2. VIDEO HALFTONING

The proposed halftone video compression scheme will be applied on the halftone video sequences generated using our video halftoning scheme [3]. The characteristic of this video halftoning scheme is: (i) group of picture (GOP) is adopted so that the halftone video frames are divided into I frame and P frame; (ii) temporal consistency is achieved to reduce the flickers that is an unlovely visual phenomenon. For an I

*Corresponding Author: Dr. C. S. Lu (lcs@iis.sinica.edu.tw)

frame, it is individually halftoned like via an image halftoning procedure. For a P-frame, it is halftoned by storing the different pixels between it and its corresponding reference I frame. Therefore, several pixels of a halftoned P frame are zeros. More specifically, since flicker flaw can be efficiently reduced by our method, the background during a series of video frames can be almost kept the same. As a result, this will lead to efficient compression of a halftone video sequence by saving a lot of bit rates used for encoding the changeable background due to the introduction of flicker flaws. In other words, a video halftoning scheme that can efficiently reduce the flickers is beneficial to achieve higher compression ratio under the constraint of the same bit-rate.

3. PROPOSED METHOD

Different from quantization of gray-scale image, quantizing the coefficients of halftone frames in the transform domain may make decoded pixel values undefined in the spatial domain. This is because a decoded pixel value must be either 0 or 1 to meet the characteristic of halftone images. Therefore, it is important to directly study spatial-domain quantization for halftone video compression.

Our halftone video compression scheme is mainly composed of three components: block decomposition, block-based halftone quantization, and source coding. It should be noted that the input to the proposed scheme is the halftone video generated using the video halftoning method described in Sec. 2.

3.1. Block Decomposition

Given a frame of size $m \times n$, it is first divided into $k \times k$ blocks, called coding units. A coding unit is further decomposed into a number of symbol blocks with a size of $r \times r$, where r is usually far less than k . Therefore, in a coding unit, there will be in total $(\frac{k}{r})^2 (= B)$ symbol blocks. Lossy compression is achieved by finding representative blocks, called quantized symbol blocks, to replace the original symbol blocks.

3.2. Block-based Halftone Quantization

The purpose of quantization in compressing halftone video is to find representative symbol blocks. That is, the symbol blocks will be represented with the quantized symbol blocks via a quantization process under the constraint of acceptable distortion. In the halftone quantization process, the smooth block (e.g., Fig. 1(a)) can be represented with different halftone blocks (e.g., Figs. 1(b) and (c)), both of which represent the same meaning in terms of gray-scale. In fact, there are in total $C_8^{16} (= 12870)$ halftone blocks to present the gray-scale block in Fig. 1(a), where 16 comes from the number of pixels in a block and 8 comes from the number of

either black points or white points in a smooth block. However, the frequently used blocks are limited. In the above example, Fig. 1(b) can be replaced by Fig. 1(c) since the visual difference between the two halftoned blocks is very small, and thus, they play the same role. For this reason, to achieve the goal of compression, the halftone quantization process will be finding frequently used blocks as the representative symbol blocks.

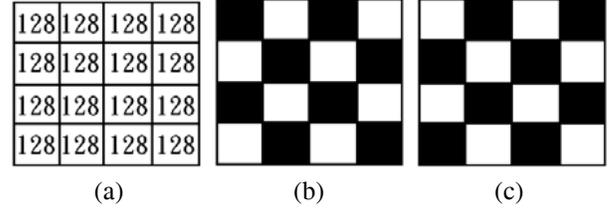


Fig. 1. An example of halftone quantization: (a) a 4×4 smooth block with gray-scale values 128; (b) a halftone version of (a); and (c) another halftone version of (a).

3.2.1. Compression Ratio vs. Number of Quantized Symbol Blocks

Let s ($1 \leq s \leq (\frac{k}{r})^2$) be the number of representative or quantized symbol blocks in a coding unit. Because $\lceil \log_2(s) \rceil$ bits are needed to store a compressed quantized symbol blocks, the total bits used to store s compressed quantized symbol blocks is $(\frac{k}{r})^2 \cdot \lceil \log_2(s) \rceil$. At the decoder side, s quantized symbol blocks are stored required for decoding purpose and $r^2 \cdot s$ bits are needed to store them. Therefore, there are in total

$$(\frac{k}{r})^2 \cdot \lceil \log_2(s) \rceil + r^2 \cdot s \quad (1)$$

bits required to store a coding unit during compression. With Eq. (1), the compression ratio of this coding scheme is derived as:

$$CR = \frac{k^2}{(\frac{k}{r})^2 \cdot \lceil \log_2(s) \rceil + s \cdot r^2}. \quad (2)$$

Since s in Eq. (2) is variable, it can be properly adjusted to achieve lossy compression of different compression ratios.

