

静止画像の無歪み圧縮に適した領域分割アルゴリズム

佐藤 琢麻[†] 石田 崇[†] 小林 学^{††} 平澤 茂一[†]

[†] 早稲田大学理工学部経営システム工学科 〒169-8555 東京都新宿区大久保 3-4-1

^{††} 湘南工科大学工学部情報工学科 〒251-8511 神奈川県藤沢市辻堂西海岸 1-1-25

E-mail: †{takuma,ishida,hirasawa}@hirasa.mgmt.waseda.ac.jp, ††kobayasi@info.shonan-it.ac.jp

あらまし 画像認識等で用いられる領域分割アルゴリズムは、画像からオブジェクトを取り出すため動画像圧縮等に適用されている。静止画像に対しても画像を領域に分割し、領域ごとに圧縮を行う事によって、分割しないままの画像圧縮に比べて圧縮率の低減が可能となると考えられる。従来の領域分割アルゴリズムはK平均法[1]や分散[4]を用いた手法が提案されているが、必ずしもこれらの分割アルゴリズムが画像圧縮に適しているとは限らない。そこで本稿ではオブジェクト抽出のための領域分割ではなく、圧縮に適していると考えられる指標としてMDL(Minimum Description Length)基準を導入し、圧縮を目的とした領域分割アルゴリズムを提案し、圧縮率の低減を図る。また、シミュレーションによってこの手法の圧縮性能を評価する。

キーワード 領域分割, エントロピー, MDL基準, 無歪み圧縮

A Segmentation Algorithm for Lossless Compression of Images

Takuma SATOH[†], Takashi ISHIDA[†], Manabu KOBAYASHI^{††}, and Shigeichi HIRASAWA[†]

[†] Dep.of Industrial and Management Systems Engineering, Waseda University 3-4-1 Ohkubo, Shinjuku-ku, Tokyo, 169-8555 Japan

^{††} Dep.of Information Science, Shounan Insititute of Technology 1-1-25 Nishikaigan Tsujido, Fujisawa-shi, Kanagawa, 251-8511 Japan

E-mail: †{takuma,ishida,hirasawa}@hirasa.mgmt.waseda.ac.jp, ††kobayasi@info.shonan-it.ac.jp

Abstract Segmentation algorithms used in image recognition methods are mainly adopted to extract objects. We can anticipate that the rate of compression is reduced, comparing with the direct compression of non-segmented image, when each segmented images are compressed for still pictures. Several segmentation algorithms using K-means method or analysis of variance have been proposed, however, they are not always suitable for still image compression. Therefore, in this paper, we propose a segmentation algorithm which aims at further reduction of rate of compression for still images. Then this algorithm apply an MDL criterion which suitable for the compression instead for extraction of objects.

Key words segmentation, entropy, MDL criterion, lossless compression

1. ま え が き

近年の情報処理技術の革命的な進歩により、音声や静止画像、さらには動画像といった大量のデータを扱うことが極めて容易となった。しかし情報伝達の見地に立つと通信容量や記憶容量が大きくなりつつある現在においても、依然としてこれらのデータ量は膨大であるため、もとの情報の損失を最小限におさえつつデータ圧縮をいかに効率よく行うかが非常に重要な課題となっている。

静止画像の無歪み圧縮において、画像認識などで用いられる領域分割の考え方をういた手法が研究されている[2],[3]。領域

分割とは画素の情報を元に画像を領域に分割し、画像内に含まれる“人”や“背景”といったオブジェクトを抽出するための手法である。抽出されたオブジェクトは相関の強い画素の集合体として現れる。そのため、オブジェクト単位で符号化を行うことによって画像の局所的な性質を有効に利用する事ができ、画像全体をそのまま圧縮するよりも効率の良い圧縮ができると考えられる。

