

ベクトル表現を用いた空間スケーラブル符号化に関する一検討

山本 勇樹[†] 河村 圭[†] 渡辺 裕[†]

[†] 早稲田大学大学院 国際情報通信研究科 〒 367-0035 埼玉県本庄市西富田大久保山 1011

E-mail: †{yuuki,kei}@tom.comm.waseda.ac.jp, †hiroshi.watanabe@waseda.jp

あらまし 近年、電子コンテンツとその閲覧端末の多様化により、スケーラビリティ機能を有するコンテンツ符号化の要求が高まっている。従来のスケーラブルビデオ符号化方式の大部分はラスタ表現を基本としており、拡大・縮小表示時の線の品質保持は十分でない。本稿では、伝送から表示までの一貫した空間スケーラビリティの確保を目的とし、ベクトル表現を用いたエッジの分離・再構成手法を検討する。特に、アンチエイリアシング処理が適用されたエッジ領域において、その輝度情報を近似する手法について提案する。提案手法により、多値のエッジ画像を再現することができ、任意の解像度での表示が可能となる。拡大・縮小表示時の画質評価を行い、有効性を確認する。

キーワード 画像符号化, ベクトル化, エッジ検出, スケーラビリティ, 解像度変換

A Study on Spatio Scalable Coding Using Vector Representation

Yuki YAMAMOTO[†], Kei KAWAMURA[†], and Hiroshi WATANABE[†]

[†] GITS, WASEDA University, 1011 Okuboyama, Nishitomida, Honjo, Saitama, 367-0035 JAPAN

E-mail: †{yuuki,kei}@tom.comm.waseda.ac.jp, †hiroshi.watanabe@waseda.jp

Abstract Vector representation has an advantage of keeping high image quality for arbitrary scaling. In recent years, a demand for scalable video coding has been increasing for the diversity of digital contents and display terminals. Conventional scalable video coding schemes are based on raster representation, and thus, line drawings deteriorate in quality when they are expanded and/or shrunk. In this paper, we propose an edge reconstruction method using vector representation for the purpose of keeping a consistent temporal scalability on transmission and display. We consider an anti-aliasing process in edge areas and approximate luminance values around the edge. We can display images at any scale by this method. Validity of the proposed methods is confirmed by quality assessment when the image is expanded and shrunk.

Key words Image coding, Vectorization, Edge detection, Scalability, Resolution conversion

1. はじめに

近年、電子コンテンツとその閲覧端末の多様化により、時空間スケーラビリティ機能を有するコンテンツ符号化の要求が高まっている。従来のスケーラブルビデオ符号化方式の大部分はラスタ表現であり、伝送時のスケーラビリティは確保されるが、表示時までの一貫した空間スケーラビリティの確保が十分でない。このため、セル

アニメーションやCG、テロップなどの人工的な映像を符号化すると、拡大処理によりジャギーが、縮小処理により線の欠落が生じる [1] [2]。線の欠落は主に主観品質に影響するが、近年 PSNR と比較して、より主観品質に近い評価手法として、線の品質に重みをおいた手法の提案もなされている [3]。

一方、ベクトル表現は、空間スケーラビリティに優れ、エッジ成分の符号化に用いることで、拡大・縮小表示時

の線の品質保持が可能となる．また，オブジェクト単位の構造化を行い，ベクトル表現の通過点を用いてフレーム間の相関をとることで，時間スケーラビリティの確保も可能となると考えられる．

本稿では，伝送から表示までの一貫した空間スケーラビリティの確保を目的とし，ベクトル表現を用いたエッジの分離・再構成手法を検討する．ベクトル化はアンチエイリアシング処理が適用されたエッジ領域に対して行う．輝度情報を曲線で近似する手法を提案し，通常一本の線に対して一つの輝度しか持つことができないベクトル表現を，複数の輝度情報を持つように拡張する．

