

## ベクター変換における階層符号化の改良

河村 圭<sup>†</sup> 山本 勇樹<sup>††</sup> 渡辺 裕<sup>†</sup>

<sup>†</sup> 早稲田大学大学院 国際情報通信研究科 〒168-0051 東京都新宿区西早稲田 1-3-10 29-7 号館

<sup>††</sup> 早稲田大学理工学部 電子・情報通信学科 〒169-8555 東京都新宿区大久保 3-4-1

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

あらまし ディスプレイの解像度や表示端末の処理能力に適した品質や符号量にするため、ベクター変換には符号量制御や階層符号化が求められる。本稿では、ベクター変換においてオーバーヘッドをほとんど生じさせることなく階層符号化を実現する手法を提案する。提案手法は閉曲線を局所線分で近似し、線分の本数と近似誤差に対して最適な多角形を得る。角度交差法を用いて視覚的な印象を保存した荒い近似を行い、基本階層とする。荒い近似から品質を向上させるための通過点を拡張階層とする。先に提案した座標解像度の階層化手法と組み合わせることで、さらに粒度の小さい階層化が実現できる。シミュレーション実験により、直線・曲線の統合手法と角度交差法を組み合わせることで50%以上の通過点の削減、基本階層の画像の主観評価により提案手法の有効性を確認する。提案手法により、符号化効率の高いベクター表現の階層符号化が実現できる。

キーワード ベクター変換, 符号量制御, 階層符号化, 画像符号化

## Improvement of Hierarchical Coding for Vectorization

Kei KAWAMURA<sup>†</sup>, Yuki YAMAMOTO<sup>††</sup>, and Hiroshi WATANABE<sup>†</sup>

<sup>†</sup> Graduate School of Global Information and Telecommunication Studies, Waseda Univ.,  
29-7 Bldg., 1-3-10 Nishi-Waseda, Shinjuku-ku, Tokyo 169-0051 Japan

<sup>††</sup> Dept. of Elec., Info. and Comm. Eng., Waseda Univ.,  
3-4-1 Okubo, Shinjuku-ku, Tokyo 169-8555 Japan

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

**Abstract** Vectorization requires functions of a rate control and a hierarchical coding. In this paper, we propose hierarchical coding method without overhead at vectorization. The proposed method consists of three steps. Firstly, closed curves are converted to optimal polygons with local straight lines in term of line numbers and distortions. Secondly, we approximate optimal polygons to rough polygons as a base layer of vector representation preserving visual impression by local straight lines and their angular intersections. Thirdly, we extract pass points as enhanced layers from difference of optimal polygons and rough ones in order to improve a quality of approximation. Combining the proposed method and conventional method, i.e. hierarchy of coordinate resolution, we realize finer granular scalability coding than individually scalability. Validity of the proposed method is confirmed by experimental results. The proposed method realizes hierarchical coding of vector representation without sacrificing overhead.

**Key words** Vectorization, Rate control, Hierarchical coding, Image coding

## 1. はじめに

ベクター表現は端末の解像度に依らず高品質な画像を表示できるため、ベクター表現によるコンテンツの需要が高まると予想される。コンテンツを効率よく蓄積、配信、閲覧するためにはデータ圧縮が必須である。ディスプレイの解像度や表示端末の処理能力に適した品質や符号量にするために、ベクター変換には符号量制御及びスケーラビリティを実現する階層符号化が求められる。

動画のオブジェクト符号化の研究において、輪郭を記述するためにベクター表現を階層化する手法が研究されてきた。これらの手法はSD (Standard Definition) 画像を対象としており、解像度に対するスケーラビリティが十分ではない。また、多角形で近似しているため、最高品質においてもなめらかな輪郭線が得られないという問題がある [1]。

我々は、多角形近似及び曲線置き換えを組み合わせるベクター変換を提案している。許容誤差を可変にして直線・曲線を統合することで符号量制御を実現している。また、座標解像度の階層化によりオーバーヘッドをほとんど生じさせることなく階層符号化を実現している。しかし、符号量制御を応用した通過点の階層符号化はオーバーヘッドが大きく、改善が必要であった [2], [3]。

本稿では、通過点の再利用を可能とすることで、オーバーヘッドをほとんど生じさせずに階層化する手法を提案する。提案手法は、最高品質のベクター変換を行った後、角度交差法により視覚的印象を保存する荒い線分近似を行う。シミュレーション実験により、直線・曲線の統合手法と角度交差法を組み合わせることで50%以上の通過点の削減、基本階層の画像の主観評価により提案手法の有効性を確認する。提案手法により、柔軟な符号量制御とオーバーヘッドがほとんど生じない階層符号化が実現される。