鈴木ら[2]は各画素に対して元画像の輝度よりもレベルを下げて量子化し、同じ輝度を持つ隣接した画素同士は同じ領域であるとして、画像全体を分割する手法を提案した。

また、土屋[3]らは2分木を用いて表現される領域分割と

MDL(Minimum Description Length) 基準 [5] を組み合わせた無歪み画像圧縮法を提案している。

本研究では画像認識等で用いられている領域分割手法を静止画像の無歪み圧縮に有効に適用する事を考える。画像の領域分割法として、色の分散などから画像を局所的な類似性によって領域に分割する手法がある [1], [4]。これらの領域分割手法では、領域中の画素の分散を指標としており、オブジェクトの抽出に有効であるが、領域分割手法を画像圧縮に適用するには必ずしもこの基準が適しているとは限らない。そこで本研究では、以上の領域分割手法を圧縮に適用する事を前提として、圧縮を考慮した指標として MDL 基準を導入し、静止画像の無歪み圧縮に適した領域分割アルゴリズムを提案する。

2. 準備

2.1 分散を用いた領域分割法 [4]

本章では色の分布の分散により画像を分割する領域分割法 [4] について説明する。この手法では、画素の輝度を 4bit(16 色) に量子化して領域分割を行う。量子化された画像において、色分布の分散の度合い (local J value) を求め、この度合いがある閾値よりも小さい画素の集合を領域とするアルゴリズムである。これは、画像中のオブジェクトについて、オブジェクト内での局所的な色の散らばり具合は小さく、オブジェクトとオブジェクトの境界付近 (境界領域) では大きくなるという性質を利用している。すなわち、局所的な色の散らばりが小さい画素の集合で領域を構成する事によって、オブジェクトを適格に抽出している。

本手法の手順は以下の通りである。

- ① 画像量子化を行う。
- ② 初期領域の種を生成する。
- ③ 初期領域の種に周辺の画素を組み込み、画像全体を複数の領域に分割する (初期領域)。

④ ③でできた初期領域において、隣接する初期領域同士を色の分布ヒストグラム間の距離によって比較し、閾値より小さければこれを統合する。

2.1.1 量子化

静止画像の領域分割を行うために、1 画素が 24 bit や 8 bit のカラー画像を 4 bit(16 色) の画像へ量子化する。量子化とはいくつかの色を 1 つの色で代表させることである。つまり似た色同士を 1 つの色で代表させ、元の画像を 16 色の画像にするのである。これにより、元の静止画像において類似した色パターンの領域をなるべく損なうことなく、扱いやすいデータへと変換することができる。

2.1.2 窓

任意の画素を中心としてその周りの一定の距離に属する画素の集合を窓と呼び、これによって各画素についてその周辺の色の散らばり具合を計算する。また、窓の大きさを窓に属する画素数で定義する^(注1)。一般にオブジェクトの境界付近の画素を



窓 1(M1 = 61)

窓 2(M2 = 152)

図 1 $U = 2$ の場合の窓 [4]

中心とした窓内の色分布の分散は大きくなり、逆にオブジェクトの内部では色パターンが類似しており、窓内の色分布の分散は小さくなる傾向がある。分散分析を用いた領域分割アルゴリズムでは、その分散の大きさによって領域を分割する点に特徴がある。さらにこの窓を用いる利点は、窓を大きくすると画像の全体的な領域の境界を捉える事ができ、小さくすると色の鋭い変化を捉える事ができる点である。つまり、大きな窓を用いると画像を大まかに分割することができ、小さな窓を用いると画像を細かく分割することができる。窓の大きさはユーザがどの程度まで画像を詳細に分割したいかによって適宜定める事ができる。