ベクトル情報として位置と輝度値をもち，輝度値を近似した曲線を利用することで，輝度値の再計算が容易になり，任意の解像度変換が可能となる．また，輝度の近似曲線における係数を変化させ，表示する解像度に最適な曲線の幅と振幅を与えることで，拡大・縮小表示時のエッジの保持ならびに強調が可能となる．提案手法のスケーラビリティを利用することにより，拡大後の画像は，従来手法のスケーラビリティ機能を利用した場合と比較して約 2dB 向上することが確認できた．

2. 従来手法と問題点

2.1 エッジの分離・再構成

エッジの分離・再構成に関する従来研究としては，一枚の画像を任意形状のオブジェクトに分割する研究が広く行われ，MPEG-4 などで導入されている．またエッジの復元に関する研究として，wavelet 変換後の画像に対してエッジモデルを導入した研究がある [4]．これらの研究では，時空間スケーラビリティの確保はなされず，縮小表示時の線の欠落への対応は困難である．

2.2 エッジ成分のベクトル化

エッジ成分のベクトル化手法としては，アニメーションに特化した符号化方式として，線画と均等色領域，背景に分割して符号化する研究がある [1]．また，自然画に対しては，ポスタリゼーションを行うことで階調を量子化し，その輝度の境界線をベクトル化する擬似ベクトル化 [5] が挙げられる．これらの手法は，エイリアシング処理による輝度変化に対する考慮が十分でない．また，階調の量子化を伴うため，テクスチャなどの輝度情報が失われやすいという問題点がある．

2.3 解像度変換

従来の SVC を始めとするスケーラブルビデオ符号化は，ワンソースマルチユースを考慮して，伝送時のスケーラビリティ確保が重視されている．しかし，表示する際には，それぞれの閲覧端末の画面に応じた画像解像度変換が必須である．例えば，ある原画像を SVC で符号化

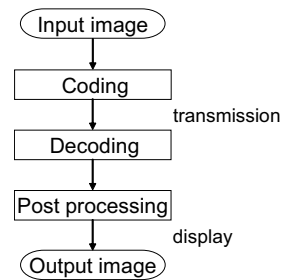


図 1 A flow of conventional method

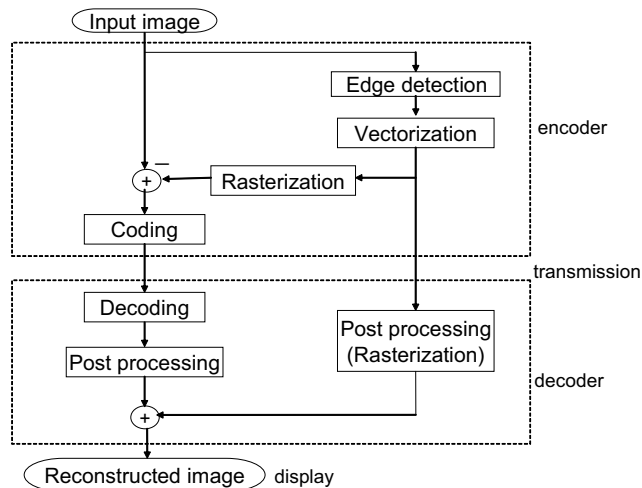


図 2 A flow of proposed method

後，1/2 の解像度で伝送することを考える．この画像を，表示端末にあわせて 1.2 倍に拡大する場合には，原画像を直接 0.6 倍に縮小する場合と比較して，品質を十分に維持することができないという問題がある．つまり，図 1 に示すように，表示までの一貫した空間スケーラビリティの確保が十分でなく，拡大・縮小表示時に線の品質が低下する．

3. 提案手法

3.1 エッジの分離・再構成

本節では，エッジ領域のエイリアシングを考慮した，ベクトル表現を用いるエッジの再構成手法を提案する．まず，本手法におけるエッジの分離・再構成の処理フローを図 2 に示す．入力画像をエッジ画像と，その差分画像（主に低周波成分）に分離する．それぞれ最適な符号化を行った後に伝送し，デコーダ側で再構成を行う．ベクトル化を行うエッジ画像は，エイリアシングを考慮するために，抽出したエッジ領域に，その画素に対応する高周波成分の輝度値を加えた画像とする．