## 2. 従来の階層化手法

### 2.1 直線・曲線の統合

本節では、直線・曲線の統合による従来の符号量制御について述べ、ベクター表現のファイル構造との関係を明らかにする。さらに、階層化によるオーバーヘッドを定量的に示す。

ラスタライズすることなく符号量を制御する手法として、直線・曲線の統合がある [2]。以下に、簡単な原理を述べる。最初に輪郭線を直線で近似し、線の本数と近似誤差についての最適多角形を得る。そして、多角形の頂点、または辺を適応的に滑らかな曲線に置き換えるが、長い曲線は複数の曲線に分割されている。次に、凸方向が同一の連続する曲線のみを統合して、符号量の削減を図る。

高解像度ビットマップではS/N比が悪化しているため、ノイズにより曲線の凹凸が頻繁に変化する。大局的には同一の曲線でも凸方向が変化するために統合できない。これらの条件を緩和することで、直線や凸方向の異なる曲線を含めて統合できる。統合により通過点が削減され、符号量削減が可能となる。

本手法は、ベクター表現のまま通過点を削減し、符号量を削減する。そのため、ラスタライズする必要がない。また、許容誤差に応じて統合される点がシーケンシャルに決定されるため、符号量制御が容易である。

### 2.2 直線・曲線の統合による階層構造

ベクター表現における1本の閉パス(閉じた曲線)は、表1のように絶対座標における開始点と、相対座標による通過点及び曲線の種類や制御点情報の繰り返しにより構成される。座標は入力解像度に対してスケーリングを施し、整数化されている。相対座標を用いるのは通過点同士の冗長性を除去するためである。付加情報は曲線の種類と膨らみ具合(制御点)を表し、これらもスケーリングにより整数化されている [4]。

直線・曲線の統合により、輪郭線の最適多角形の一部は図1のようになり、ファイル構造は表2のようになる。ここで、基本階層と拡張階層における通過点の個数について検討する。基本階層では3点の座標が必要である。このうち2点は拡張階層と共通であり、残りの1点は拡張階層では利用されずに破棄される。拡張階層では、基本階層に加えて3点の座標が必要である。統合される一連の点について着目すると、階層化しない場合には2点必要であるのに対して、階層化する場合には最終的に5点必要である。つまり、階層化される通過点において符号量が2.5に増加すると考えられる。これがオーバーヘッドの原因である。

表 1 ベクター表現のファイル構造

Table 1 File structure of vector representation

| X   | Y     | Info.   | note                        |
|-----|-------|---------|-----------------------------|
| 0   | 70131 |         | Start (absolute) coordinate |
| 0   | -147  | i       | Relation coordinate         |
| 24  | 0     | 60 c    |                             |
| 0   | -15   | 7 c     |                             |
| -32 | -32   | 12 7 C  |                             |
| 33  | -33   | 20 15 C |                             |
| -25 | 0     | 12 c    |                             |
| 0   | -25   | 7 c     |                             |
| 32  | 18    | 13 c    |                             |
| 0   | -13   | 9 c     |                             |
| 39  | -18   | 11 c    |                             |
| ⋮   | ⋮     | ⋮       |                             |

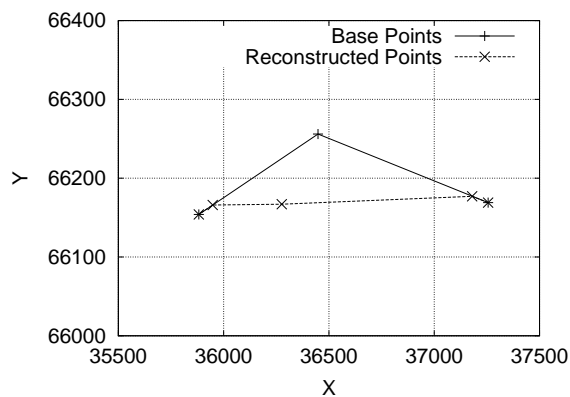


図 1 統合処理例

Fig.1 Integration sample

基本階層と拡張階層のファイルサイズを合計したときのレートディストーションを図 2 に示す。また、階層化しない場合のレートディストーションを合わせて示す。この図からも、階層化により符号量が最大で 1.5 倍程度に増加していることが分かる。