本手法では窓の大きさ $M_u (u = 1, 2, \dots, U)$ は以下の通りに定義する。ここで U は窓の数である。

$$M_1 < M_2 < \dots < M_U \quad (1)$$

窓の例を図 1 に示す。

2.1.3 local J value

窓内における色パターンの偏り具合から、その窓の中心画素がオブジェクトの境界領域である度合い (以下 local J value と呼ぶ) を計算する。この local J value の値 J は窓の中心画素に対して与えられ、以下の式で求められる。

$$S_T = \sum_{z \in Z} \|z - m\|^2 \quad (2)$$

$$S_W = \sum_{i=1}^C \sum_{z \in Z_i} \|z - m_i\|^2 \quad (3)$$

$$J = (S_T - S_W) / S_W \quad (4)$$

ここで、

Z : 窓に属する画素の集合

z : 画素の座標^(注2)

m : 窓内の Z の重心^(注3)

C : 量子化された後の色の数

Z_i : 窓内における色別の各画素の集合 ($i = 1, 2, \dots, C$)

m_i : 窓内の Z_i の重心 ($i = 1, 2, \dots, C$)

である。

式 (4) で求められる J の値が小さい場合は、窓内の画素のばらつき方が小さいことを表している。つまり、このときには窓

(注1): ただし窓の大きさが同一のときには、窓の形状は中心画素の位置によらず同一とする。

(注2): 画像を x, y 軸表現で表せるとすると、 $z = (x, y)$ となる。原点は任意で良い。

(注3): 一般に窓の中心であるが、画像の端では中心画素の位置とは異なる。



初期領域の種になる画素 初期領域の種にならない画素

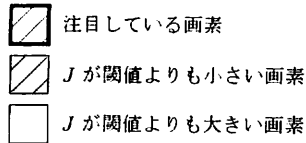


図2 初期領域の種となる画素とならない画素の例

の中心画素が領域境界にある可能性が低いことを意味している。逆に J の値が大きい場合には、窓の中心画素が領域境界にある可能性が高いことを意味する。

2.1.4 初期領域の種の生成

まず、領域分割の初期段階として確実にオブジェクトの内部に属していそうな画素の集合を作る。この画素の集合のことを初期領域の種と呼ぶことにする。

J の値が小さいと、その画素は境界領域ではなくオブジェクトの内部に位置している可能性が高い画素であると考えられる。また、ある閾値を設定し、閾値よりも小さい J を持つ画素について、もしその画素の周辺の画素も閾値より小さい J 値を持っているならば、その画素がオブジェクトの内部に位置する可能性がさらに高いといえるであろう。逆に注目した画素の周辺の画素の J が閾値よりも大きいときには、その画素が領域境界である可能性が高くなると考える。

以上より、次の手順で初期領域の種を作る。

- ① 画像内の全ての画素に対し式(4)より J の値を計算する。
- ② J の平均値 μ_J と標準偏差 σ_J を求める。
- ③ 適当な値 α を用いて $\mu_J + \alpha\sigma_J$ を閾値とし、この値よりも小さい J を持つ画素を領域の種の候補(領域候補画素)とする。
- ④ 1つの領域候補画素に着目し、もし隣接する上下左右の画素も候補画素となっているならば、着目した候補画素を初期領域の種とする。

初期領域の種になる候補の画素の例、初期領域の種の例を図2、図3に示す。

2.1.5 初期領域への拡大

2.1.4 で求められた初期領域の種は、確実にオブジェクトの内部であると考えられる画素の集合として生成された領域である。選択されずに残った画素は以下の手順に従って、初期領域の種に付け加えてゆく。この手順により画像は初期領域によって分割されることになる。

- ① 初期領域の種に含まれなかった画素全体から J の平均値 $\mu_{J'}$ を新たに求める。
- ② 初期領域の種に接している画素のうちで J の値の小さい順にその接している初期領域の種に加え、種を拡大する。ただ

