

J-28

空間分割モデルデータの圧縮に関する一手法

Data Compression for Spatial Partitioning Representations

大久保 宏† Hiroshi Ohkubo 手島 裕詞† Yuji Teshima 西尾 孝治† Koji Nishio 小堀 研一† Ken-ichi Kobori

1. はじめに

近年、マルチメディアデータのやりとりが多くなっているが、画像データの伝送はネットワークに大きな負荷を与える。特に3次元画像ではデータ量が膨大となる。そこで、画像データ伝送時におけるネットワーク負荷を抑えるためのデータ圧縮技術が必要となっている。

ところで3次元モデル表現として、形状を2値の単位立方体の集合で表現するボクセルモデルがある。ボクセルモデルではデータ構造が単純であり、立体集合演算が容易であるなどの利点がある。しかし、形状を正確に表現しようとすると単位立方体の数が増加し、データ量が膨大になるという問題点がある。

ボクセルモデルのデータ圧縮手法として、オクトリー表現^[1]やランレングス符号化などがあるが、本研究では主に2次元画像処理の分野で用いられるモルフォロジー^[2]を2値のボクセルモデルに適用し画像圧縮を行う。また、圧縮で得られたデータに対して算術符号化^[3]を用いて更なる圧縮を行い、その効果についても考察する。

2. ボクセルモデルの圧縮

本研究では、モルフォロジーの基本演算であるミンコフスキー和、エロージョンを導入する。各演算では、入力画像と構造要素と呼ばれる画像を用いて行う。以下、対象は3次元画像であるが2次元で説明を行う。

図2.1に示すように、圧縮対象画像に対してミンコフスキー和、エロージョン、排他的論理和を繰り返し行い、スケルトンと呼ばれる画像を生成することで圧縮を行う。同図では S_1, S_2, S_3 の論理和をとった画像が結果のスケルトンとなる。ここで E_r はエロージョン、 $M+$ はミンコフスキー和を表す。

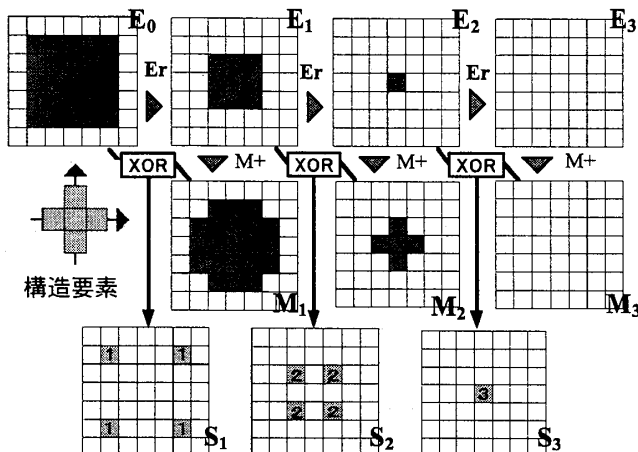


図2.1 スケルトン生成過程

また、結果のスケルトンに対して圧縮に用いた構造要素でミンコフスキー和と論理和を繰り返し行うことで復元を行う。その例を図2.2に示す。

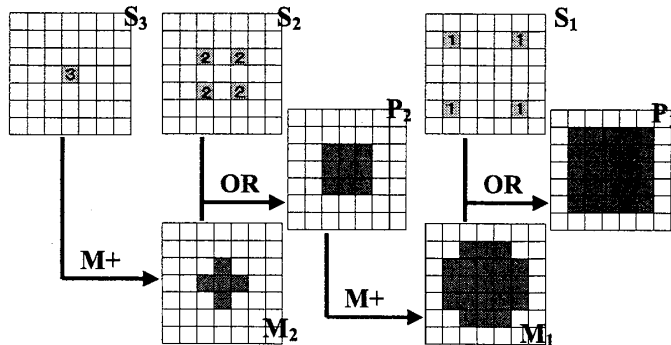


図2.2 復元過程

3. 分割圧縮

単一のボクセルに対して、ある構造要素を用いてミンコフスキー和のみを繰り返し行い生成された形状があるとき、その形状に対し、ミンコフスキー和に用いた構造要素で圧縮を行うと高い圧縮率を得ることができる。モルフォロジー演算を用いた圧縮では、このような特性がある。