入力画像を $f(x, y)$ とすると，エッジ画像 $f_H(x, y)$ は，

$$f_H(x, y) = \text{Edge}\{f(x, y)\} \quad (1)$$

と表せる． $\text{Edge}\{f\}$ はラスタ画像 f からエッジ画像を抽出する関数とする．また，ベクトル情報 $V_H(n)$ を，

$$V_H(n) = \text{Vectorize}\{f_H(x, y)\} \quad (2)$$

と表す。Vectorize{ f } は、ラスタ画像 f をベクトライズする関数とする。一方、Rasterize{ v, s } をベクトル画像 v をスケール s でラスタライズする関数とすると、ベクトル情報 $V_H(n)$ と、3.2 で述べる輝度の近似曲線の係数 $h(n)$ を利用して、ラスタライズ時に 1 倍の拡大・縮小処理を施したエッジ画像 $R_H(x, y)$ は、

$$R_H(x, y) = \text{Rasterize}\{h(n)V_H(n), 1\} \quad (3)$$

となる。

低周波成分 $f_L(x, y)$ は、式 4 に示すように、輝度の近似曲線とベクトル情報によりラスタライズしたエッジ画像と、入力画像との差をとることで生成する。

$$f_L(x, y) = f(x, y) - R_H(x, y) \quad (4)$$

低周波成分は既存の手法により符号化 X を行い、 $Xf_L(x, y)$ を得る。 $Xf_L(x, y)$ と、エッジ画像の情報として、 $V_H(n)$ ならびに $h(n)$ をそれぞれ伝送する。

次にデコーダ側の処理として、まず拡大・縮小処理を行わない場合を述べる。受信側が持つ情報は、従来が信号処理で取得した信号系列であるのに対して、提案方式は信号系列ならびにベクトル情報となる。符号化ノイズを $n_c(x, y)$ とすると、低周波成分 $Xf_L(x, y)$ をデコードした画像

$$X^{-1}Xf_L(x, y) = f_L(x, y) + n_c(x, y) \quad (5)$$

とラスタライズしたエッジ画像 (式 3) との和をとることで再構成画像 $f_R(x, y)$

$$f_R(x, y) = R_H(x, y) + f_L(x, y) + n_c(x, y) \quad (6)$$

を得る。

以上の処理により、エッジ成分を分離し、それぞれ符号化し、デコーダ側で再構成することが可能となる。また、特に低周波成分に対してロスレス符号化を適用すると、

$$n_c(x, y) = 0 \quad (7)$$

であるので、

$$\begin{aligned} f_R(x, y) &= R_H(x, y) + f_L(x, y) + 0 \\ &= R_H(x, y) + f(x, y) - R_H(x, y) \\ &= f(x, y) \end{aligned} \quad (8)$$

より、可逆な分離・再構成が可能となる。

次に、この分離・再構成を実現するため、エイリアシングを考慮したエッジ情報の保持を目的としたエッジ成分のベクトル化手法を示す。

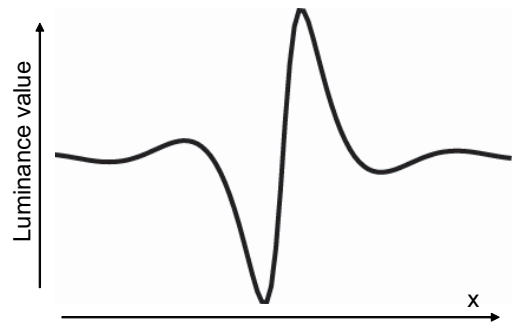


図 3 Luminance value of HP image

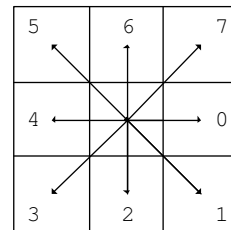


図 4 Freeman chaincode

3.2 エッジ成分のベクトル化

ベクトル化を行うエッジ画像は、主として高周波成分であり、輝度情報を持つ。しかし、ベクトル表現は、通常一本の線に対して一つの輝度しか持つことができない[6]。今回は、複数の輝度情報を持つように、ベクトル表現を拡張する。

