A STUDY ON SPATIAL SCALABLE CODING USING VECTOR REPRESENTATION

Yuki YAMAMOTO, Kei KAWAMURA, and Hiroshi WATANABE

Graduate School of Global Information and Telecommunication Studies,
Waseda University
1011, Okuboyama, Nishitomida, Honjyo, Saitama, Japan

ABSTRACT

The major advantage of vector representation of an image is that the image quality is maintained for arbitrary scaling. In recent years, a demand for scalable image coding has been increasing because of the wide variety of available digital contents and display terminals. Conventional scalable coding schemes are based on raster representation, and thus, line drawings deteriorate in quality when expanded and shrunk. In this paper, we propose an edge reconstruction method using vector representation for the purpose of keeping a consistent spatial scalability on transmission and display. We take an anti-aliasing into account in edge areas for approximation of luminance values around the edge. The proposed method can improve PSNR by up to 2 dB as compared to the conventional methods when image is expanded and shrunk.

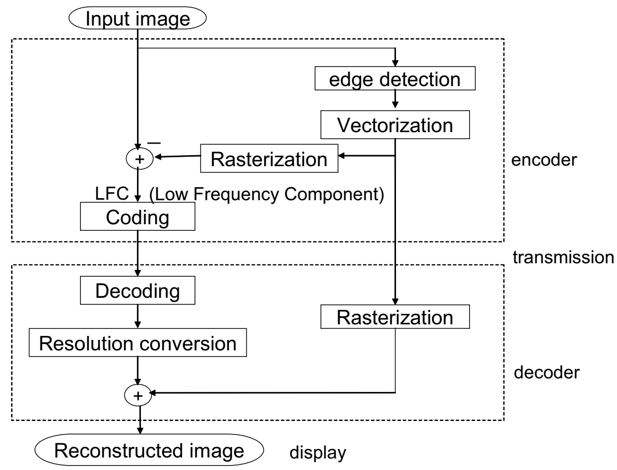


Fig. 1. Flow of proposed method

1. INTRODUCTION

The demand for scalable image coding has been increasing in recent years, since a wide variety of digital contents and display terminals have become available. The conventional image coding schemes are based on raster representation, and thus, line drawings deteriorate in quality when expanded and shrunk [1].

Vector representation, on the other hand, is able to maintain image quality for arbitrary scaling. When an edge component in an image is extracted and vectorized, spatial scalability can be improved and image quality can be maintained around the edges.

In this paper, we propose an edge reconstruction method using vector representation for the purpose of keeping consistent spatial scalability after transmission of an image to any display. For approximation of luminance values around the edge, we take an anti-aliasing into account in edge areas. The image can be displayed with any scale by the scalability of vector representation. We validate the proposed method by comparing PSNR values with the conventional method when images are expanded and shrunk.

In Section 2, we show the conventional methods for edge reconstruction. In Section 3, we describe the proposed method to keep the spatial scalability. Evaluation of experiment results are shown in Section 4, and the conclusions are given in Section 5.

2. EDGE RECONSTRUCTION

An edge reconstruction methods have been studied for object based coding in MPEG-4, and model-based edge reconstruction for wavelet-compressed images has been proposed in [2]. However, they cannot keep consistent spatial scalability after transmission of the image to any display. Thus, lack of line segment occurs when images are expanded and shrunk.

In addition, pseudo-vectorization scheme, which vectorizes the border line by quantizing the luminance value, has been proposed in [3]. However, the anti-aliasing process in edge areas has not been considered.

A vectorization method for edge components has been proposed as an animation coding scheme, which separates three components - line drawing, continuous area, and texture [1]. However, it is not suitable for anti-aliasing process since line drawing has only one luminance value. Thus, the luminance values around texture areas are lost.

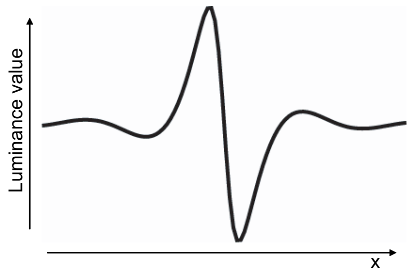


Fig. 2. Luminance value of edge component image

3. PROPOSED METHOD

Considering the anti-aliasing process in edge areas and consistent spatial scalability on transmissions and display, we propose an edge reconstruction method using vector representation. We separate an edge component from input image, and vectorize it. The flow of the proposed method is shown in Fig. 1.