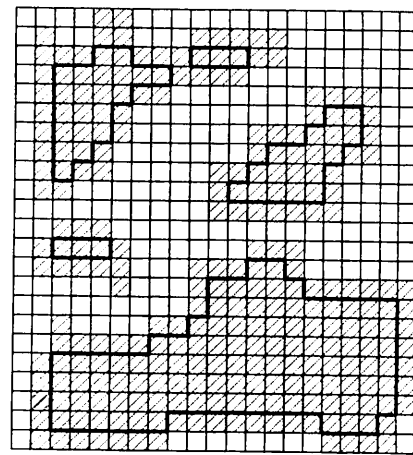


図3 初期領域の種の例

し $\mu_{J'}$ よりも大きな J をもつ画素は含めない。また、2つ以上の領域に接している画素についても含めない。

③ さらに各領域に含まれない画素はより小さい窓に対して J の値を求め直し、①、②を繰り返す。

④ 大きさ最小 (M_1) の窓まで J の値を計算し、かつどの初期領域の種にも含まれずに残った画素は J の値の小さな画素から隣接した初期領域の種に含める。

2.1.6 領域の統合

一般的に領域分割アルゴリズムには必要以上に細かく画像を分割してしまうという問題がある。そこで、これまでの手順で分割された領域(初期領域)の統合を行う。もし、隣接した領域を比較して色の分布が似ている場合には必要以上に領域が分割されてしまった可能性が高いといえる。そこで、以下の手順により初期領域を統合する。

① 2.1.5 節で求めた各領域に対して、色の分布を比較するために量子化された色のヒストグラムを作成する。

② 相対頻度の総和が1となるようにヒストグラムを正規化し、隣接する領域間のヒストグラムの距離 D_{CH} を以下の式から求める。

$$D_{CH} = \|P_A - P_B\|^2 = \sum_{i=1}^C \left(\frac{n_A(i)}{N_A} - \frac{n_B(i)}{N_B} \right)^2 \quad (5)$$

ここで、 P_A, P_B は隣接する領域 A, B の正規化されたヒストグラムを表している。また $n_A(i), n_B(i)$ は、 A, B それぞれにおける色 i ($i = 1, 2, \dots, C$) の頻度、 N_A, N_B はそれぞれの領域の画素数である。

③ 適当な閾値 $\beta (> 0)$ を用意し

$$D_{CH} < \beta$$

を満たすとき、隣接した領域 A, B 間で色の分布が類似しているとみなし、領域を統合する。

2.1.7 実画像への射影

以上の手法により領域分割した画像は量子化された画像であるが、この分割を実画像に射影し、実画像に対する領域分割とする。

2.2 MDL(Minimum Description Length) 基準 [5]

MDL 基準とは、与えられたデータ系列に対して、データを出力する情報源モデル自身の記述も含めて最も記述長が短くなるような確率モデルが、最良の確率モデルであるという基準である。MDL 基準は以下の式で表される。

$$MDL(m) = -\log_2 P_m(x^n) + \frac{k_m}{2} \log_2 n \quad (6)$$

ただし、

x^n : 長さ n のデータ系列 $x^n = x_1 x_2 \dots x_n$

m : 候補となる確率モデル

k_m : モデル m のパラメータ数

$P_m(x^n)$: モデル m における系列 x^n の尤度

である。

ここで MDL で表される式の第 1 項はあるモデル m の元での理想符号長、第 2 項はモデル化に必要な確率パラメータを表現する符号長を表している。MDL 基準では上記の式を最小とする確率モデルが、そのデータ系列に対して最良の確率モデルであるとする基準である。MDL 基準は複雑なモデルに対しては第 1 項の値が小さくなり、第 2 項の値は記述する量が増えるため大きくなる傾向がある。逆に単純なモデルに対しては第 1 項が大きくなり、第 2 項が小さくなる。つまり、第 1 項と第 2 項はトレードオフの関係にあることがわかる。

3. 提案方式

3.1 着眼点

2.1.6 で述べたように、従来手法において初期領域の統合の際には、色の分布を比較している。すなわちこれは、色のばらつき方が似ている領域は同じ領域であるとみなすことによって領域の統合を行っている事になる。しかし、この手法は圧縮を前提とした場合には必ずしも最良な統合がなされているとは考えられない。