分割圧縮では、この圧縮の特性を利用する。まず、圧縮対象画像をミンコフスキー和のみで生成できる複数の基本形状で表現する。次にそれぞれの基本形状に対して2章で述べた圧縮を行い、画像全体を圧縮する。分割圧縮では圧縮対象画像を表現する基本形状に立方体を用いる。

圧縮の手順は以下のようにになっている。

- 手順1: 図3(a)に示す圧縮対象画像に対して、各軸方向で最も長い黒画素列を探索する。その画素列を同図(b)のAに示す。
- 手順2: 手順1で探索した黒画素列に対して直交する最長の黒画素列を探索する。その画素列を同図(b)のBに示す。
- 手順3: 同図(b)に示すように手順1, 手順2で探索した2つの黒画素列の交点Cを求め、その交点Cを基点とし、圧縮対象画像の形状内部で正方形を膨張させ、内部に入る最大の正方形を求める。その様子を同図(c)に示す。
- 手順4: 手順3で得られた正方形に対して同図(g)に示す 2×2 の構造要素で圧縮を行い、スケルトンを生成する。得られたスケルトンを同図(d)に示す。また、圧縮対象画像から手順3で得られた正方形と重複する領域を背景画素に置き換える。その様子を同図(e)に示す。なお、図中の数字は2章で述べた圧縮でのスケルトンを得るまでの繰り返し処理の回数を各画素に付加したものである。

† 大阪工業大学 情報科学研究科

手順5：同様に、手順1から手順4までを繰り返し行うことで画像全体のスケルトンを得る。その結果を同図(f)に示す。なお、繰り返し処理の手順3で生成される正方形は以前に生成された正方形との重複を許すものとする。終了条件として、圧縮対象画像がすべて背景画素に置き換えられた時点で処理を終了する。

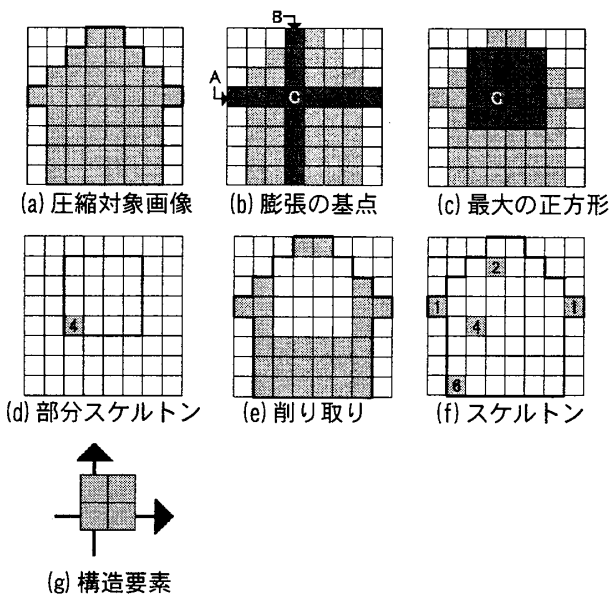


図3 スケルトン生成

4. 実験

ボクセルモデルの一般的なデータ圧縮手法であるオクトリー表現やランレングス符号化と、提案した分割圧縮法の圧縮率を比較するため、14種類の形状のボクセルモデルに対して各手法で圧縮を行い、ファイルに出力した際のデータ量の比較を行った。その結果を図4.1に示す。なお、グラフの縦軸は対数で表記している。

