

ベクター表現による文字や網点を含む画像符号化システム

河村 圭[†] 山本 勇樹[†] 石井 大祐[†] 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 A310

E-mail: †kawamura@suou.waseda.jp

あらまし 文字や網点，線画を含む画像を解像度が異なるさまざまなディスプレイに表示する機会が増大している．異なる解像度を一致させるためには画像の解像度変換が必須である．我々はこれまでに，入力が容易なラスター表現を解像度変換に適したベクター表現に変換する手法を提案している．本稿ではコンテンツ需要の大きいマンガを画像として想定し，文字列，線画，網点に分離して符号化するシステムについて述べる．また，コマの分割線を用いた視線遷移や文字列の文字認識によりメタデータを付与する．システムのブロック図とそれぞれの要素技術について述べる．提案システムを実装し，PDA を用いて評価実験を行う．

キーワード ベクター表現，線画，文字，網点，画像符号化

Image Coding System based on Vector Representation with Characters and Halftone Dots

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

[†] Graduate School of Global Information and Telecommunication Studies, Waseda University, A310, 1011 Okuboyama, Nishi-Tomida, Honjo-shi, Saitama 367-0035, Japan.

E-mail: †kawamura@suou.waseda.jp

Abstract Images containing characters, line drawings, and halftone dots are displayed on many panels which has different resolutions. Resolution conversion of images is mandatory technique to adjust different resolutions. We have proposed to convert raster representation into vector representation. In this paper, we assume an input image is a comic image because it is in great demand for contents, and then we describe a comic image coding system. This system separates the input image into characters, line drawings, and halftone dots. We show a block diagram of the system and describe elemental techniques. Evaluation experiments are done by implementation for PDA.

Key words Vector representation, line drawings, character, halftone dots, image coding.

1. はじめに

読書専用端末のディスプレイは，読みやすさの観点から階調数は少ないが高精細である．一般的に日本語をきれいに表現するために 160dpi 以上のディスプレイが必要といわれている [1]．このように，文字や網点，線画を含む画像を解像度が異なるさまざまなディスプレイに表示する機会が増大している．

画像コンテンツを効率よく蓄積，配信，閲覧するため

には画像圧縮が必須である．既存の静止画像圧縮（符号化）方式としては，ファクシミリ向けの JBIG やデジタルカメラ向けの JPEG などが国際標準化されている．これらの符号化方式はラスター表現をベースとしているためにディスプレイの性能を十分引き出すことができず，解像度の異なる様々なディスプレイに対するスケラビリティも十分でない．

大部分のディスプレイはラスター表現により画像を表示するため，解像度変換により画像とディスプレイの解

像度を一致させる必要がある。我々はこれまでに、入力が容易なラスタ表示を解像度変換に適したベクター表現に変換する手法を提案している。ベクター表現は、画像を線、面、グラデーションなどの幾何的、数学的な要素を組み合わせて表現する方式である。自由な拡大・縮小処理が可能となるため、1ソースで様々な解像度の表示媒体へ高品質な表示が可能となる。

本稿ではコンテンツ需要の大きいマンガを画像として想定し、符号量が十分小さくて機能性の高いビットストリームを得ることを目的とする。マンガは主に線画、網点、文字、テキストチャ（細かい模様、網点を除く）により構成されているため、それぞれを分離して符号化する。また、コマの分割線を用いた視線遷移や文字列の文字認識によりメタデータを付与する。システムのブロック図とそれぞれの要素技術について述べる。提案システムを実装しを用いて評価実験を行う。

2. マンガ閲覧システムの従来手法

2.1 携帯コミック閲覧システム

携帯電話などの小さなディスプレイ（主に QVGA 未満の解像度）でコミックを閲覧するシステムとして、山田らは携帯コミック閲覧システムを提案している [2]。このシステムはコミックをフレーム（コマ）ごとに閲覧することを目的としている。フレームの閲覧順序はページをフレームに分割する線の傾きから決定される。

線画はアンチエイリアシング処理を考慮し、多値画像として符号化を行う。次に、文字列は連結黒画素の矩形を探索して文字切出しを行い、OCR により文字情報に変換する。そして、セリフが画面下部に携帯電話のフォントで表示される。閲覧ソフトウェアは携帯電話の Java により実装され、線画や文字はサーバからダウンロードされる。

