

二値画像のベクトル化における符号量削減に関する検討 A study on rate reduction in vector representation of binary images

山本勇樹† 河村圭‡ 渡辺裕‡ 富永英義†‡
Yuki YAMAMOTO Kei KAWAMURA Hiroshi WATANABE Hideyoshi TOMINAGA

† 早稲田大学理工学部 電子・情報通信学科 ‡ 早稲田大学大学院 国際情報通信研究科
† Dept. of Elec.,Info. and Comm. Eng., Waseda Univ. ‡ Graduate School of GITS, Waseda Univ.

Abstract: Vector representation of binary images has an advantage of keeping image quality for arbitrary scaling as well as editing capability in units of objects. However, a conventional technique is not enough to achieve high compression efficiency compared with JBIG etc. In this paper, we show the main cause of coding inefficiency and propose how to improve it.

1. はじめに

近年、電子化されたコンテンツの要求が高まっており、従来の紙媒体の出版物を電子化する機会も増加している。過去に出版された漫画などを電子化する場合、原画が残っていないことが想定され、ノイズを含んだ印刷物をスキャナで取り込むことが多い。電子化後に配信する際にはコンテンツに適した形式で符号化することも必要となる。

ラスター表現からのベクトル化は一般的にラスター・ベクタ変換と呼ばれる。ラスター表現は、一般的にディスプレイの解像度に合わせた縮小・拡大処理には不適切である。一方、ベクトル表現は、拡大・縮小を行っても画質の維持ができ、表示するサイズや解像度に依存しない。また、オブジェクト単位の構造化により、コマ単位の表示などの機能性の付加が可能であり、2値画像のベクトル表現は有効な手法である。

本稿では、ラスター表現のベクトル化における符号化効率の向上を目的とし、その低下の主たる原因が直線や曲線上のノイズによる unnecessary 通過点であることを明らかにする。そして、これを削減する手法について検討する。

2. 現状と課題

2.1 予備実験

ラスター表現とベクトル表現の各符号化方式における符号化効率の検討が十分に行われていない。そこで、週刊漫画雑誌を実際に10枚分をスキャンし、ラスター表現とベクトル表現の符号化効率についてファイルサイズを比較する。実験条件を表1に示す。

ラスター表現は、MH, MR, MMR, JBIG, JBIG-2で符号化し、ベクトル表現は potrace[2] 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、汎用の圧縮方式である bzip2¹ で圧縮した。結果を図1に示す。横軸は符号量が小さい順に画像を並べたもので、縦軸はファイルサイズである。

2.2 考察

スキャンした画像に2値化処理のみを施してから符号化すると、ラスター表現の符号化方式の方がベクトル表現よりファイルサイズが小さくなる。この原因は大きく次の二つに分けられる。

- 網点
- ノイズ

Scanner	Canon Canoscan LIDE30
Source images	Weekly magazine
Paper size	B5
Scanning resolution	600dpi
Component/Bit depth	Gray scale/8bit
Binalize threshold	0.6 (0:white, 1:black)

¹LZ系のアルゴリズムと Burrows-Wheeler 変換(ブロックソート)を併用し、その上でハフマン符号化を行う圧縮方法

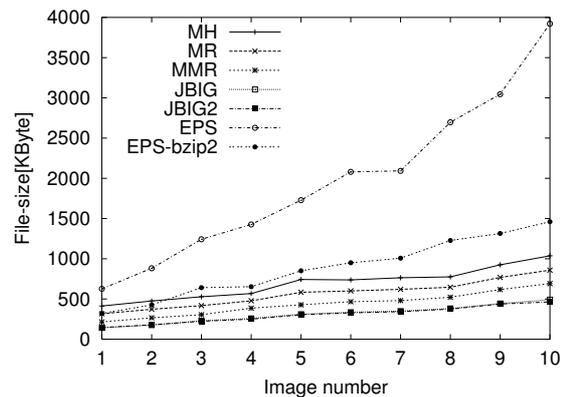


Figure 1 File-size of binary images

網点は、ベクトル表現に不適である[3]。また、入力画像には、漫画雑誌特有の印刷時のインクや紙の性質による多数のノイズが存在し、符号化効率低下の要因となっている。ここで、ノイズの種類はさらに以下の3つに分けられる。