そこで本研究では Rissanen によって提唱された MDL 基準を導入し、初期領域の統合を行う手法を提案する。MDL 基準を用いることにより、符号長を最小にするような領域の統合法を行うことになるので、圧縮率を低減させることが可能になると考えられる。

前章の準備をもとに提案手法の流れを以下に示す。

- ① J 値による初期領域の生成。
- ② MDL 基準による初期領域の統合。
- ③ 領域分割された画像の領域情報と領域のデータの符号化。

3.2 初期領域の生成

本研究では、初期領域の種の生成、初期領域の種の拡大までは従来手法 (2.1 節) と同様の手法を用いた。

3.3 MDL 基準による初期領域の統合

従来手法によって得られた初期領域に対して、MDL 基準を用いて領域の統合を行う。なお、初期領域を生成するまでは

4bit に量子化されたデータに基づいて行ったが以下の手順では元の画像を用いる。このとき、元の画像の色の数を C_0 とする。

分割される領域はあまり小さすぎると圧縮の効率を悪くすると考えられるので、画素数の少ない初期領域から順に考慮してゆく。

いま、注目した領域 A に隣接しているすべての領域 B_1, B_2, \dots, B_t において

m_0 : 領域をまったく統合しないモデル

m_1 : 領域 A, B_1 を統合するモデル

m_2 : 領域 A, B_2 を統合するモデル

⋮

m_t : 領域 A, B_t を統合するモデル

を考える。4 に領域統合モデルの例を示す。

すべてのモデル m_0, m_1, \dots, m_t に対して $MDL(m_i)$ を計算し、最小となるモデルを選択する。すなわち選択されたモデルにしたがって領域の統合を行う。

具体的な $MDL(m)$ の計算法を述べる。いま、領域 A の画素数を N_A 、 N_A における色 i ($i = 1, 2, \dots, C_0$) の頻度を $n_A(i)$ と書く。また、領域 B_j ($J = 1, 2, \dots, t$) についても同様の記法を用いる。確率モデルとして無記憶の多項分布を仮定したとき、情報源領域 A における画素データの対数尤度 L_A は

$$L_A = \sum_{i=1}^{C_0} n_A(i) \log_2 \frac{n_A(i)}{N_A} \quad (7)$$

で計算される (領域 B_j についても同様)。また考慮している領域の総画素数を $N (= N_A + \sum_j N_{B_j})$ とおくと、 $MDL(m)$ は以下のように求めることができる。

- (1) 領域を統合しないモデル

$$MDL(m_0) = -L_A - \sum_{s=1}^t L_{B_s} + \frac{t \cdot (C_0 - 1)}{2} \log_2 N \quad (8)$$

- (2) 領域 A と B_j ($j = 1, 2, \dots, t$) を統合するモデル

$$MDL(m_j) = -L_{A, B_j} - \sum_{s=1}^t L_{B_s} + L_{B_j} + \frac{(t-1) \cdot (C_0 - 1)}{2} \log_2(N)$$

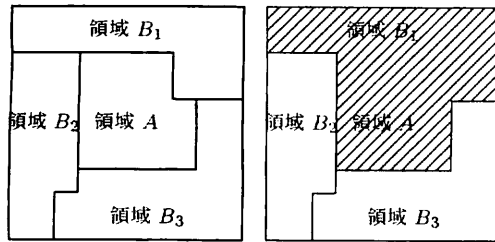
ただし、 L_{A, B_j} は統合された領域の対数尤度を表し、

$$L_{A, B_j} = \sum_{i=1}^{C_0} (n_A(i) + n_{B_j}(i)) \log_2 \frac{n_A(i) + n_{B_j}(i)}{N_A + N_{B_j}} \quad (10)$$

