An Image Compression Method for Frame Memory Size Reduction using Local Feature of Images

Yuki FUKUHARA† Akihisa YAMADA†,‡ Takao ONOYE†

†Dept. Information Systems Eng.
Osaka University
Suita, Osaka, 565-0871 Japan
{fukuhara.yuki,a-yamada,onoye}@ist.osaka-u.ac.jp

‡Electronic Components and Devices Group
Sharp Corporation
Fukuyama, Hiroshima, 721-8522 Japan
yamada.akihisa@sharp.co.jp

Abstract— This paper proposes a method of image compression aiming at reduction of frame memory size of digital appliances. In spite of adopting a variable length coding, this method can guarantee the minimum compression ratio by adaptively controlling pixel data reduction rate. Utilizing local feature of images, the reduction rate is more finely controlled so as to maintain visual quality of images. Experimental results show that it attains a compression ratio of 1/3, while keeping visual quality of images by means of adequate bit allocation.

I. Introduction

Recently, high-resolution imaging devices are widely used in such as Digital TVs, digital cameras, and so on. These devices require a large size of frame memory to store image data, which causes problems such as cost and power consumption increase. In order to solve these problems, the image data is compressed before being stored to frame memory, and the decompress process is performed prior to image processing, to reduce the capacity of frame memory. Requirements for such compression techniques are represented as follows. (1) The minimum compression ratio must be guaranteed for the compressed data stored to frame memory, (2) the amount of computation must be small enough to enable real-time processing, (3) compressed images must maintain high quality, (4) the size of compression and decompression circuits must be small, and (5) compression must be performed within a single frame.

Several block based image compression methods which guarantee a minimum compression ratio have been proposed[1],[2], however, these methods need memory for three or more than three lines because of compression by 4×4 blocks. On the other hand, [3] proposes a pixel by pixel compression method which does not require line memory. This fixed-rate image compression, which is based on predictive coding with variable length coding, guarantee 1/3 as a minimum compression ratio. However, this method suffers from visual quality degradation in partial images such as with very detailed texture.

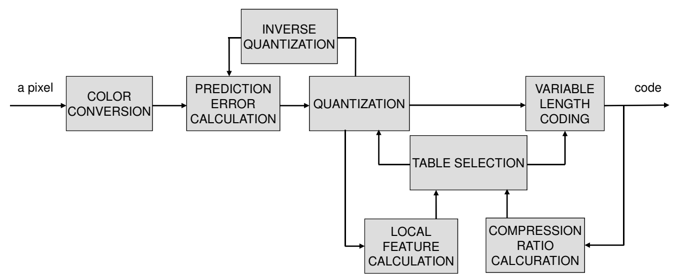


Fig. 1. Compression flow of the proposed method.

In this paper, we improve the method proposed in [3] by utilizing local feature of images. Specifically, each partial image is categorized in $Rough$, $Normal$, and $Flat$, regions, and then based on this classification data amount for pixels (bpp; bits per pixel) is finely controlled. Experimental results show that it achieves adequate bit allocation and improves image quality.

II. Proposed Method

A. Image Compression Flow

The proposed method compresses image data using local feature of image, which guarantees a compression ratio of 1/3 and keeps visual quality of images. Fig. 1 shows its compression flow.

In Color conversion, RGB data are converted into the luminance component (Y) and chrominance components (U, V), with the lossless color space conversion adopted in JPEG2000[4]. Next, the value of a target pixel is predicted as that of its left adjacent pixel, and then its prediction error is quantized by using a nonlinear table.

B. Compression Ratio Calculation

In order to guarantee a compression ratio of images, a margin of bit length for a target compression ratio is calculated when the prediction error of each pixel is coded, which determines the state of the next pixel, st. When st is small, quantization is performed coarsely and its result

TABLE I

QUANTIZATION STATE SELECTION BASED ON BIT MARGIN.

Bit Margin	0∼1	2∼3	4∼7	8∼9	10∼
st	0	1	2	3	4

TABLE II

TABLE SELECTION BASED ON st'.

st'	0	1	2	3	4	5
Y Table	2		2		4	5
U,V Table	1	2		3		5
Max. Bit Length	8	9	10	12	13	15
Min. Bit Length	4	6	6	6	6	6

is assigned to a shorter code. When st is large, quantization is done finely and its result could be assigned to a code of longer than 8 bits, 1/3 of 24 bit of a pixel, which improves image quality.

