

# ピクセル順番テーブルを用いた 3次元画像データの可逆符号化

3 E - 1

明上山 温      坂本 雄児      山本 強  
北海道大学工学部   室蘭工業大学   北海道大学大型計算機センター

## 1 はじめに

現在では幅広い分野でデジタル画像が用いられている。コンピュータの計算速度と表示能力の向上により、医療画像(CT,MRI画像)などのようにデータ容量が大きい画像を保存することも一般的になってきた。更に膨大なデータ容量である3次元画像データも保存の必要性が高くなった。我々は、このような1ピクセルあたりの容量が大きく、ピクセル数の多い2次元画像データに対して有効な符号化法として、後述するピクセル順番テーブルを用いた手法を提案した。[2]

本研究ではこのピクセル順番テーブルを3次元画像データに適用し、2次元画像データと比較した符号化効率の変化について考察した。

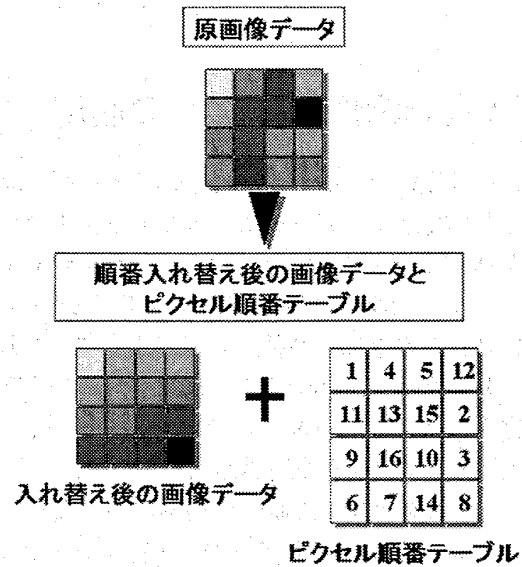


図1 ピクセル順番テーブルの概念

## 2 ピクセル順番テーブル

画像データを正方形ブロックに分割し、そのブロック内についてラスタ走査で差分符号化を行う場合、最も効果的な符号を生成するのはピクセル濃度値の変動が単調増加(又は減少)である場合である。これはブロック内のピクセルを濃度値が低い順に並べ替えることで実現できるが、そのとき同時に元に戻すためのテーブルを作成することが要求される。よって、原画像は図1のように分割され、このとき生成されるテーブルをピクセル順番テーブルと呼ぶことにする。このテーブルを画像内のブロック全部に行うと画像データに匹敵する容量となってしまうため、画像内で差分符号化が困難であると予測されるブロック順、そして同じピクセル順番テーブルを利用できるブロックが多い順にこのテーブルを適用して予測符号化効率を改善する。

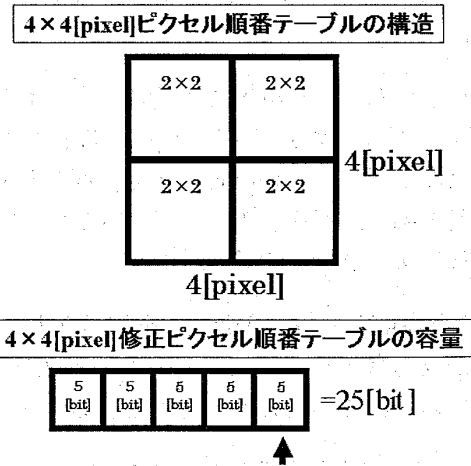


図2 2×2[pixel]ブロックの合成による4×4[pixel]のピクセル順番テーブルの構造と容量

ピクセル順番テーブルはブロックの大きさにより必要な容量が変わる。正方形ブロックの場合、1辺のピクセル数の階乗だけそのブロックの並び方は存在するため、大きなブロックサイズはテーブルの作成に不利である。[1] 2次元画像データに対しては我々の研究より図2のような2×2[pixel]のブロックの合成により作成された4×4[pixel]のブロックが適しているという結果となった。