### 3. 視覚的印象を保存する階層符号化

#### 3.1 階層化の方針

本節では、基本階層の点を拡張階層で再利用することでオーバーヘッドを削減する点の階層化手法について検討する。

まず、基本階層はできるだけ符号量を抑え、大まかな内容が把握できれば良いと考えられる。そこで、直線のみを用いて多角形で表示することとする。また、基本階層で利用する通過点は、高品質な画像の通過点に包含されるという条件を追加する。

次に、近似誤差について検討する。高品質な画

表 2 統合前後の座標データと階層化の仕方

Table 2 Coordinate of both before and after integration and making hierarchical coords

| Base coordinate |       |          | Reconstructed coordinate |       |           |
|-----------------|-------|----------|--------------------------|-------|-----------|
| x               | y     | Info.    | x                        | y     | Info.     |
| ⋮               | ⋮     | ⋮        | ⋮                        | ⋮     | ⋮         |
| 37256           | 66169 | 10 c     | 37256                    | 66169 | 10 c      |
| 36448           | 66256 | 30 17 C  | 37180                    | 66177 | 30 17 C   |
| 35882           | 66154 | 533 61 C | 36276                    | 66167 | 528 451 C |
|                 |       |          | 35948                    | 66166 | 207 c     |
|                 |       |          | 35882                    | 66154 | 90 c      |
| ⋮               | ⋮     | ⋮        | ⋮                        | ⋮     | ⋮         |

| Base layer |      |          | Enhanced layer |     |           |
|------------|------|----------|----------------|-----|-----------|
| x          | y    | Info.    | x              | y   | Info.     |
| ⋮          | ⋮    | ⋮        |                |     |           |
| -9         | -14  | 10 c     |                |     | 1 1 3 E   |
| -808       | 87   | 30 17 C  | -76            | 8   | 30 17 C   |
| -566       | -102 | 533 61 C | -904           | -10 | 528 451 C |
|            |      |          | -328           | -1  | 207 c     |
| ⋮          | ⋮    | ⋮        |                |     | 90 c      |

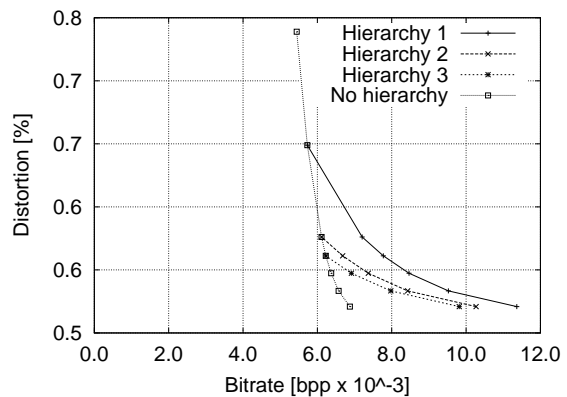


図 2 直線・曲線の統合による階層化のレートディストーション

Fig.2 Performance of hierarchy using integration.

像では原画像と近似画像間の位置ずれや方向ずれによる近似誤差の評価が最適であると考えられる。従来手法では、通過点における位置ずれを評価尺度としてきた。一方、低品質な画像では、原画像を見たときの印象と近似画像を見たときの印象が近いことを評価する。印象を評価尺度にすることで大幅な符号量の削減が可能となる [5]。

そこで、従来のベクター変換プログラムに基本階層を生成するための荒い近似処理を追加する。図 3

に従来のブロックダイアグラム，図 4 に提案するブロックダイアグラムを示す．荒い近似は最適多角形に対して適用する手法（実線）と，直線・曲線の統合後の多角形に対して適用する手法（点線）が考えられる．

荒い近似画像を基本階層とし，拡張階層は荒い近似画像と高品質画像の差分情報を割り当てる．差分情報には通過点と制御点を表す情報も含まれる．

### 3.2 角度交差法

荒い近似画像を得る手法として，改良角度交差法を提案する．角度交差法は，接線方向の変化分を近似の誤差評価に用いる手法である [6]．デジタル曲線では極限值としての接線は定義できないため，本稿では前工程で得られる多角形の辺を利用する．

