International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 24 (2017) pp. 15664-15667
© Research India Publications. http://www.ripublication.com

Analysis of color transforms for lossless frame memory compression

Joungeun Bae[1] and Hoon Yoo[2]*
[1]*Department of Computer Science, SangMyung University, Korea.*
[2]*Full Professor, Department of Electronics, SangMyung University, Korea.*
**Corresponding author*

Abstract

This paper describes the analysis of color transforms for lossless compression and proposes a line-based lossless frame memory compression using a color transform. Line-based compression methods are required to save frame memory in display devices. A color transform is not included in the existing lossless frame memory compression based on the modified Hadamard transform (MHT) and adaptive Golomb-Rice (AGR) coding. Here we propose a color transform for the existing method by use of analyzing candidate color transforms such as YCoCg-R and RCT. Experimental results indicate that the proposed method gives a shorter bitrate compared with the existing method.

Keywords: Lossless compression, Color transform, Display devices

INTRODUCTION

Compression is a necessary technique to efficiently store and transmit large amounts of image data. Especially, a low-complexity compression is key condition to overcome the limited resources of 3-dimensional display devices [1-4]. Compression methods are divided into lossy and lossless. The lossy compression causes irreversible quality loss. Thus, the lossless compression is suitable for keeping image quality untouched on the display devices.

Several frame lossless compressions have been discussed to propose low computational-complexity algorithms for display devices. Block-based frame compression methods have been studied which are also used in video coding standards [5-13]. However, these methods cause extensive memory traffic because the frame memory in display devices is transmitted line by line.

Line-based algorithms are required since the frame memory in a display device is processed line by line. Thus, line-based lossless compressions have been addressed [14-17]. Among those studies, lossless compression for LCD devices has been proposed for the first time [14]. This method uses 8-point MHT for band separation and AGR for entropy coding. This method achieved a good performance in terms of compression ratio [17]. However, this method does not consider a color transform in a preprocessing step.

Applying a color transformation to a compression method increases the compression ratio because it converts redundant components in the RGB to less correlated components. Normally a color transform converts RGB components into one luminance and two chrominances. The color transform applied for lossless color image compression must be reversible, thus it should be implemented by an integer operation. During a color transform, a component range expands, which means that the number of bits required to store the transformed pixel increases. Thus, it is necessary to consider the component range expansion when applying a color transform to the existing compression.

In this paper, we present a method of applying color transform for a lossless frame memory compression providing a line-based processing method. Analysis of color transforms describes that they lead to better compression by reducing the correlation of RGB components. Based on the analysis, we propose a modified method of the existing method considering a component range expansion from a color transform. Lossless color transforms included in this analysis are YCoCg-R used in JPEG XR and RCT used in JPEG 2000. Experimental results show that the proposed method can reduce bitrate of code by 20% compared to the existing method without a color transform.

LOSSLESS COLOR TRANSFORM

Karhunen-Loeve Transform (KLT) has been used to optimize the decorrelation of color components. However, applying KLT to compression is not feasible because computational complexity of KLT is very high. Thus, color transforms based on KLT have been proposed. The color transforms must be an integer-reversible so that they can be performed in lossless frame compression. Furthermore, an analysis of component range expansion and complexity of transforms is required to predict the efficiency of the transformation.

$$\begin{array}{lcl} Co = R - B & & t = Y - \lfloor Cg/2 \rfloor \\ t = B + \lfloor Co/2 \rfloor & & G = Cg + t \\ Cg = G - t & \Leftrightarrow & B = t - \lfloor Co/2 \rfloor \\ Y = t + \lfloor Cg/2 \rfloor & & R = B + Co \end{array} \tag{1}$$

$$\begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix} \approx \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ -1 & 0 & -1 \\ -1/2 & 1 & 1/2 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \Leftrightarrow \begin{bmatrix} R \\ G \\ B \end{bmatrix} \approx \begin{bmatrix} 1 & 1/2 & -1/2 \\ 1 & 0 & 1/2 \\ 1 & -1/2 & -1/2 \end{bmatrix} \begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix} \tag{2}$$

$$\begin{aligned} Cv &= R - G & \qquad G &= Y - \lfloor (Cu + Cv)/4 \rfloor \\ Cu &= B - G \qquad \Leftrightarrow & R &= Cv + G \\ Y &= G + \lfloor (Cu + Cv)/4 \rfloor & B &= Cu + G \end{aligned} \qquad (3)$$

$$\begin{bmatrix} Y \\ Cu \\ Cv \end{bmatrix} \approx \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 0 & -1 & 1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \Leftrightarrow \begin{bmatrix} R \\ G \\ B \end{bmatrix} \approx \begin{bmatrix} 1 & -1/4 & 3/4 \\ 1 & -1/4 & -1/4 \\ 1 & 3/4 & -1/4 \end{bmatrix} \begin{bmatrix} Y \\ Cu \\ Cv \end{bmatrix} \quad (4)$$

The YCoCg-R is a color transform used in JPEG-XR, the integer-reversible equation is (1), and the transformation is (2) [18]. Each component of RGB is converted into Y, Co, and Cg, respectively. The YCoCg-R transform, approximated to KLT, calculates the green channel at the highest ratio, considering human perception. In this transform, one bit of component range expansion occurs in the chrominances. Four add operators and two shift operators are required per pixel, in terms of computational complexity.