- 孤立点として現れるノイズ
- べた塗り領域に現れるノイズ
- 直線・曲線上の凹凸の原因となるノイズ

特にベクトル表現の際には、直線・曲線上のノイズにより、図2(a)にみられるような多数の通過点が生じる。

2.3 符号化効率の改善

予備実験により明らかとなったノイズを除去するために、網点除去(HDS:half-tone dot separation)、孤立点除去とべた塗り再現(NRFA:noise reduction and fill in areas of a graphic)、境界線をぼかすガウスフィルタ(GAUSS:Gaussian filter)を施す。入力画像との差分を図2(b)に、ノイズ除去後のベクトル表現を図2(c)に示す。直線・曲線の平滑化によってノイズによる unnecessary 通過点を削減できた。さらにファイルサイズを測定し、結果を表2に示す。

以上より、符号化効率低下の主たる原因は、直線や曲線上のノイズによる unnecessary 通過点であることが明らかとなった。

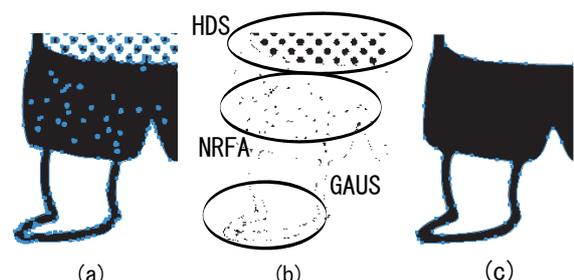


Figure 2 Pass points of vector images

Table 2 File-size after modified

Coding system	Operation	File size[KByte]
JBIG	-	141
MH	-	412
EPS&bzip2	-	320
EPS&bzip2	HDS	275
EPS&bzip2	HDS,NRFA	224
EPS&bzip2	HDS,NRFA,GAUSS	104

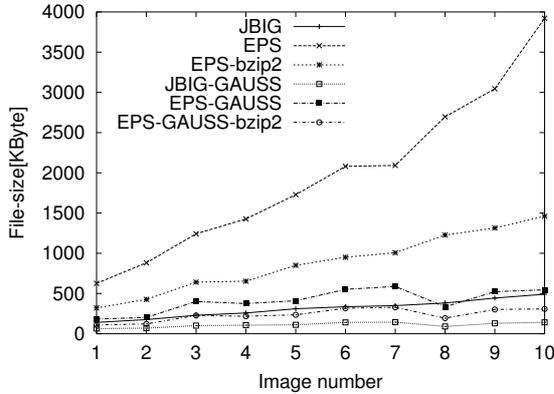


Figure 3 File-size by Gaussian filtering

2.4 ガウフィルタ

ガウフィルタは一般的に画像をぼかす効果があるため、網点とノイズの除去、線の平滑化を一度に行える。入力画像に半径2画素のガウフィルタをかけた場合のファイルサイズを図3に示す。

除去した網点はグラデーションなどベクトル表現で再現し、線画像にガウフィルタをかけてからベクトル表現に変換することで、JBIGと同程度の圧縮率を実現できることが明らかとなった。しかし、ガウフィルタの半径を大きくすると、図の細部がつぶれてしまうため、通過点削減手法としては適していない。ラスター画像のノイズを除去し、線の凹凸を減らす前処理が必要である。

3. 提案方式

ベクトル化の際に通過点増加の原因となる直線・曲線上のノイズを取り除くことを目的として、Freemanのchaincodeを利用した輪郭追跡を用いた手法を提案する。ノイズを含む輪郭を追い、その平均をとることで平滑化を行う。以下にアルゴリズムを示す。