また、分割圧縮法、オクトリー表現、ランレングス符号化の各手法で得られた圧縮後のデータに対して算術符号化を用いて2次圧縮を行った。そのデータ量を図4.2に示す。

なお、グラフ中のLevelとは、空間の一边を 2^{Level} とした際の空間の解像度を示すものである。実験ではこのLevelを5~8まで変化させて行った。

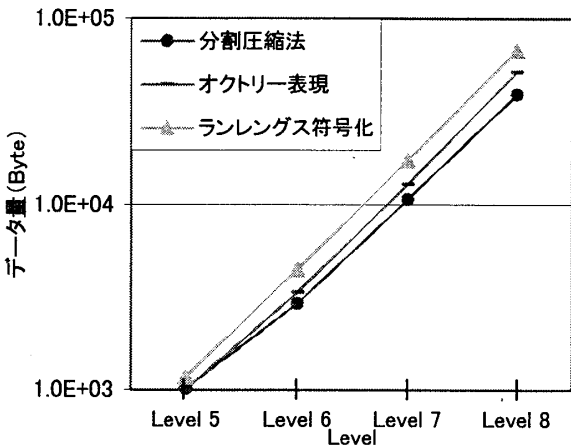


図4.1 平均データ量

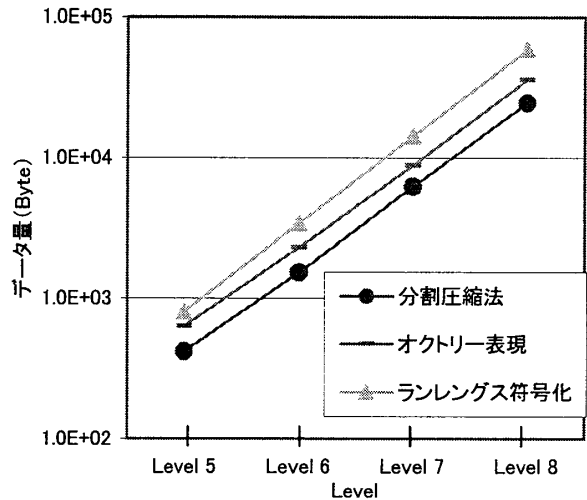


図4.2 算術符号化による平均データ量

分割圧縮法、オクトリー表現、ランレングス符号化の圧縮後のデータ量を比較したところ、図4.1に示すように分割圧縮法を用いて圧縮を行った際のデータ量がオクトリー表現やランレングス符号化での圧縮によるデータ量よりも少なくなっている。提案した分割圧縮法はオクトリー表現に比べて約75%、ランレングス符号化に比べて約60%のデータ量となっており、分割圧縮法の有効性を確認できる。

分割圧縮法、オクトリー表現、ランレングス符号化の圧縮後のデータに対して算術符号化を用いて圧縮を行ったところ、図4.2に示すように分割圧縮法のデータ量が、オクトリー表現やランレングス符号化のデータ量よりも少なくなっている。提案した分割圧縮法はオクトリー表現に比べて約65%、ランレングス符号化に比べて50%のデータ量となっている。このことから算術符号化を用いて2次圧縮を行ったとしても分割圧縮法がオクトリー表現やランレングス符号化よりも有効であるといえる。

5. おわりに

本研究では、モルフォロジー演算の特性を利用し、圧縮対象画像をミンコフスキー和のみで生成できる複数の基本形状で表現し、圧縮を行った。その結果をデータ量で比較した場合、オクトリー表現やランレングス表現よりも圧縮率が高くなることが確認できた。また、算術符号化を用いて更に圧縮した場合においても提案した手法が有効であることが確認できた。

今回、圧縮対象画像を表現する基本形状として立方体を用いたが、今後は立方体だけでなく更に良い圧縮率を得ることのできる基本形状を探すことが課題である。

6. 参考文献

[1] 中嶋正之 他：“技術編 CG標準テキストブック”，画像情報教育振興協会，pp87-88，(1998)
 [2] 小畑秀文：“モルフォロジー”，コロナ社，pp12-42，(1996)
 [3] M.ネルソン，J.L.ゲイリー：“データ圧縮ハンドブック”，プレントリスホール，トッパン，pp105-138，(1996)