本システムは複雑なコミックのフレーム順序が、分割線の傾きなどにより自動的に決定可能であることを明らかにした。しかし、網点や異なるディスプレイサイズが考慮されていないため、中間調の表現やスケーラビリティが不十分である。

2.2 ebi.j Book Reader

電子書籍販売サイトの 10DaysBook を運営するイーブックイニシアティブジャパンが開発した電子書籍のためのファイル形式である [1]。B6 サイズ程度の高解像度大画面を用いて、ページ単位で閲覧することを目的としている。書籍の版面を画像（ラスタ画像）として取り込んで表示するため、日本語の自由な組み版が実現可能である。また、マンガのような画像中心のコンテンツを扱うことができる。ファイルサイズが大きく、網点に起

因するノイズはぼかし処理で対応してると考えられる。

2.3 ComicSurfing

モバイル端末上でマンガの閲覧を可能にするビューアとそのシステムである [3]。携帯電話の小さなディスプレイで、フキダシ拡大や画像スクロールなどを実現している。専用ソフトウェアで作成された画像についてはベクター表現で蓄積や表示ができ、スキャンして得られた画像についてはラスタ表現のまま蓄積する。

本システムはコマ単位や視線遷移の順に拡大表示して、コミックが閲覧できることを明らかにした。しかし、ラスタ表現のまま符号化している場合がある。

3. マンガ符号化システム

3.1 マンガの画像特性

本節では、まずマンガの画像特性や符号化のアプローチについて述べ、次に提案する符号化システムの概要を示す。最後に、提案システムを実現する要素技術について言及する。

本稿で対象としているマンガは、特性の異なる様々な画像から構成されている。符号量が十分小さくて機能性の高いという観点で最適なビットストリームを得るために、それぞれを分離して信号特性と利用方法を考慮した符号化を行う。また、マンガはコマから構成されているが、コマの境界が明確でない場合やコマが複数のページに渡る場合がある。そのため、必ずしもページをコマに分割出来ない。

マンガの構成要素を挙げると、a) セリフなどを表すための文字列、b) キャラクターや背景を構成する線画、c) 中間調を表現する網点、d) 上記以外のテキストチャとなる。以下にそれぞれの要素について述べる。

3.1.1 文字列

文字列は、線画として扱うことも可能である。従って簡易的なシステムでは特に分けて考える必要はない。しかし、文字の歪みは目立ちやすく、画像インデキシングやセリフの多言語化を実現するためにも、文字列の分離は必要である。

人は活字を見慣れているため、文字においてはわずかな歪みでも目立ちやすいという傾向がある。そのため、同じ近似精度による文字と線画のベクター変換は効率が悪い。文字の歪みが目立たない精度で線画のベクター変換を行うと、線画の近似精度も向上するが、それに見合うだけの主観品質が向上しない。逆に、線画を基準に近似精度を設定して、文字のベクター変換を行うと歪みが目立ち、主観品質が著しく低下する。このような理由から、文字認識の有無にかかわらず、文字を分離することは、最適なビットストリームを得るために有効である。

画像インデキシングのアプリケーション例として、セリフによる検索を行う場合、少なくとも文字列を分離して文字情報が抽出されている必要がある。また、半自動的にコンテンツの多言語化を行う場合、あらかじめ文字列領域が抽出され、自動翻訳による下訳が行われていることで作業者の負担が著しく軽減される。

3.1.2 線画

線画は解像度変換（拡大処理）によりジャギーが発生する。そのため、ベクター表現に変換しておくことで、ディスプレイの持つ解像度で滑らかな線画が得られる。階調表現が可能な場合にはアンチエイリアシング処理により、さらに主観品質の高い線画が得られる。

線画はキャラクターや背景を構成しており、多くの場合それぞれの線幅は異なる。解像度が低い場合には線幅の差異は判別できないため、中心線によって符号化可能である。一方、解像度が十分に高い場合には線幅を考慮する必要があるため、中心線ではなく輪郭線（白と黒の境界線）による符号化が実用的である。このように、表示デバイスの解像度に対して、必要十分なベクター変換の精度が存在する。ベクター表現には解像度に関する品質スケラビリティ機能が必須である。