一般的に境界線に対する高周波成分の輝度は一次元では図 3 のように表され、式 9 の曲線 $g(x)$ で近似できる。ベクトル方向に沿った輝度値の取得を行い、近似曲線で輝度値を表すことで、ベクトル表現時の輝度値の付加が可能となる。

$$g(x) = \frac{a \sin(bx + c)}{x^2 + dx + e} \quad (9)$$

近似曲線の係数取得は以下の手順で行う。

- (1) ベクトル化する多値のエッジ画像を細線化する
- (2) Freeman の chain code [7](図 4) を用いて、線に沿って方向を探索する
- (3) n 画素ごとに方向の平均をとる
- (4) 方向が同じ隣接集合をまとめる
- (5) 方向に対して直角の輝度を幅 w 画素で取得し、集合ごとに平均化する
- (6) 得られた輝度値を用いて、最小二乗法により係数を取得する。

この操作の様子を図 5 に示す。

デコーダ側でラスタライズする際、この手法で取得した係数を利用して、輝度値をベクトル情報に付加することで、図 6 に示すように多値のエッジ画像の再現が可能となる。

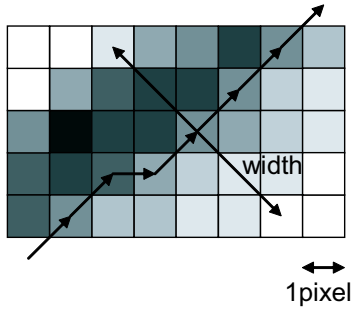


図5 Acquirement of Luminance values

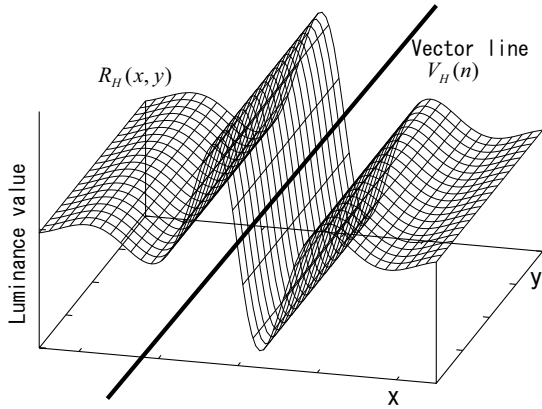


図6 Rasterization

3.3 解像度変換

次に、近似曲線の係数を利用して、デコーダ側で任意の解像度変換を行う手法を示す。\$s\$ 倍の拡大・縮小処理を行う場合、ラスタライズ後のエッジ画像 \$R_{sH}(x, y)\$ は、伝送後のベクトル情報 \$V_H(n)\$ を用いて、

$$R_{sH}(x, y) = \text{Rasterize}\{h(n)V_H(n), s\} \quad (10)$$

と表せる。また、低周波成分に対しては、\$\text{rconv}\{f_L(x, y) + n_c(x, y), s\}\$ の処理を行う。\$\text{rconv}\{f, s\}\$ は、ラスター画像 \$f\$ をスケール \$s\$ で解像度変換する関数とする。エッジ画像は、ベクトル画像をラスタライズする際に、目標とする解像度にあわせた輝度を、3.2 で求めた近似曲線を使って画素ごとに求めることで生成する。低周波成分に対しては、デコード後に解像度変換を施す。

以上の処理により、再構成画像 \$f_R(x, y)\$

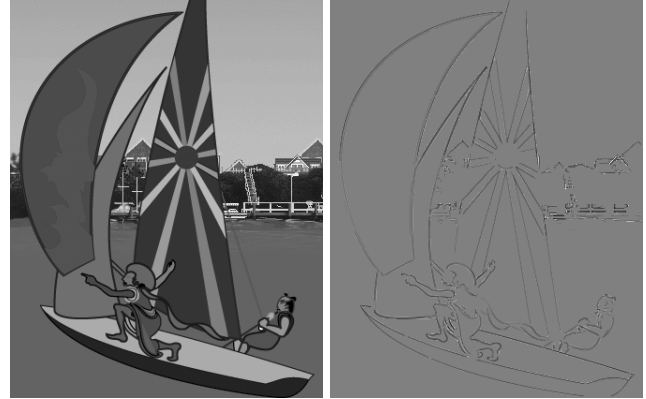
$$f_R(x, y) = R_{sH}(x, y) + \text{rconv}\{f_L(x, y) + n_c(x, y), s\} \quad (11)$$