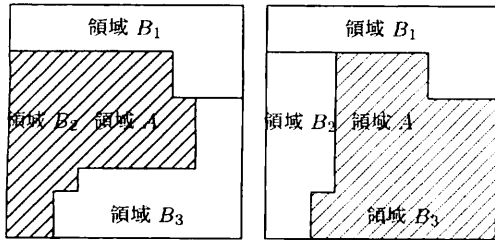
である。

以上で求めた $MDL(m)$ において最小の値をとるモデルを選択する。すなわち、 $MDL(m_0)$ が最小であれば領域 A についてはどの領域とも統合しない。 $MDL(m_j)$ が最小であれば領域 A と領域 B_j ($j = 1, 2, \dots, t$) を統合する。

この手順ののち、改めて画素数が最小となる領域に着目し同様のことを繰り返す。すべての領域について MDL の計算を行いモデルの選択を行ったら終了する。



m_0 : 領域を統合しない m_1 : 領域 A, B₁ を統合



m_2 : 領域 A, B₂ を統合 m_3 : 領域 A, B₃ を統合

図4 領域統合におけるモデルの具体例

3.3.1 符号化

次に、これまで分割された領域を

- (1) 分割された領域の境界の情報
- (2) 領域内の画素データ

に分けてそれぞれ符号化する。

データの符号化にあたって、領域境界の情報を表現するために、領域分割されたデータに対して“領域内の画素”と“領域境界”の画素に分類する。すなわち、上下左右で隣接している画素と属している領域が異なっている場合には、この画素は領域境界にある画素であるとする。それ以外の画素は領域内の画素となる。

以下では領域境界を示すためのデータ系列と分割した領域ごとに画像データ系列を作り、符号化を行う流れを示す。

① 領域境界を示すために、領域内の画素に“1”，領域境界の画素に“0”を割り当てる。このデータを領域境界データと呼ぶ。

② 領域境界データを左下から右上へと横に順次読み込む。新たに現れる領域に対して逐次新しい領域番号を割り振る。ただし領域の境界であるデータは、読み込まれた順序の早い領域に含めるものとする。

③ 領域番号の順に、領域に属する画素に対して画像データを左下から右上へと横に読み込み、画像データのファイルを作成する。

④ この領域の境界ファイルと領域ごとの画像データを従来の無歪みデータ圧縮法を用いて圧縮する。

提案方式で増加するデータ量は領域境界データのみである。

4. シミュレーション結果と評価

4.1 シミュレーション条件

提案方式の有効性を示すためにシミュレーションを行った。以下にシミュレーションの条件を挙げる。

- ① シミュレーションの対象は画像テストデータである water-

表1 圧縮率 (%)

ファイル名	サイズ (bytes)	分割なし	png	従来手法	提案手法
lena	263222	0.634	0.632	0.585	0.579
peppers	263222	0.473	0.469	0.463	0.455
monarch	394294	0.461	0.457	0.464	0.442
tulips	394294	0.554	0.550	0.539	0.530
sail	394294	0.665	0.664	0.667	0.652

表2 領域境界データの割合 (%)

ファイル名	提案手法のサイズ (bytes)	領域境界 1	領域境界 2
lena	153997	0.111	0.033
peppers	122520	0.111	0.039
monarch	178683	0.111	0.042
tulips	214215	0.111	0.028
sail	258279	0.111	0.043

表3 領域数

ファイル名	従来手法	提案手法
lena	26	26
peppers	51	33
monarch	40	45
tulips	93	77
sail	25	45

loo color の lena, peppers(ともに 512 × 512 画素), monarch, tulips, sail(ともに 768 × 512 画素)を用いる。また、これらの画像は 1 画素 8bits, 256 色の画像である。

② 初期領域の種の生成で用いられる閾値は $\alpha = -0.6$ とする。

③ 従来手法の領域統合における閾値は $\beta = 0.2$ とする。

④ 分散を用いた領域分割法で用いられる窓は図1の M_1 と M_2 とする。これは小さい窓を用いることにより画像を細かく分割し、本提案方式により圧縮に適した領域を生成する有効性を示すためである。