3.1.3 網点

網点とは、2値画像で中間調を擬似的に再現する印刷の手法である。広義にはテクスチャに包含されるが、解像度変換（縮小処理）によりモアレが発生するため、特に分けて考える必要がある。

単位面積あたりの黒領域の面積が濃度に相当する。また、点は格子状に配置されるため、周期的に濃度が変化する。点同士の距離は線数（1インチあたりの点の個数）で表される。縮小処理はダウンサンプリングと等価であり、網点の周期とサンプリング周波数が整数倍の関係がない場合、うねりという現象が生じる。これがモアレとして知覚され、主観品質を著しく低下させる。

3.1.4 テクスチャ

大部分のマンガは2値画像であるから、テクスチャも線画とみなしてベクター変換可能である。しかし、ベクター表現は網点以外の孤立点集合や細かい模様に対して、通過点が多数必要となり符号量が増加する傾向にある。

3.2 マンガ符号化システムのブロック図

入力画像には、容易にデジタル画像が取得できるスキャナを用いて得られた画像を想定する。このとき、300dpi以上の高解像度2値画像を仮定する。一方、すでに蓄積されているデジタル画像は1800×1200画素の多値画像が多い。原稿がB6サイズの場合150dpiに相当する。このような低解像度多値画像も入力画像として想定する。ただし、印刷物のように本来2値であった画

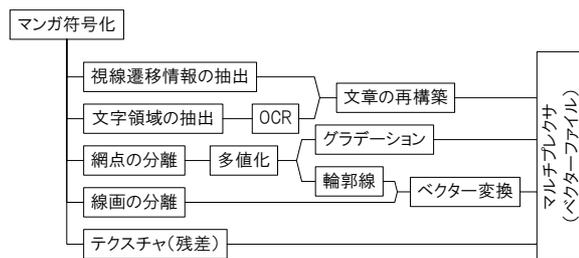


図1 マンガ符号化システムのブロック図

Fig.1 A block diagram of comic image coding system

像のみに限定する。自然画像のようなグレースケールやカラー画像は今後の課題である。

出力は、線画、網点、文字がベクター表現に変換され、既に広く普及しているベクターグラフィックスのプラットフォームであるSVGやFlash、Postscriptファイルとなる。また、テクスチャはラスター表現のまま符号化され、ベクターグラフィックスに含めることも可能である。同様に、変換されたテキストはメタデータとしてベクターグラフィックスに付与される。

マンガの構成をふまえ、信号特性と利用方法を考慮したマンガ符号化システムを図1に示す。マンガ符号化システムは、まず、網点の分離、線画の分離、文字領域の抽出から構成される。さらに、分離した網点はベクター表現に最適な形式として、領域とグラデーションに分けて表現する。網点領域の境界線と線画はそれぞれベクター変換によりベクター表現に変換される。抽出された文字領域は既存のOCRにより文字として認識され、後述する視線遷移情報を抽出し、文字から文章を再構築する。ただし、線画の分離手法は十分確立されておらず、本稿では網点以外をすべて線画として扱う。

3.3 各要素技術

3.3.1 網点分離

網点分離手法は入力画像の解像度と網点の線数の比によって適用する手法を変える。2値化したときに網点間隔が H_{th} 画素以上存在するときには、我々の提案している2値画像の網点分離手法を適用する[4]。網点間隔が H_{th} 画素未満かつ多値画像の場合には、大内らの提案する像域分離手法を適用する[5]。また、網点間隔が H_{th} 画素未満かつ2値画像の場合には、中間調として網点は失われているため一連の分離処理を適用しない。

3.3.2 小数画素精度の2値化

低解像度多値画像を適切なしきい値で2値化しても、線の連続性や異なる線幅が失われる。我々は小数画素精度で2値化する手法を提案している[6]。入力された低解像度画像をフィルタにより拡大し、Niblack法により2値化する。