を得る。

また、ラスタライズ時に、近似曲線の係数を操作し、表示する解像度に最適な曲線の幅と振幅を与えることで、拡大・縮小表示時のエッジの保持ならびに強調を行うことも可能である。このとき、エッジ画像 \$R_{sH}(x, y)\$ は、

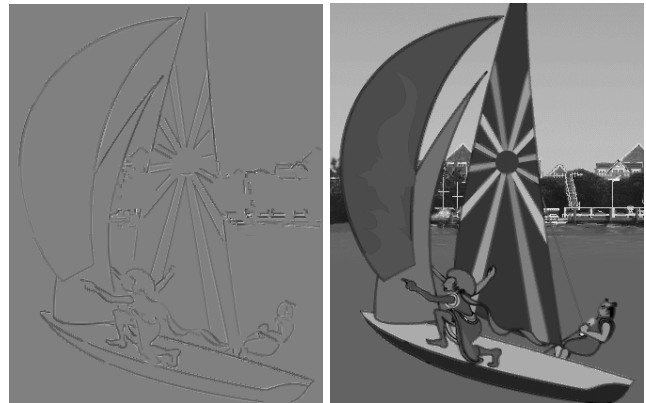
表1 Experiment condition

Image size	width:334 hight:426[pixel]
Liminance value	0 ~ 255
Edge detection	Canny operator
	Length filtering : 30[pixel]
Vectorization	EPS(by potrace [10])
	Length :n= 3[pixel]
	Width :w =5[pixel]
Lossless Coding	DPCM



(a) Input image

(b) Edge image



(c) Rasterized edge image(proposed method)

(d) LFC(Low Frequency Component) image:(a)-(c)

図7 Modified images

$$R_{sH}(x, y) = \text{Rasterize}\{h'(n)V_H(n), s\} \quad (12)$$

で与えられる。

4. 実験と考察

4.1 エッジの分離・再構成

提案手法を実装し、実験を行った。実験条件を表1に示す。入力画像は、図7(a)のように急峻なエッジを含むアニメーション画像を用いた。エッジ画像は、以下の手順によって求めた。

- (1) ハイパス成分の画像を求める
- (2) 入力画像に対して、canny operator [8] を用いて

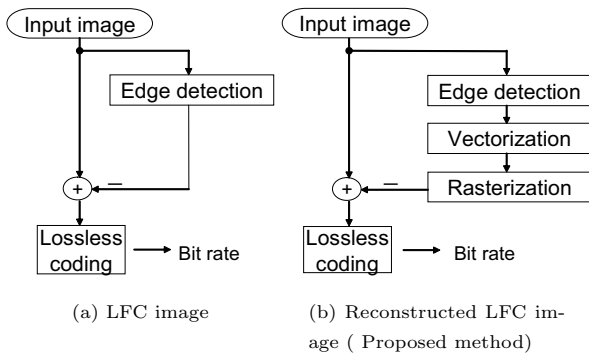


図 8 Evaluation method of vectorization

エッジ領域を抽出する

(3) 周囲長の短い画素集合を取り除く

(4) (3) で求めた画素領域に対応するハイパス成分をエッジ画像とする。

canny operator は、ガウシフィルタによる平滑化の後、微分オペレータを施すフィルタであり、ベクトル化の際のエッジ抽出手法として、最も良い手法の一つであるとされている [9]。

ベクトル表現はベクトル化ツール [10] を用いて EPS で符号化した。また、EPS はエントロピー符号化を行わないため、汎用のテキスト圧縮ツール [11] で圧縮した。ベクトル表現に付加された輝度情報も同様に bzip2 で圧縮した。