3.1. Vectorization of the edge component

A high frequency component around the edge area is regarded as the edge component. The resulting luminance values of an edge component can be approximated by the curve shown in Fig. 2. This curve is represented by

$$g(x) = \frac{a\sin(bx + c)}{x^2 + dx + e}, \tag{1}$$

where a, b, c, d, e are parameters to be tuned optimally.

Next, the edge component is vectorized. Since vector representation has only one luminance value [4], a data of luminance values on vector representation is represented by $g(x, y)$ in two dimensions (x, y). The process of vectorization involves determining the values of the coefficients a, b, c, d, e. The necessary steps to acquire luminance values before the coefficients can be found are:

1. Thin the edge image.
2. Find an edge direction using Freeman's chain code[5].
3. Decide the direction by taking an average of pixel n.
4. Sum up the value of adjacent pixels which have same direction.
5. Find the luminance value with a width w perpendicular to the direction and average by each set,

where n and w are parameters to be tuned optimally.

These steps are shown in Fig. 3. Once luminance values are obtained, the coefficients of $g(x, y)$ can be found by using least-squares method.

Next, we convert $g(x, y)$ to $h(n)$ at vector location n.

Finally, using vector representation $V_H(n)$ and data of luminance values $h(n)$, we can vectorize edge component as shown in Fig. 4.

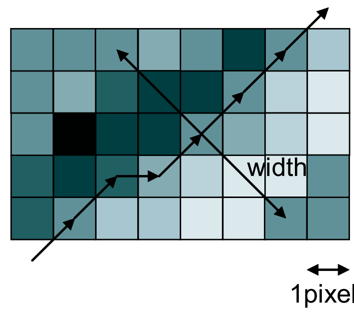


Fig. 3. Acquiring luminance values

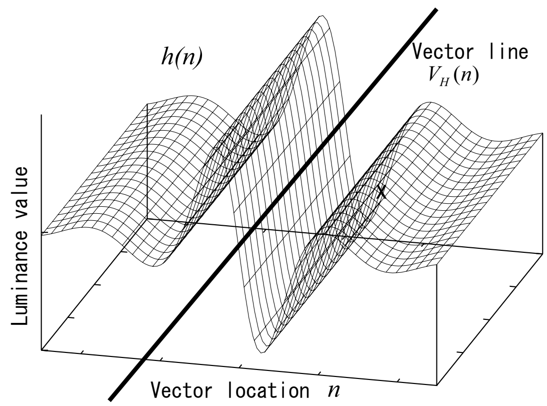


Fig. 4. Vectorization

3.2. Edge reconstruction

In the encoding process, the edge component is separated from input image $f(x, y)$ as shown in Fig. 1. Edge component $f_H(x, y)$ is described by

$$f_H(x, y) = \mathrm{Edge}\{f(x, y)\}, \tag{2}$$

where Edge$\{f\}$ is a function of edge detection from raster image f. Vector representation $V_H(n)$ is defined by

$$V_H(n) = \mathrm{Vectorize}\{f_H(x, y)\}, \tag{3}$$

where Vectorize$\{f\}$ is a function of vectorization of raster image f. The luminance values is approximated as $g(x, y)$, and $g(x, y)$ is converted to $h(n)$ on vector representation $V_H(n)$. This vectorized edge component is rasterized as

$$R_{1H}(x, y) = \mathrm{Rasterize}\{h(n)V_H(n), 1\}, \tag{4}$$

where Rasterize$\{hv, s\}$ is a function of rasterization using luminance value h, vector representation v, and scale of resolution conversion s. Then, low frequency component $f_L(x, y)$ is generated as

$$f_L(x, y) = f(x, y) - R_{1H}(x, y). \tag{5}$$

Table 1. Experiment conditions

Image size	width:334 hight:426[pixel]
Luminance value	0-255
Edge detection	Canny operator
	Length filtering : 30[pixel]
Vectorization	EPS(by Potrace[8])
	Length :n= 3[pixel]
	Width :w =5[pixel]
Lossless Coding	DPCM
Lossy Coding	JPEG2000

We compress $f_L(x, y)$ using the conventional coding method X, such as JPEG, and generate $Xf_L(x, y)$.