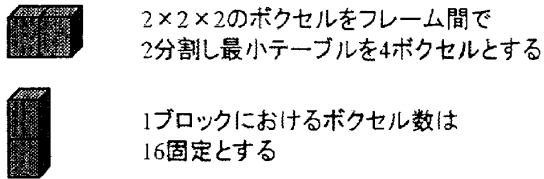


図3 ブロックのとりうる形状

### 3 3次元画像データへの適用

ピクセル順番テーブルを3次元画像データに適用することで2次元画像データに比べてデータ中のブロック数が増加する。このとき、ブロックの大きさを変化させなかった場合、同一ピクセル順番テーブルを持つブロックの増加も期待できる。よって、2次元画像データに比べ3次元画像データの方がより効果的に符号化を行うことが可能になると考えられる。また、3次元方向にブロックを作成することができるため、ブロック内ピクセルの濃度値の相関が高くなり、実際に保存するピクセル順番テーブル(基本テーブル)数を削減することが可能になるとも予想される。

ピクセル順番テーブルを3次元画像データに適用する際、2次元画像データの場合に対して以下のように改良した。

1. ブロックのボクセル化 2次元では $4 \times 4$ [pixel]であったブロックを $2 \times 2 \times 2$ [voxel]を2つ組み合わせた1ブロック16[voxel]とし、図3のように2次元方向、3次元方向をブロック内ボクセルの濃度値により選択できるようにした
2. 3次元参照テーブルの追加 単位ボクセルブロックが立方体ではないため、ブロック間の予測符号化の追跡を示したテーブルを追加する
3. 同一ピクセル順番テーブルを持つブロックの扱い 同一ピクセル順番テーブルを持つブロックが多い順に基本テーブルを作成するが、ブロック数が同じ場合はブロックのエントロピーと分散が大きい順に基本テーブルを作成する

### 4 実験結果と考察

本実験に用いた画像データはX線ヘリカルCT画像であり、10枚の連続した2次元axial画像データを重ねて3次元画像データとした。1枚の2

次元画像データは $512 \times 512$ [pixel]、16[bit/pixel]より容量524288[Byte]、よって3次元画像データは5242880[Byte]である。予測符号化には単純な差分符号化(DPCM)を用い、エントロピー符号化にはAdaptive Huffman符号化を用いた。これらについて、従来のピクセル順番テーブルを用い、1枚ずつ圧縮したとき(従来法)のデータ容量と今回の提案法により圧縮されたデータ容量、UNIXのcompressを用い3次元画像データを圧縮した容量とを比較した。結果を表1に示す。表1より、3次元画像データの場合には従来法に比べ今回我々が提案した手法の方がより圧縮率が高いことがわかった。

表1 符号化法による圧縮ファイル容量

符号化法	DPCM+compress	従来法	提案法
圧縮後容量	2148715 [Byte]	1944570 [Byte]	1263739 [Byte]

### 5 おわりに

本研究では、2次元画像データに用いてきた従来のピクセル順番テーブルを3次元画像データに適用し、符号化効率について従来法との比較を行った。今回我々が提案した手法は従来法よりもより高い圧縮率が得られる事がわかった。提案法は圧縮を行わない画像データと比較して3次元画像データを1/5に圧縮することが可能であった。

今後は、アルゴリズムを改善して、長い計算時間を短縮する必要がある。また、3次元画像データの多くがボリュームレンダリングのためのデータであることを考慮し、ボリュームレンダリングを補助する圧縮アルゴリズムを組み込む必要があると考えられる。

### 参考文献

- [1] N.Ranganathan, Steve G.Romaniuk and Kameswara Rao Namuduri, "A Lossless Image Compression Algorithm Using Variable Block Size Segmentation," IEEE Trans. Image Processing Vol.4, No.10, pp.1396-1406, Oct. 1995.
- [2] 明上山温, 坂本雄児, "ピクセル位置記憶テーブルを用いた画像の可逆符号化法," 電子情報通信学会全国大会, Sep. 1997.