However, to achieve the target compression ratio, it is necessary to determine the number of quantized symbol blocks, s . Based on Eq. (2), we can obtain the relationship between the target compression ratio, CR , and the number of quantized symbol blocks, s , as:

$$CR \leq \frac{k^2}{(\frac{k}{r})^2 \cdot \log_2(s) + s \cdot r^2}, \quad (3)$$

because $\lceil \log_2(s) \rceil$ is larger than or equal to $\log_2(s)$. Eq. (3) can be further derived as:

$$0 \leq 2^\beta - s \cdot 2^{s \cdot \alpha^2}, \quad (4)$$

where $\alpha = \frac{r^2}{k}$ and $\beta = \frac{r^2}{CR}$. Let CR^t denote the target compression ratio and let CR^c denote the compression ratio calculated based on Eq. (4). For the s that satisfies the condition of minimizing $|CR^t - CR^c|$ can be used to achieve the target compression ratio.

As an example, let a coding block be of size 128×128 and let a symbol block be of size 4×4 . There are $(\frac{128}{4})^2 = 1024$ symbol blocks in a coding block. If $CR = 3$, then the number of quantized symbol blocks used to represent the symbol blocks is:

$$s \cdot 2^{s \cdot (\frac{16}{128})^2} \leq 2^{\frac{16}{3}}. \quad (5)$$

The final value of s is calculated to be 29, which achieves the compression ratio .3.012 that is nearest to the target one.

After the number of quantized symbol blocks is decided as s , the symbol blocks with the first s largest occurrence probabilities are determined as the quantized symbol blocks. Under this circumstance, the mean square quantization error (MSQE) can be written as:

$$MSQE = \sum_{i=1}^s p_i \cdot (e_i)^2, \quad (6)$$

where p_i is the probability of symbol block i appearing in a coding unit and e_i is the quantization error, which is calculated as the number of different halftone bits between the symbol block i and its quantized symbol block. The smaller the s is, the larger the compression ratio is. If $s = B$, then $MSQE = 0$ and no compression occurs.

3.2.2. Selection of Quantized Symbol Blocks

For each symbol block of size $r \times r$, its binary form is first directly read from the halftone bits in a row-wise order, and then the binary form is decimalized. Finally, the occurrence probabilities of symbol blocks are calculated according to the decimal form. Once s is determined, as described in Sec. 3.2.1, the symbol blocks with the first s largest occurrence probabilities are selected as the quantized symbol blocks.

3.3. Source Coding

After a coding unit is quantized in a lossy manner, it is further compressed via the source coding to achieve lossless compression. Source coding is used to losslessly decrease the average code length of symbols in a coding system. The major concept is to use less bits to represent the symbols that frequently occur, and use more bits to represent the symbols that occur infrequently. In this paper, the symbol is, in fact, a block of size $r \times r$. As a result, we call the used entropy coding, block-based entropy coding. Since P frames are always composed of zero halftone bits, run-length coding is further applied to increase compression ratio.

4. EXPERIMENTAL RESULTS

In the experiments, a number of 24-bits color video sequences [7] with frame size of 640×480 and frame rate of 15 frames per second were used. Among them, the results obtained from the ‘‘Vassar’’ video with slow motions and the ‘‘Ballroom’’ video with large motions were reported here. The size of coding unit was set to 80×80 . A symbol block is with size of 4×4 . As a result, there were in total $(\frac{80}{4})^2 = 400$ symbol blocks in a coding unit. Lossless compression and lossy compression were, respectively, used to verify the compression capability of our method.

In order to make sure that the method we proposed can efficiently compress halftone video, we first verify the lossless capability of our method and compare it with the lossless coding standard, JBIG2 [6]. For fair comparison, the same halftone video organized in the same GOP structure was used as the input to our method and JBIG2. Our method can also achieve lossless compression if halftone quantization, as described in Sec. 3.2, is ignored. The compression ratios for the first 30 frames of the ‘‘Vassar’’ and ‘‘ballroom’’ video sequences are, respectively, shown in Fig. 2 and Fig. 3. For ‘‘Vassar,’’ the first frame is an I frame and the others are P frame. For ‘‘ballroom,’’ the frames with indices 1, 14, 25, and 29 are I frames. It can be observed from Fig. 2 and Fig. 3 that the compression ratios of I frames are lower than those of P frames. In addition, it is obvious that our method consistently obtains higher compression ratios than JBIG2 for P frames. For I frames, our method and JBIG2 nearly obtain the same compression ratios.

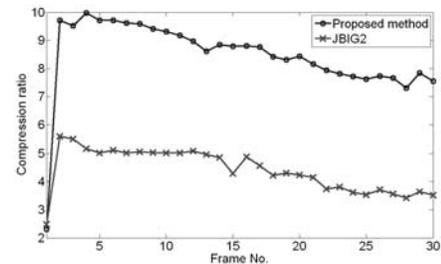


Fig. 2. Comparison of lossless compression ratio for the ‘‘Vassar’’ sequence between our method and JBIG2.