In the decoding process, a decoded $Xf_L(x, y)$ image is described by

$$X^{-1}Xf_L(x, y) = f_L(x, y) + n_c(x, y), \qquad (6)$$

where coding noise is defined as $n_c(x, y)$. When a low frequency component is converted to a small or large image with the scale of s, Rconv$\{f_L(x, y)+n_c(x, y), s\}$, where Rconv$\{f, s\}$ is a function of resolution conversion f with the scale of s, is generated. Vectorized edge component $V_H(n)$ is rasterized with the scale s. Then, a rasterized image $R_{sH}(x, y)$ is generated as

$$R_{sH}(x, y) = \text{Rasterize}\{h(n)V_H(n), s\}. \qquad (7)$$

Finally, a reconstructed image $f_R(x, y)$ is described by

$$f_R(x, y) = R_{sH}(x, y) + \text{Rconv}\{f_L(x, y) + n_c(x, y), s\}. \qquad (8)$$

When the vectorized edge component $V_H(n)$ is rasterized with the scale $s = 1$, a reconstructed image $f_R(x, y)$ is described by

$$\begin{aligned} f_R(x, y) &= R_{1H}(x, y) + f_L(x, y) + n_c(x, y) \\ &= R_{1H}(x, y) + f(x, y) - R_{1H}(x, y) + n_c(x, y) \qquad (9) \\ &= f(x, y) + n_c(x, y). \end{aligned}$$

In particular, when the coding method X is lossless coding scheme,

$$n_c(x, y) = 0. \qquad (10)$$

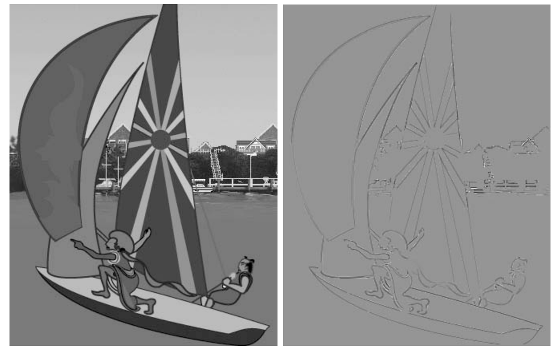
Then, $f_R(x, y)$ is described by

$$\begin{aligned} f_R(x, y) &= R_{1H}(x, y) + f_L(x, y) + 0 \\ &= f(x, y). \end{aligned} \qquad (11)$$

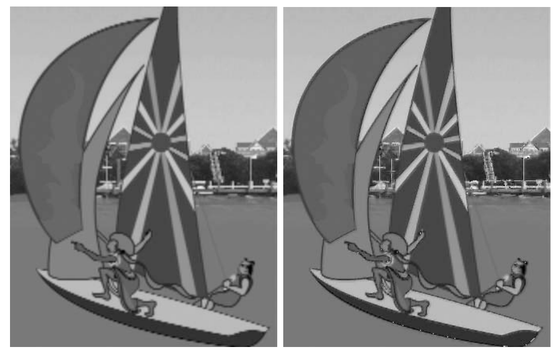
Thus, the lossless image can be reconstructed.

4. EXPERIMENT

We evaluate the proposed method by comparing bit rate and PSNR with the conventional method. Experiment conditions are shown in Table 1. An input image is a cartoon with precipitous edge as shown in Fig. 5(a). An edge area is detected by Canny operator [6, 7], and the edge component is converted into vector representation (EPS format) by Potrace [8]. The edge component is shown in Fig. 5(b) with luminance value of 128 added to the image. Since the EPS format does not include entropy coding, it is compressed by bzip2 [9]. Bzip2 is also applied to the data of luminance values.



(a) Input image (b) Edge component



(c) Expanded image (conventional method) (d) Expanded image (proposed method)

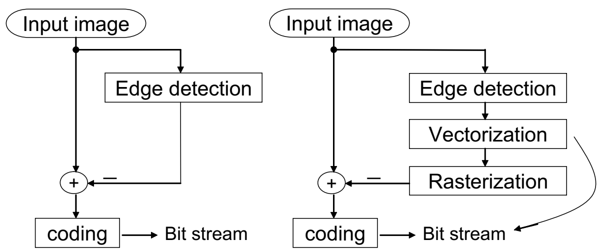
Fig. 5. Modified images

In this experiment, we validate the proposed method by comparing the modified image with the original image. Thus, the original image is shrunk to the half size in the encoding process. Then, the proposed method and the conventional method are applied to the shrunk image. In the decoding process, each image is expanded with $s = 2$. Each reconstructed image is evaluated by a bit rate and PSNR.