⑤ 符号化には代表的な無歪み圧縮法である LZH を用いた。

4.2 シミュレーション結果と考察

提案手法による符号化のシミュレーション結果を表1に示す。表の分割なしは、画像を領域分割せずに従来、提案方式と同様に LZH 形式で圧縮を行った結果で、png とは一般的に用いられる無歪みの静止画像圧縮形式である。表の値は圧縮率を表している。提案手法の結果はどの形式に対しても、全ての画像でわずかながら良くなっている。

また、提案手法では原画像のデータ量よりも領域境界を示すファイルの部分 (512 × 512 の画像は 32768byte, 768 × 512 の画像は 49152byte)が増えているのにもかかわらず、分割を行わない圧縮よりも結果は良い。これは、本研究で分割された領域が相関の強い画素の集合体として抽出され、圧縮を考えた統合を行っているため、全体の圧縮率が低減されたためと考えられる。また、??で示す領域境界1は増加した領域境界の圧縮前の画像に対する割合で、領域境界2は増加した領域境界の提案手法に対する割合である。割合としては高々3～4%となつて

いる。これはこの領域境界ファイルの要素がほとんど1であり、境界のみ0を与えているので、非常に小さな大きさに圧縮可能なためである。したがって画像を領域に分割し圧縮を行う有効性を示すことができた。

次に表3より lena に着目すると従来と提案で領域数は同じであるが、圧縮率は従来手法よりも提案手法のほうが良くなっている。このことより、従来手法と提案手法とで異なった統合がされたことが分かる。また、monarch, sail に着目すると提案手法よりも従来手法の方が領域の数は少ないが、提案手法の方が圧縮率は良い。これはもともと従来手法がオブジェクト抽出のための手法であり、圧縮という観点から見ると必要のない統合がなされたためだと考えられる。

以上の考察により圧縮の観点から見るとヒストグラムを用いた統合よりも、MDL 基準を用いた統合のほうが良い統合がなされていると考えられる。

5. まとめと今後の課題

本論文では画像を色の分布により領域分割し、MDL 基準を用いて圧縮率の低減を図るように領域を統合するアルゴリズムを提案し、シミュレーションにより有効性を示した。

本研究の手法において、領域分割アルゴリズムによりできる領域は、初期領域の種に依存しやすいと考えられる。そこで領域の統合のみならず、初期領域の種を生成する時点で圧縮の観点に立った種を生成することができれば、更なる圧縮率の低減が可能と考えられる。したがって圧縮の観点に立った種を生成することが今後の課題である。また、今回は圧縮を行う時点でLZH 形式の圧縮を行ったが、領域分割に適した符号化の検討が必要だと考えられる。また、今回は土屋らの手法との比較を行っていないが、今後土屋らの手法との比較が課題である。

6. 謝 辞

本研究を行うにあたり、数多くの御助言、御支援を賜りました武蔵工業大学 後藤正幸先生、早稲田大学 中澤真先生、八木秀樹氏並びに平澤研究室各氏に心より感謝申し上げます。

文 献

- [1] 高木, 下田, “画像解析ハンドブック,” 東京大学出版会
- [2] 鈴木馨, 加藤茂夫, 宮道壽一, “領域分割に基づく自然画像の可逆符号化についての検討,” 画像電子学会誌 vol.28,no.1, 1999.
- [3] 土屋英亮, 伊藤秀一, 渡辺修史, “MDL 原理と2分木構造セグメンテーションを用いた画像の無ひずみ符号化アルゴリズム,” 電子情報通信学会論文誌 D-II vol.J80-D-II no.2 pp.415-425, Feb 1997.
- [4] Y.Deng,B.S.Manjunath,H.Shin, “Color image segmentation,” Proc. of IEEE Conf. on Computer Vision and Pattern Recognition,(CVPR),1999.
- [5] J.Rissanen, “Universal coding,information,prediction,and estimation,” IEEE Trans.Inf.Theory, vol.IT-30,no.4,pp629-636,July 1984.