多角形の辺と与えられる許容角から，図 5(a) の扇形で示されるような，各辺に対する許容方向範囲を求める．多角形上の 1 点を荒い近似における線分の始点とし，始点から順に許容方向範囲の積を求める．図 5(b) に示すように， $\angle a, \angle b, \angle c$  の 3 つの角度は積（Angular Intersection）を持つので，点  $O$  と点  $C$  は接続可能である． $\angle a, \angle b, \angle c, \angle d$  の 4 つの角度は積が空であるため，点  $O$  と点  $D$  は接続できない．

近似誤差は角度の積で評価する．角度の積が大きいほど良い近似であると考えられる．また，近似に必要な線の本数を最小にし，さらに近似誤差を最小にする近似図形を最適と定義する．

### 3.3 従来の角度交差法との差異

従来提案されていた角度交差法では，デジタル曲線における接線として局所線分を用いていた．局所線分とは，真直線をデジタル化したときに現れる点列パターンにより，デジタル曲線を分割したものである．また，近似線分の終点は，統合可能な接線群の作る最大の角の中央方向に等しい点とする．この手法は局所線分を分断するため，短線分の併合と直線に近い 2 直線の併合という 2 つの併合調整が必須である．

提案手法では，局所線分の代わりに前工程で得られた多角形を利用している．そして，始点と終点は多角形の頂点から選択している．その結果，荒い近似の頂点は多角形の頂点に包含され，階層化が容易に実現できる．また，グラフ理論を用いた最適な荒い近似を取得可能である．

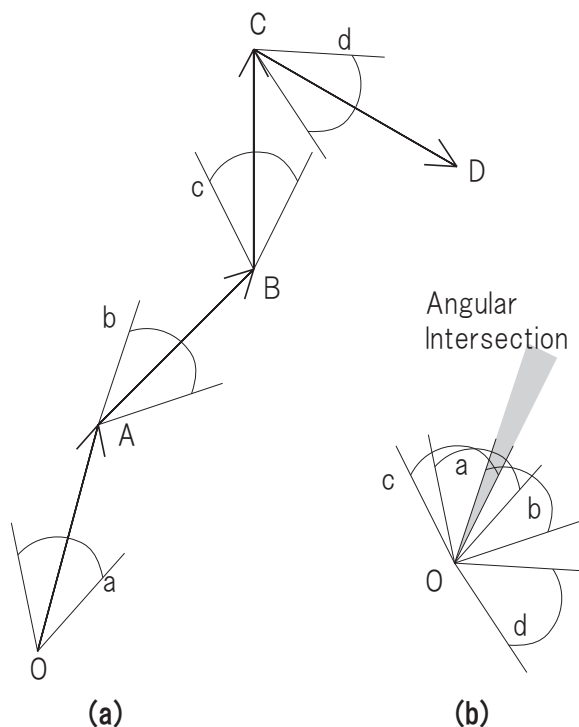


図 5 角度交差法

Fig. 5 Angular intersection method

## 4. シミュレーション実験

### 4.1 角度交差法による荒い近似

提案手法を実装し，シミュレーション実験を行った．入力画像は 1200dpi，2 値画像をスキャンした．許容誤差は 45 度とした．最適多角形を図 6 に，これを角度交差法により荒く近似した結果を図 7 に示す．また，直線・曲線の統合後に荒い近似をした結果を図 8 に，直線・曲線の統合のみを行った結果を図 9 に示す．

一連の図より，角度交差法を用いた荒い近似では視覚的な印象が保存されている．直線・曲線の統合を伴うと劣化が大きいが，基本階層としては十分な品質であると言える．

フォントに着目すると，イラストに比べて劣化が大きい印象を受ける．フォントの近似における要求品質は高いと考えられるので，文字内容を含めて基本階層で把握する必要がアプリケーションに対しては改善が必要である．イラストや文字があることが確認できればよいアプリケーションに対しては，現状で十分な品質が提供可能であることが確認できる．