図 7 において、エッジ画像を (b) に、ラスターサイズ画像を (c) に、低周波成分の符号化前の画像を (d) に示す。ここで、図 7(b), (c) のエッジ画像は、それぞれ輝度値 128 を付加したものを表示している。図 7 より、ベクトル画像をラスターサイズする際に、近似曲線の係数を与えることで、多値画像のベクトル化が可能であることがわかる。これにより、縮小・拡大表示時の輝度の計算が可能となり、空間スケーラビリティが確保できる。

また、低周波成分にロスレス符号化を適用した際には、再構成画像は入力画像と等しくなることも確認できた。

4.2 エッジ成分のベクトル化

多値で表されたエッジ画像のベクトル化における再現性を測るために、低周波成分のロスレス圧縮符号化時の符号量を比較した。測定方法を図 8 に示す。入力画像から、単純に抽出したエッジ画像を引いた差分画像と、エッジ画像をベクトル化しラスターサイズしたものを引いた差分画像の符号量を比較することで、再現性を示す。

各符号量を表 2 に示す。表 2 より、提案手法によって生成される低周波成分の符号量は、入力画像の符号量とほぼ同等であることがわかった。これは、ラスターサイズして得たエッジ成分画像は輝度を一画素ずつ与えるため、

表 2 Evaluation of reconstructed image

image	Bitrate[bit/pel]
Input image	4.34
LFC image	3.94
Reconstructed LFC image(proposed method)	4.33

表 3 Bit rate of edge images

Resolution	Bit rate[bit/pel]		
	Vector coding	Luminance value	Total
334x428			
668x856(x2.0)	0.65	0.44	1.09
334x428(x1.0)	0.38	0.91	1.29
167x214(x0.5)	0.17	0.22	0.39

参照画素が少しでもずれると、図 3 のような波形によって、ハイパス成分を強めるような輝度が与えられることが原因にあると考えられる。近似曲線の係数の与え方についてのさらなる検討を行い、再現性を高めていく必要がある。

4.3 解像度変換

4.3.1 エッジ成分の符号量

解像度を变化させた際のエッジ成分の符号量を測定した。エッジ成分の符号量は、ベクトル情報とその輝度情報の和であるが、ベクトル化を行う解像度によって変化する。つまり、

- ベクトル情報におけるパス数の変化
- ベクトル付加情報である輝度情報量の変化

が考えられる。解像度とエッジ成分の符号量の関係を表 3 に示す。なお、輝度を求める際の長さ、ならびに幅は、それぞれの解像度に応じて変換を行った。また、少数点以下に対しては四捨五入した。

表 3 より、解像度と符号量との関係は、非線形であることがわかる。また、この情報を使って、任意の解像度で画像を表示する際は、一般的に高い解像度で抽出したエッジ情報を使った方がより再現性は高いといえる。

4.3.2 解像度変換における画質評価

任意の解像度の画像を生成する際に、ベクトル情報を使った再構成手法が有効であることを示すために、従来法の信号の補間手法と比較した。原画像を伝送時に 1/2 に縮小することを仮定し、デコーダ側で再び 2 倍に拡大した際の品質を測定した。従来手法は、原画像を符号化し、伝送時に 1/2 に縮小した。一方、提案手法は、原画像からエッジ画像を引いた低周波成分を符号化し、伝送時に 1/2 に縮小した。デコーダ側では、従来手法、提案手法ともに、それぞれ二倍に拡大し、原画像との PSNR を測定した。縮小画像の生成には JPEG2000 で導入されている 9/7 タップの Daubechies Filter を用いた。Daubechies Filter の係数を表 4 に示す。