The RCT is used in JPEG 2000. The forward and inverse RCTs are operated by (3) and the transform is given by (4) [19]. Each component of R, G and B is converted into Y, which is a luminance component, and Cu, Cv, which are chrominance components. One bit of component range expansions occurs in the chrominance components. In terms of complexity, there are four add operators and one shift operator per pixel.

Component range expansion is inappropriate when transforms are performed in a limited data space. Thus, mRCT is a version of RCT with a modular operation, which is a color transform without component range expansion [20]. However, this transform has trade-off between component range expansion and complexity.

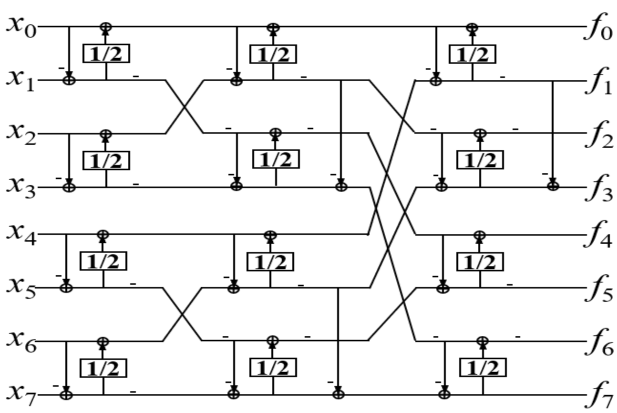


Figure 1: Signal flow of 8-point MHT.

EXISTING METHOD

Conventional lossless frame memory compression without color transform is a method for line-based processing and display devices [14]. The line data of the input image becomes a bit stream segment after this compression. This method uses 8-point MHT with a lifting scheme for decorrelation. Lifting schemes is well known for fast wavelet transforms [21]. Fig. 1 shows a signal flow when a size of 8 points block is input.

The overall flowchart of the existing method is depicted as Fig. 2. This method performs a total of two 8-point MHTs. After performing the first MHT, only the DC components are collected and the 8-point MHT is applied one more time. This process provides high compression efficiency by reducing the correlation of DC components.

AC components are entropy coded through Adaptive Golomb-Rice (AGR). Rice mapping converts a negative number to a non-negative number since AGR coding can not process a negative number, defined as (5). AGR coding uses a different binary code of k for each line, which is a value that can maximize compression performance. The value of k is calculated as (6), where length of k minimizes the sum of GR code lengths.

$$v = \begin{cases} 2c, & c \geq 0 \\ -2c - 1, & c < 0 \end{cases} \qquad (5)$$

$$length = k + 1 + v/2^k \qquad (6)$$

The final step is packing as a bit stream, which is done in the following order. The value of k is packed with 3-bit fixed length coding (FLC) in the far-left position. Next, DC components are packed with 8-bit FLC and AC components packed with AGR code.

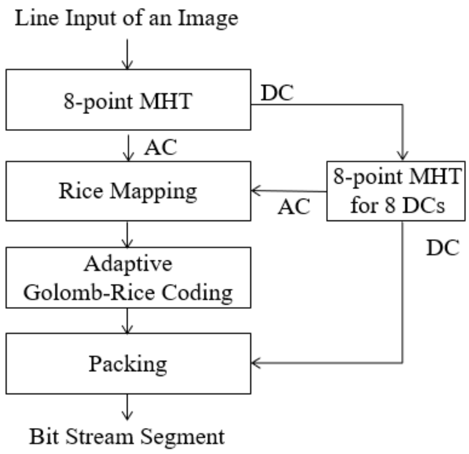


Figure 2: Existing method for Y component (method1).

PROPOSED METHOD

In this paper, we apply a color transform to existing methods and propose modified methods accordingly. U and V components contain negative numbers when the RGB components are converted to YUV components through the color transform. It is not appropriate that U and V components are input as they are in the existing method, since the existing method only processes non-negative numbers.