First, Input images are shrunk by 9/7 tap Daubecies filter. The coefficients of Daubecies filter are shown in Table 2.

Second, images are expanded by following three methods,

- nearest neighbor,
- bi-cubic,
- and bi-linear.



(a) Bit stream of the conventional method

(b) Bit stream of the proposed method

Fig. 6. Evaluation method of vectorization

Table 2. 9/7 Daubecies filter coefficients

k	Low Pass	High Pass
0	0.6029490182363579	1.115087052456994
+/ -1	0.2668641184428723	-0.5912717631142470
+/ -2	-0.07822326652898785	-0.05754352622849957
+/ -3	-0.01686411844287495	0.09127176311424948
+/ -4	0.026748741080976	-

Third, as considered the coding noise $n_c(x, y)$,

- lossless coding,
- and lossy coding,

is performed as compression schemes. The bit rate of conventional method with lossy coding is adjusted to the same value of the total bit rate of the proposed method in Fig. 6.

Last, we compare the bit-rate in Table 3 and PSNR in Table 4, 5 for each reconstructed image.

The proposed method provides better PSNR as compared to the conventional methods. Especially, PSNR is improved by up to about 2 dB by Bi-linear method. Refer to Table 4, 5, the proposed method is not affected by the coding noise $n_c(x, y)$. However, the proposed method with lossless coding does not reduce the bit rate sufficiently because of the misalignment of reference pixels.

Expanded images, which are shrunk by Bi-linear method and converted lossless, are shown in Fig. 5(c), 5(d) . The proposed method keeps the relations between width and length of line segments. Moreover, jaggy curves can be avoided.

Several other images are tested, and the similar results (up to 0.8–1.6dB) are obtained.

Table 3. Bit rate of each component

Component		Bit rate[bit/pel]	
		conventional	Proposed
Vector coding		-	0.38
Luminance value		-	0.91
Lossless coding	LFC	-	4.33
	total	4.34	5.61
Lossy coding	LFC	-	1.00
	total	2.29	2.29

Table 4. PSNR of reconstructed images (Lossless coding)

Operation	PSNR[dB]	
	Conventional method	Proposed method
Nearest neighbor	23.47	24.66
Bi-cubic	24.40	25.63
Bi-linear	24.43	26.54

Table 5. PSNR of reconstructed images (Lossy coding)

Operation	PSNR[dB]	
	Conventional method	Proposed method
Nearest neighbor	23.31	24.57
Bi-cubic	24.19	25.61
Bi-linear	24.23	26.26

5. CONCLUSION

In this paper, we proposed the edge reconstruction method using vector representation for keeping a consistent spatial scalability on transmission and display. Considering an anti-aliasing process in edge areas, we approximated luminance values around the edge. The proposed method could display images at any scale and provide better results than the conventional approach. In particular, our approach could improve PSNR by up to about 2dB as compared to conventional methods when the image was expanded and shrunk.

ACKOWLEDGMENT

We thank the Okawa Foundation for partial support of this research.

6. REFERENCES

[1] O. Nakagami, T. Miyazawa, H. Watanabe, H. Tominaga, "A Study on Two-layer Coding for Animation Images, " IEEE ICME2002, Vol. 1, pp. 26–29, Aug., 2002.

[2] Guoliang Fan, Wai-Kuen Cham, "Model-based Edge Reconstruction for Low Bit-rate Wavelet-compressed Images, " IEEE Trans. Circuits Syst. Video Technol., Vol. 10, No. 1, pp. 120–132, Feb., 2000.

[3] "Adobe Illustrator, " http://www.adobe.com/products/illustrator/newfeatures.html

[4] Adobe Developer Support, "Encapsulated PostScript File Format Specification, " May., 1992.

[5] H. Freeman, "Computer Processing of Line-Drawing Images, " Computing Surveys, Vol. 6, No. 1, pp. 57–97, 1974.

[6] J. Canny, "A Computational Approach to Edge Detection, " IEEE Trans. Pattern Anal. Machine Intell., Vol. 8, No. 6, pp. 679–698, Nov., 1986.

[7] K. Tombre, C. Ah-Soon, Ph. Dosch, G. Masini, S. Tabbone, "Stable and Robust Vectorization : How to Make the Right Choices, " GREC' 99, pp. 3–18, Sep., 2000.

[8] "Potrace, " http://potrace.sourceforge.net/

[9] "bzip2, ver.1.0.2, " http://sources.redhat.com/bzip2/