拡大処理には、

表 4 9/7 Daubechies 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	-

表 5 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

表 6 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

- ニアレストネイバー法
- バイキュービック法
- バイリニア法

を利用した。

実験は符号化ノイズの影響を考慮して二通り行った。

- ロスレス符号化
- ロッシー符号化

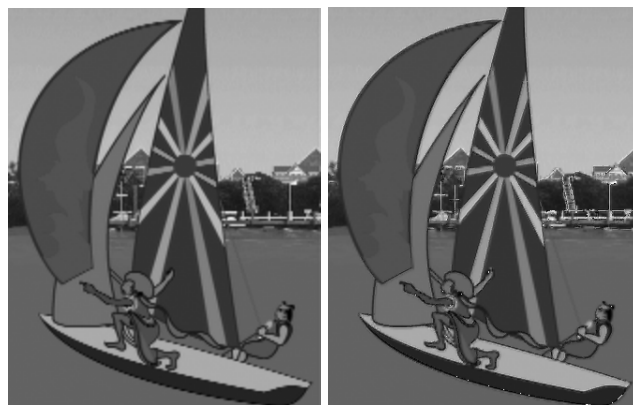
ロッシー符号化には JPEG2000 を用い, 1.0[bit/pel] で圧縮を行った。それぞれの PSNR を測定した結果を表 5, 6 に示す。

表 5, 6 より, 提案手法は従来手法と比較して, いずれも良い結果が得られた。特に, 拡大処理にバイリニア法を利用すると, 提案手法は従来手法と比べて約 2dB 向上することが確認できた。また, 符号化ノイズ $n_c(x, y)$ による再構成時の影響は特に見受けられないことがわかった。

ロスレス変換において, バイリニア法を用いた際のそれぞれの出力画像を図 9 に示す。

図 9 より, 従来手法では, エッジの部分にジャギーが見られ, 複雑なエッジに対しては線の欠落やぼけが生じているが, 提案手法ではいずれも線の品質保持が可能となっていることがわかった。

本実験は, 算出したエッジ輝度の近似曲線係数をそのまま利用し, 解像度での輝度を与えたが, 拡大・縮小表示時のエッジの保持ならびに強調を実現するためには, ラスタライズ時に, 表示する解像度に最適な曲線の幅と振幅を与える必要がある。



(a) Conventional method

(b) Proposed method

図 9 Expanded images

5. ま と め

本稿では, 伝送から表示までの一貫した空間スケラビリティの確保を目的とし, ベクトル表現を用いたロスレスのエッジの分離・再構成手法を検討した。特に, エイリアシングを考慮したエッジのベクトル化において, 輝度情報を曲線で近似する手法について提案した。本手法によって, 輝度を持ったエッジ画像を再現することができ, 任意の解像度での表示が可能となることがわかった。拡大・縮小表示時の評価を行い, 従来手法と比較して有効性を確認することができた。

文 献

- [1] Nakagami, O. Miyazawa, T. Watanabe, H. Tominaga, H. "A Study on Two-layer Coding for Animation Images," IEEE International Conference on ICME2002, Vol. 1, pp. 26-29, Aug. 2002.
- [2] Kawamura, K., Watanabe, H., Tominaga, H. "Vector Representation of Binary Images Containing Halftone Dots," 2004 IEEE International Conference on Multimedia and Expo. TP2-1. June. 2004.
- [3] ITU-T Recommendation J. 144: "Objective Perceptual Video Quality Measurement Techniques for Digital Cable Television in the Presence of a Full Reference," Jan. 2001.
- [4] 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.
- [5] "Adobe Illustrator 新機能," <http://www.adobe.co.jp/products/illustrator/newfeatures.html>
- [6] Adobe Developer Support, "Encapsulated PostScript File Format Specification," May. 1992.
- [7] H. Freeman, "Computer Processing of Line-Drawing Images," Computing Surveys, vol.6, no.1, pp.57-97, 1974.
- [8] Canny J, "A Computational Approach to Edge Detection," IEEE Trans. Pattern Anal. Machine Intell., vol. 8, No. 6, pp. 679-698, Nov. 1986.
- [9] K. Tombre, C. Ah-Soon, Ph. Dosch, G. Masini, and S. Tabbone, "Stable and Robust Vectorization :How to Make the Right Choices," GREC '99, pp. 3-18, Sep. 2000.
- [10] "Potrace," <http://potrace.sourceforge.net/>
- [11] "bzip2, ver.1.0.2." <http://sources.redhat.com/bzip2/>