The flowchart of the modified compression method considering the negative number of U and V components is shown in Fig.3. Separating the band into two 8-point MHT is the same as the existing method. DC and AC of U and V components are coded with AGR unlike DC are packed with FLC in the existing method. The Existing method is defined as method1 and the modified method is defined as method2.

A flow chart of the proposed method is illustrated in Fig.4. After the color transform, the value of Y is processed through method1, and the values of U and V are processed through method2. The process of band separation in entropy coding is depicted in Fig.5. Fig. 5 (a) is the band separation of method 1 and Fig. 5 (b) is the band separation of method2. Fig. 5 (c) shows the whole band separation process when method 1 and method 2 are applied to YUV components, respectively.

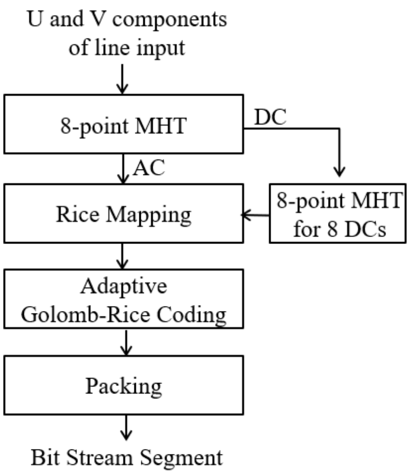


Figure 3: Modification for U and V components (method2).

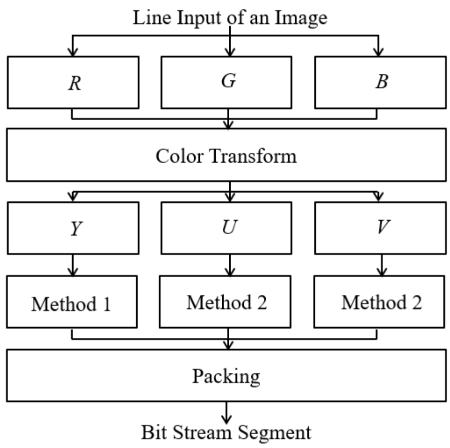
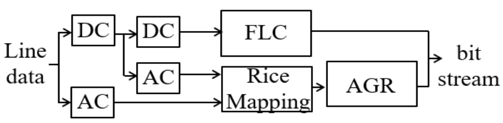
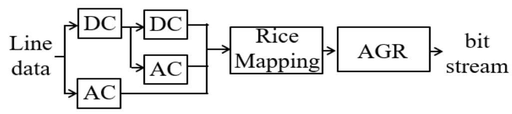


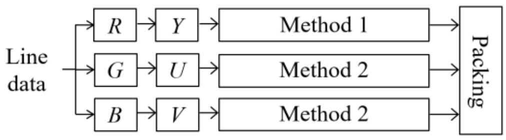
Figure 4: Flowchart of the proposed method.



(a)



(b)



(c)

Figure 5: Subband decomposition of the entropy coding. (a) method1, (b) method2, (c) proposed method with method1, method2

EXPERIMENTAL RESULTS

An experiment for obtaining the bitrate was conducted to evaluate the proposed method. The input images were 10 images released for the compression study of color image by Kodak corporation. Table 1 shows the bitrate obtained by the existing method and the bitrate obtained by the proposed method applying the color transform of YCoCg-R or RCT. The bitrate is the value of compressed file length divided by the pixel size of an input image and defined as (7). It is a value indicating the number of bits required for one pixel. The bitrate of the uncompressed image is 24 bit/pixel.

$$bitrate = \frac{Compressed\ bitdata\ size}{pixels} \quad (7)$$

Experimental results show that the bitrates of the proposed methods are shorter than those of the existing method. The average bitrate of RCT was 20% shorter than that of the existing method. The bitrate of RCT was slightly shorter than that of YCoCg-R. This experiment shows that color transforms can improve compression performance by eliminating redundancy and correlation of RGB components.

Table 1: Bitrate

Kodak	Existing method [14]	Proposed Method1 (YCoCg-R)	Proposed Method2 (RCT)
1	18.96	14.31	14.12
2	15.52	14.20	13.60
3	14.96	12.70	12.49
4	16.26	14.15	13.57
5	19.65	15.40	15.37
6	16.47	12.85	12.64
7	16.16	13.43	13.21
8	20.76	15.79	15.73
9	15.78	12.29	12.32
10	16.09	12.74	12.77
Average	17.06	13.79	13.58

CONCLUSION

This paper has proposed a method of applying color transform to lossless frame compression for display devices. Analysis was performed to select color transforms suitable for lossless compression. The proposed lossless frame compression method considers the component range expansion generated by the color transform. Experimental result shows that the reduced bitrate by applying color transform.