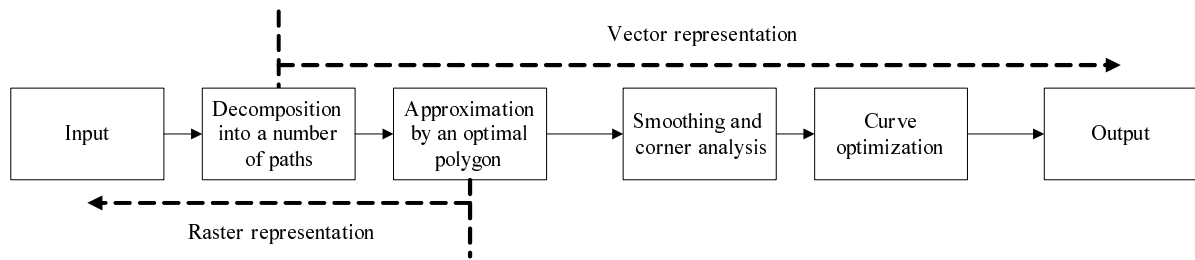


図 3 無階層のブロックダイアグラム

Fig.3 Block diagram without layer structure

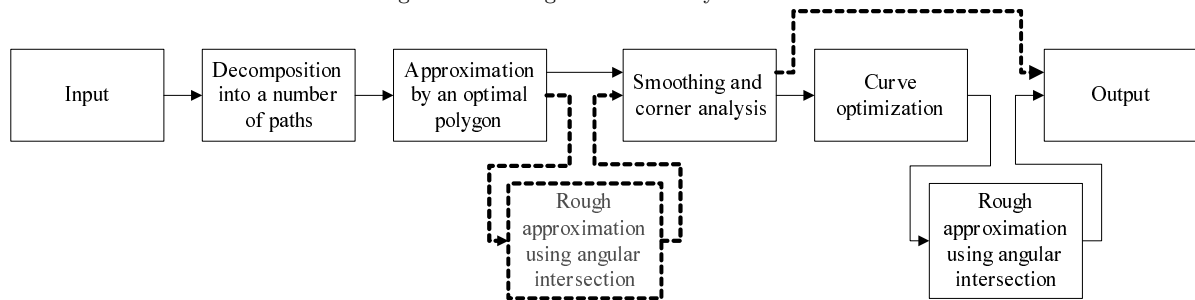


図 4 階層化ブロックダイアグラム

Fig.4 Block diagram with layer structure

#### 4.2 角度交差法と統合手法の比較

画像に含まれる輪郭線の本数と、それぞれの輪郭線を構成する通過点の数について評価する。最適多角形、直線・曲線統合手法、角度交差法、二つを組み合わせた場合の4通りについて評価を行う。

表3に構成する通過点の個数ごとに分類した輪郭線の本数、通過点の総数を示す。通過点の個数は、輪郭線の本数が概ね20%になり、さらに100より多い通過点を持つ輪郭線を長い輪郭線として分類した。

80%以上の輪郭線は16点以下の通過点により構成されており、100個を超える通過点をもつ輪郭線も3%弱存在することが分かる。また、直線・曲線の統合により通過点は25.7%削減され、角度交差法では42.4%が削減される。両者を組み合わせることによって51.3%の通過点が削減できることが分かる。

各種法による通過点の削減率を図10にしめす。輪郭線を構成する通過点の個数が3または4個ではほとんど削減できない。曲線の統合処理は通過点の個数が増えるほど削減率が低下するのに対して、角度交差法は通過点の個数が増えるほど削減率が向上する。両者を組み合わせることであらゆる輪郭線で40%以上の通過点を削減できる。

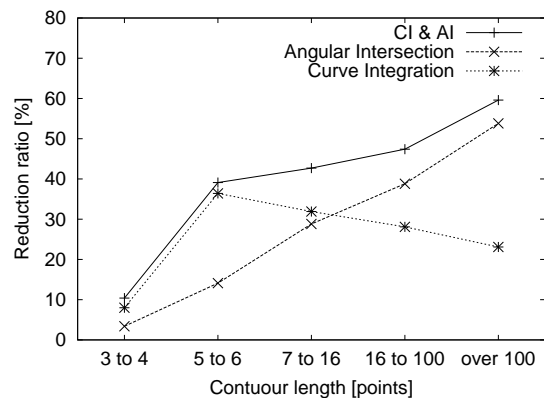


図 10 各種法による通過点の削減率

Fig.10 Reduction ratio of pass points

## 5. ま と め

本稿では、ベクター表現の階層符号化を実現する手法を提案した。従来手法ではオーバーヘッドが大きという問題があり、基本階層の情報を再利用しないことが原因であることを明らかにした。

基本階層は視覚的印象を保存する画像とし、角度交差法により基本階層を生成した。提案した角度交差法は従来手法と比較して併合調整が不要であり、画像を構成する線分の本数と接線方向の変化分において最適な画像が得られる。さらに、基本階層の通過点は入力画像の通過点に包含される