For verification of lossy compression, the rate-distortion performance was adopted. In this paper, it should be noted that the number of quantized symbol blocks is inversely proportional to the compression ratio. To measure the quality of compressed halftone video, Halftone PSNR (HTPSNR) was adopted as the criterion in this paper. HTPSNR is defined between two (halftone) video frames, X and Y as:

$$HTPSNR(X, Y) = PSNR(HVS(X), HVS(Y)), \quad (7)$$

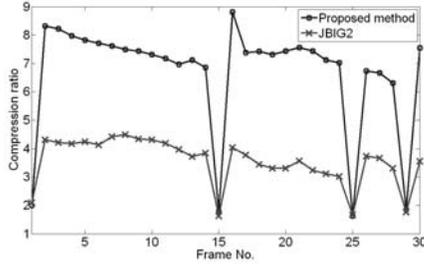


Fig. 3. Comparison of lossless compression ratio for the “ballroom” sequence between our method and JBIG2.

where $HVS()$ denotes a contrast sensitivity function of the human visual system [1]. In Eq. (7), the video frame X is a compressed halftone video frame and Y is an original gray-scale video frame. The RD performance for the two videos is shown in Fig. 4. The original uncompressed bit rate is 4500 Kbits/s.

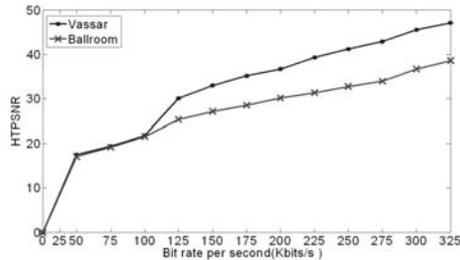


Fig. 4. Rate-distortion performance of our halftone video lossy compression scheme.

For visual quality inspection, the results are shown in Fig. 5 for comparison. Fig. 5(a) shows an original video frame with a size of 128×128 while Figs. 5(b)~(d), respectively, show the lossy compression results with 128, 32, and 8 quantized symbol blocks. We can observe there is obvious quality degradation in Fig. 5(d). These results suggest that the visual quality can be acceptable when the number of quantized symbol blocks is large than or equal to 32.

5. CONCLUSION

In view of the fact that video halftoning will play an important role in supporting the use of the emerging display device, e-paper, this paper proposes a new halftone video compression method. For lossless compression, comparison with the binary image compression standard, JBIG2, indicates that our method achieves effectiveness compression. For lossy compression, we have demonstrated the RD performance. To our knowledge, this should be the first work in halftone video compression. We believe that halftone video compression will continue to play an important role for the

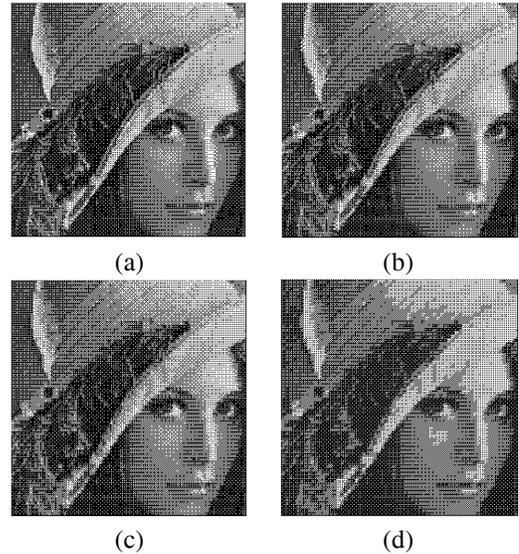


Fig. 5. Lossy halftone video compression: (a) the original video frame; (b)~(d), respectively, show the lossy compression results with 128, 32, and 8 quantized symbol blocks.

upcoming display - electronic paper.

Acknowledgment: This research was supported by the National Science Council under grants NSC 96-2221-E-001-029 and NSC 95-2422-H-001-031.

6. REFERENCES

- [1] P. J. Barten, “Physical model for the contrast sensitivity of the human eye,” in *Proc. IS&T/SPIE Int. Symp. on Electronic Imaging Science and Technology*, Vol. 1666, San Jose, CA, Feb. 9-14, pp. 57-74, 1992.
- [2] C. Gotsman, “Halftoning of image sequence,” *Vis. Comput.*, vol. 9, no. 5, pp. 255-266, 1993.
- [3] C. Y. Hsu, C. S. Lu, and S. C. Pei, “Video Halftoning Preserving Temporal Consistency,” *Proc. IEEE Int. Conf. on Multimedia and Expo*, Beijing, China, 2007.
- [4] *ISO CCITT Recommend. T.4, Standardization of group 3 facsimile apparatus for document transmission*, 1980.
- [5] *ISO/IEC JTC1 CD 11544: Coded representation of picture and audio information-progressive bi-level image compression*, 1993.
- [6] *IBIG2 Final Draft International Standard*, ISO/IEC JTC1/SC29/WG1N1545, Dec. 1999.
- [7] Mitsubishi Electric Research Laboratories, “MERL multiview video sequences,” <ftp://ftp.merl.com/pub/avetro/mvctestseq>.
- [8] Z. Sun, “Video Halftoning,” *IEEE Trans. on Image Processing*, Vol. 15, no. 3, pp. 678- 686, 2006.