C. *Local Feature Calculation*

In addition to the st control, we change the quantization table according to local feature of image to perform adequate quantization for each area. A given image is divided into $M \times N$ segments whose width and height are I pixels and J pixels, respectively. Then, the feature of a segment is defined as the average prediction error of luminance pixels in the segment as indicated in Eq. (1).

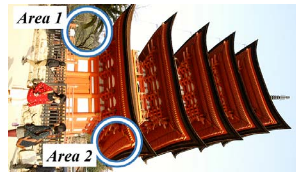
$$avg(m,n) = \frac{1}{I \times J} \sum_{b_j=0}^{J-1} \sum_{b_i=0}^{I-1} |sub_{m,n}(b_i, b_j)| \qquad (1)$$

Here, $avg(m,n)$ is the feature of the segment (m,n) and $sub_{m,n}(b_i, b_j)$ is the prediction error of a pixel (b_i, b_j) in the segment (m,n).

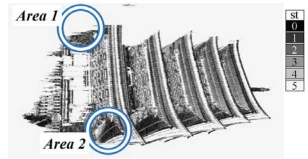
Each segment is categorized in *Rough*, *Normal*, and *Flat* with two threshold values. In *Rough* segment which has variety among pixels, large quantization errors tend to be inconspicuous so that its state considering the feature, st' is calculated as $st' = st - 1$ to perform quantization coarsely. Conversely, in *Flat* area which has little variety among pixels, even small quantization errors tend to be conspicuous so that st' is calculated as $st' = st + 1$ to perform quantization finely. Note that the feature of each segment is not coded in order to decrease code size. Each segment is categorized by the feature of its left and upper segment.

D. *Table Selection*

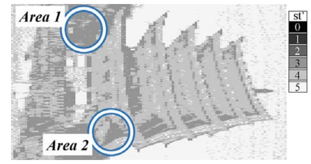
The luminance and chrominance component tables are determined with st'. TABLE II shows the allocation of the table and the maximum and minimum length of bits for each st'. Here, when both st equals 4 and the feature of the segment is *Flat*, st' is set to 5 in order to compress finely pixels that the prediction errors are small and quantization errors are conspicuous such as gradation area.



(a) original image (shrine).



(b) previous method[3].



(c) proposed method.

Fig. 2. Comparison of st and st' transitions.

III. EXPERIMENTAL RESULT

Fig. 2 shows transitions of st and st' as a result of the compression of the original image, 'shrine' with [3] and the proposed method. In the proposed method, segment size is set to $I = 16, J = 1$. As for *Area* 1, the upper left area of trees in Fig. 2, [3] utilize the fine st table of $st = 5$ because of large margin of bits, although deterioration of visual quality of image is inconspicuous in this area which has much variety among pixels. The proposed method categorizes this area into *Rough*, and utilizes the table of $st' = 3$ in order to get margin of bits. On the other hand, as for *Area* 2, the middle area behind the eaves of shrine in Fig. 2, [3] utilizes the extremely coarse table of $st = 0$. The proposed method utilizes the finer table of $st' = 3$.

IV. SUMMARY AND CONCLUSIONS

In this paper, we proposed an image compression method which guarantees a compression ratio and use local feature of images. Experiment results show that effective distribution of bits is achieved compared with [3] even in images which have variable density by considering efficiency of coding for local feature.

REFERENCES

[1] J.W. Hao, M.C. Hwang, S.G. Kim, T.H. You, S.J. Ko, "Vector quantizer based block truncation coding for color image compression in LCD overdrive," *IEEE Trans. Image Processing*, vol. 9, no. 29, pp. 1103-1127, Nov. 2000.

[2] Y. Jin, Y. Lee, and H.J. Lee, "A new frame memory compression algorithm with DPCM and VLC in a 4x4 block," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, pp. 1-18, 2009.

[3] T. Nakamae, A. Yamada, M. Yamaguchi, and T. Onoye, "Near-lossless image compression method using adaptive variable length coding," *ICESIT2012*, pp. 12-15, 2012.

[4] ISO/ISE 15444-1, "JPEG 2000 image coding system: Core coding system," 2004.