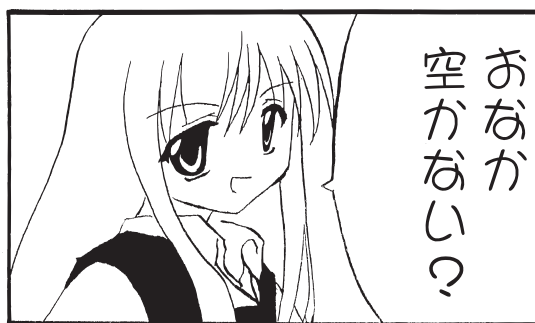


図 6 最適多角形画像  
Fig. 6 Image of an optimal polygon

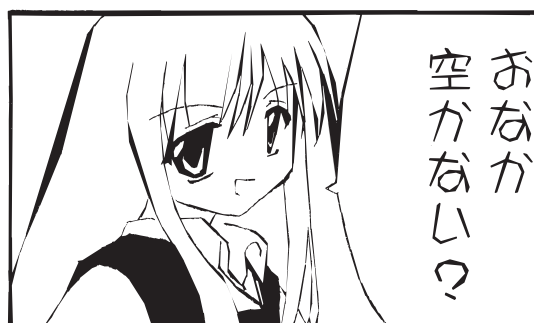


図 7 直線・曲線の統合前の荒い近似画像  
Fig. 7 Rough approximated image without integration

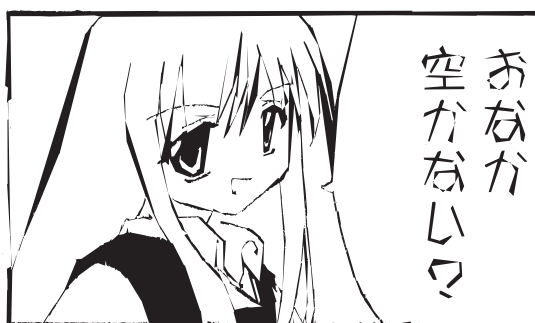


図 8 直線・曲線の統合後の荒い近似画像  
Fig. 8 Rough approximated image with integration



図 9 直線・曲線の統合画像  
Fig. 9 Integrated Image

表 3 画像に含まれる輪郭線の本数と平均長

Table 3 Numbers and average length of contours in an image

| Numbers / Average Length | Total | 3 to 4 | 5 to 6 | 7 to 16 | 16 to 100 | 100 over |
|--------------------------|-------|--------|--------|---------|-----------|----------|
| Contours                 | 7,811 | 1,855  | 2,343  | 2,115   | 1,278     | 220      |
| Optimal polygon          | 23.7  | 3.8    | 5.6    | 10.2    | 36.0      | 442.6    |
| Curve integration        | 17.6  | 3.5    | 3.5    | 7.0     | 25.9      | 340.3    |
| Angular intersection     | 13.6  | 3.7    | 4.8    | 7.3     | 22.0      | 204.7    |
| CI & AI                  | 11.6  | 3.4    | 3.4    | 5.9     | 18.9      | 178.6    |

ため、拡張階層を容易に生成できる。

生成された基本階層の主観評価を行い、実用に耐えうることを確認した。また、曲線の統合処理と角度交差法を組み合わせることで、通過点の削減率は50%以上であることをシミュレーション実験により確認した。

提案手法により、オーバーヘッドを生じさせることなく階層符号化が実現可能となった。また、解像度の階層化と組み合わせることで、粒度の小さいスケラビリティを実現できる。

#### 文 献

- [1] K.J. O'Connell, "Object-Adaptive Vertex-Based Shape Coding Method," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 7, No. 1, Feb. 1997.
- [2] 河村, 渡辺, "ベクター表現における符号量制御に関する検討," 画像符号化シンポジウム資料 19th, P-5.04, pp. 67-68, Nov. 2004.
- [3] 河村, 山本, 渡辺, "ベクター表現の階層符号化に関する検討," 情報処理学会 オーディオビジュアル複合情報処理研究報告, 2004-AVM-47, no.23, Dec. 2004.
- [4] Peter Selinger, "Potrace: a polygon-based tracing algorithm," <http://potrace.sourceforge.net/potrace.pdf>, 2003.
- [5] 青山, 河越, "視覚的印象を保存する線分表現法," 情報処理学会 グラフィックスと CAD 研究報告, 89-GCAD-39, no.8, Jul. 1989.
- [6] 青山, "局所線分と許容校偏差を用いた線分近似法," 電子情報通信学会秋季全国大会, D-369, 1990.