1. 輪郭を追跡し、開始点からの差分コードを用いて、座標を縦軸・横軸それぞれ加算する。
2. 縦軸もしくは横軸の座標が設定したしきい値を越えれば、その区間に対して直線で置き換える。
3. 開始点をずらし、同様の処理を行う。
4. しきい値に満たない輪郭線に対しては、置き換えを行わない。

しきい値を6画素としたときの平均の取り方の一例を図4(a)に示す。ここで、正方形一つが一画素を、矢印が差分コードを示している。この例では横軸の座標が6画素の場合を取り上げたが、縦軸の場合も正負をとわず

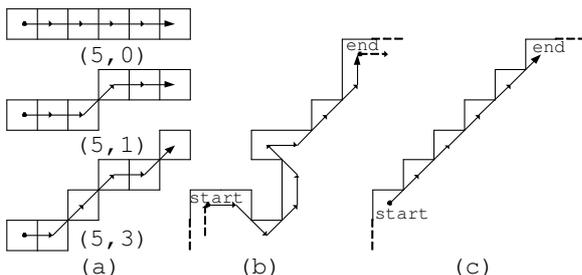


Figure 4 Proposed method

Table 3 File-size by proposed method

Images	File-size[KByte]
No operation	334
Proposed method	234
GAUSS(radius of 1pixel) & proposed method	207
GAUSS(radius of 1pixel)	242
GAUSS(radius of 2pixel)	196
GAUSS(radius of 3pixel)	180

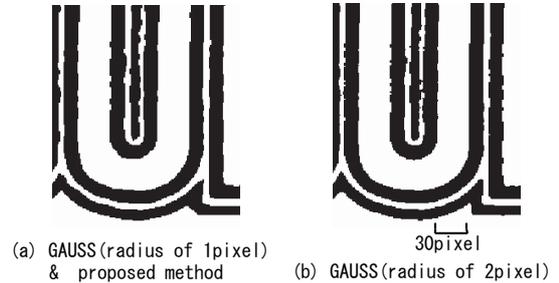


Figure 5 Result images by proposed method

同様に置き換えることができる。実際の処理では図4(b)の輪郭をもつ入力画像を、図4(c)のように平滑化する。

4. 実験と考察

提案手法を実装し、実験を行った。本実験ではしきい値を6とする。ある周囲長以下の点集合を除くことにより、網点とべた塗り領域のノイズを除去した画像を入力画像とする。ガウフィルタ、半径2画素のガウフィルタ後に提案手法を施した結果を表3に示す。

提案手法は、半径1画素のガウフィルタと比較して、ファイルサイズが小さい。

ガウフィルタは、半径を大きくすると細部がつぶれるため、小さい半径でのフィルタリングに限られる。つまり、通過点のある値より少なくできない。しかし、フィルタリングした画像に提案手法を適用することで、さらにファイルサイズを削減できる。表3において、半径1画素のガウフィルタと提案手法による処理を行った画像は、半径2画素のガウフィルタをかけた画像とほぼ同等のファイルサイズとなった。

それぞれの画像を図5に示す。図5(b)では所々で細部がつぶれるが、図5(a)ではつぶれずに、凹凸の削減のみが行えたことがわかる。以上より、提案手法は不必要な通過点の削減手法として有効であるといえる。

5. まとめ

本稿では、ベクトル化の際の符号化効率低下の要因となる直線・曲線上のノイズをラスター画像の段階で削減する手法について検討した。さらに、ガウフィルタと組み合わせることでさらなる削減が可能となることを示した。今後は、ファイルサイズと差分を用いた提案手法の評価方法について検討する。

参考文献

- [1] Donggang Yu et al, "An efficient algorithm for smoothing, linearization and detection of structural feature points of binary image contours," IEEE Pattern Recogn Vol.30, No1, pp.57-69, Jun.1997.
- [2] "Potrace," <http://potrace.sourceforge.net/>
- [3] 河村, 渡辺, 富永, "網点を含んだ2値画像のベクトル表現に関する検討," 画像符号化シンポジウム PCSJ2003, P-2.13, Nov.2003.