REFERENCES

[1] Y. Lee, and H. Yoo, "Low-cost 3D motion capture system using passive optical markers and monocular vision," Elsevier Optik, vol. 130, pp.1397-1407, Feb. 2017.

[2] Y. Lee, and H. Yoo, "Three-dimensional visualization of objects in scattering medium using integral imaging and spectral analysis," Elsevier Optics and Lasers in Engineering, vol. 77, no. 2, pp. 31-38, Feb. 2016.

[3] M. Ha, and H. Yoo, "Lossless Compression of 3D Geometry Data via Optimal Selection," International Journal of Applied Engineering Research, vol. 11, no. 21, pp. 10640-10643, Nov. 2016.

[4] M. Ha, and H. Yoo, "Preprocessing Using Classification for Lossless Compression of 3D Geometry Data," International Journal of Applied Engineering Research, vol. 11, no. 13, pp. 7947-7950, July 2016.

[5] S. H. Lee, M. K. Chung, S. M. Park, and C. M. Kyung, "Lossless frame memory recompression for video codec preserving random accessibility of coding unit," IEEE transactions on consumer electronics, vol.55, no. 4, Nov. 2009.

[6] J. Kim, J. Kim, and C. M. Kyung, "A lossless embedded compression algorithm for high definition video coding," Multimedia and Expo(ICME) 2009. IEEE International Conference on, pp.193-196, Jul. 2009.

[7] X. Bao, Z. Dajiang, and G. Satoshi, "A lossless frame recompression scheme for reducing DRAM power in video encoding," Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on. IEEE, pp.667-680, Jun. 2010.

[8] Yang. L, X. He, G. Zhang, L. Qing, and T. Che, "A low complexity block-based adaptive lossless image compression," Elsevier Optik-International Journal for Light and Electron Optics, vol. 124, no. 24, pp. 6545-6552, Dec. 2013

[9] D. Venugopal, S. Mohan, and S. Raja, "An efficient block based lossless compression of medical images," Elsevier Optik-International Journal for Light and Electron Optics, vol. 127, no. 2, pp. 754-758, Jan. 2016.

[10] S. Andriani, and G. Calvagno, "Lossless compression of color sequences using optimal linear prediction theory." IEEE Transactions on Image Processing, vol. 17, no. 11, pp. 2102-2111, Sep. 2008.

[11] D. Brunello, G. Calvagno, G. Z. Mian, and R. Rinaldo, "Lossless compression of video using temporal information," IEEE Transactions on Image Processing, vol. 12, no. 2, pp. 132-139, Apr. 2003.

[12] J. Lei, X. Zou, Z. Wu and W. Fan, "Research of an image map encoding algorithm on frame buffer," ASIC, 2007. ASICON'07. 7th International Conference on. IEEE, pp. 894-897, Oct. 2007.

[13] X. Lian, Z. Liu, W. Zhou, and Z. Duan, "Lossless frame memory compression using pixel-grain prediction and dynamic order entropy coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 1, pp. 223-235, Aug. 2015.

[14] T. L. B. Yng, B. G. Lee, and H. Yoo, "A low complexity and lossless frame memory compression for display devices," IEEE transactions on consumer electronics, vol. 54, no. 3, pp. 1453-1458, Oct. 2008.

[15] H. T. Yang, J. W. Chen, H. C. Kuo, and Y. L. Lin, "An effective dictionary-based display frame compressor," Embedded Systems for Real-Time Multimedia, 2009. IEEE/ACM/IFIP 7th Workshop on. IEEE, Oct. 2009.

[16] S. Dikbas, and F. Zhai, "Lossless image compression using adjustable fractional line-buffer," Elsevier Signal Processing: Image Communication, vol. 25, no. 5, pp. 345-351, Jun. 2010.

[17] H. C. Kuo, and Y. L. Lin, "A hybrid algorithm for effective lossless compression of video display frames," IEEE Transactions on Multimedia, no. 14, vol. 3, pp.500-509, Jun. 2012.

[18] H. S. Malvar, G. J. Sullivan, and S. Srinivasan, "Lifting-based reversible color transformations for image compression," Proc. SPIE, vol. 7073, 2008.

[19] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," IEEE transactions on consumer electronics, vol. 46, no. 4, pp. 1103-1127, Nov. 2000.

[20] R. Starosolski, "New simple and efficient color space transformations for lossless image compression," Elsevier Journal of Visual Communication and Image Representation, vol. 25, no. 5, pp. 1056-1063, Jul. 2014.

[21] H. Yoo, and J. Jeong, "Signal-dependent wavelet transform and application to lossless image compression," Electronics Letters, 2002, vol. 38, no. 4, pp.170-172, Aug. 2002.