3.3.3 ベクター変換

輝度値の境界を輪郭線としてベクター表現に変換する。2値画像の場合には白と黒の境界線をパスとして抽出する。線分とパスの距離が1/2画素以下という制約条件の下、線分の本数と誤差が最小になる多角形で近似を行う。多角形の辺または頂点をなめらかな曲線で置換してベクター表現を得る。改良角度交差法により、品質スケラビリティをもつデータ構造にも変換可能である [7]。

3.3.4 文字の抽出

既存の文字切出し手法では、線画と文字が混在している環境下で線画を文字と誤認識することが多く、十分な精度が得られない。本稿では、線画中の文字は活字であることに着目する。

まず、連結黒画素の外接矩形を用いて文字候補を切出す。次に、文字候補の集合を文字列として切出す。また、文字列長が短い場合は読みやすさのために周囲に余白があるため、孤立文字候補も文字列として切出す。

3.3.5 視線遷移情報

限られた画面サイズでマンガを閲覧するために、ページ内の視線遷移情報を抽出することを提案する。本手法はコマが画面サイズより大きくても有効な手法であり、陽にコマ分割が困難な場合にも破綻しない。

ページ内のコマ分割および整列手法として、下記の2つの手法が挙げられる。石井らはコマが直線で分割されることに着目し、エッジ方向の統計情報からコマ分割を行う手法を提案している [8]。また、山田らは分割後のコマおよびコマ内のセリフについて整列する手法を提案している [2]。

4. 実験と考察

提案システムを実装し、マンガをスキャンして符号化を行った。PDA を用いて表示している様子を図 2 に示す。

入力画像は B6 サイズの原稿を 300dpi でスキャンした 2 値画像である。出力は EPS を経由した Flash ファイルである。PDA のディスプレイは 320 × 480 画素 (約 140dpi) で、FlashPlayer を用いて表示している。

PDA のように携帯電話に比べて大きなディスプレイでは、ページ単位の閲覧も可能であることがわかる。逆に、これよりも小さなディスプレイではコマのようなページ未満の大きさで閲覧する必要がある。

現状では 1 ページ表示するのに約 1 秒かかるため、符号量を削減するなど Flash への最適化が必須である。ベクター表現のハードウェア高速化が標準化されつつあり、ハードとソフトの両面から高速化が可能である。



図 2 PDA による表示例

Fig. 2 An example of display by PDA

5. まとめ

本稿では、線画や文字、網点を含む画像を対象として、ベクター表現を基本とする画像符号化システムを提案した。それぞれの画像特性について述べ、システムをブロック図にまとめ、要素技術を示した。PDA の表示実験を行い、提案システムの実現性を明らかにした。

謝 辞

この研究は、財団法人大川情報通信基金研究助成による。

文 献

- [1] 高木, “電子書籍ビジネス調査報告書 2004,” 株式会社インプレス ネットビジネスカンパニー インターネット生活研究所, 2004.
- [2] M. Yamada, R. Budiarto, M. Endoh, and S. Miyazaki, “Comic image decomposition for reading comics on cellular phones,” IEICE Transaction on Information and Systems, vol.E87-D, no.6, pp.1370–1376, Jun. 2004.
- [3] CELSYS, <http://www.celsys.co.jp/solution/comic/>
- [4] K. Kawamura, H. Watanabe, and H. Tominaga, “Vector representation of binary images containing halftone dots,” 2004 IEEE ICME Proceedings., Jun. 2004.
- [5] 大内, 今尾, 山田, “文字/絵柄 (網点, 写真) 混在画像の像域分離方式,” 信学論 D-II, Vol.J75-D-II, No.1, pp.39–47, Jan. 1992.
- [6] 河村, 山本, 渡辺, “文字や網点を含む低解像度多値画像のベクター表現に関する検討,” FIT2006, J-032, Sep. 2006.
- [7] 河村, 山本, 渡辺, “ベクター表現の階層符号化に関する検討,” 情処研報 2004-AVM-47, no.23, Dec. 2004.
- [8] 石井, 河村, 渡辺, “分割線選択によるコミックのコマ分割処理に関する検討,” FIT2006, J-033, Sep. 2006.