

2P-10

マルチツリーコーディングを用いた  
画像のベクトル量子化

加藤 誠巳 長屋茂喜 吉岡稔陽  
(上智大学理工学部)

1. まえがき

画像の圧縮の一手法としてベクトル量子化 (VQ) が挙げられるが、この VQ の主な論点となるのはコードブック作成と、コード探索である。コードブック作成の代表的なアルゴリズムである LBG アルゴリズム<sup>[1][2]</sup>は、コードブック作成の処理時間にかかなりの時間を要すること、初期値をどの様に設定するかという問題、また収束するまで繰り返し計算するので収束条件により結果が異なるなどの問題がある。本稿では、コードブック作成とコード探索の処理時間の短縮を目的としたコードブック作成アルゴリズムについて述べる。

2. 基本的手法

LBG アルゴリズムでは n 次元のベクトル空間内でそれぞれのクラスタの重心を求め、さらにクラスタリングを行う。本手法では n 次元のベクトル空間である超立方体を二等分割して、大きさを半分とする超立方体に分割する。つまり一回の分割で n 次元の超立方体が 2^n 個の超立方体に分割され、クラスタリングが行われていくことになる。この様に超空間を超立方体 (Hypercube) に分割してベクトル量子化 (VQ) を行っていくものである。(以下 HQ 法と呼ぶ。)

2.1 コードブック作成手順

ここでは色の 3 原色 B,R,G(0~255) をそれぞれの軸とした n=3 次元の場合を例にとって、提案する手法を説明する。

- Step 1 Fig.1 に示すように全空間を 2^n = 8 個の立方体に分割する。
- Step 2 それぞれの立方体内の全ベクトルの重心を求める。
- Step 3 その重心とその立方体内の全ベクトルとの間の歪を求める。  
歪は次式で定義するものとする。(k:ベクトルの個数)

$$\text{歪} = \sum_{i=1}^k \{ (b_g - b_i)^2 + (r_g - r_i)^2 + (g_g - g_i)^2 \}$$

- Step 4 現在分割されている全立方体中で歪が最大となっている立方体を分割する。もし全立方体が設定された候補数以上になれば、それぞれの立方体の重心を代表ベクトルとしコードブックを作成し終了する。そうでなければ Step 2 に戻る。  
(分割したときにベクトルが存在しない立方体は候補とは認めないことにする。) このとき、候補となったそれぞれの立方体のツリー (位置情報) とコード番号を含んだマルチツリーを作成する。

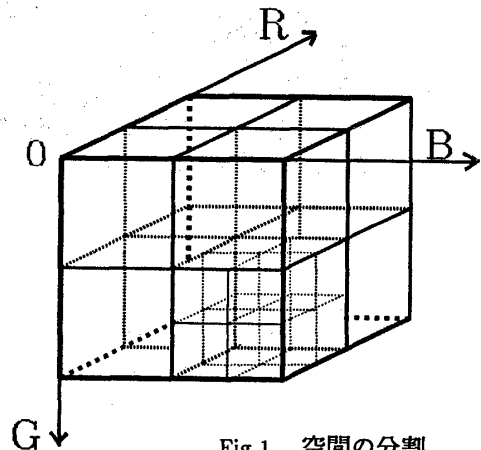


Fig 1. 空間の分割

2.2 コード探索

コードブック作成時に作られたマルチツリーを用いて、候補ベクトルをツリーコードに変換してから木探索を行う。これは階層化構造の性質を既に持っているため、高速に処理することが可能である。だが立方体の代表値が重心に割り当てられているので、隣接または周囲の立方体の探索も必要となる。

3. 実際の画像への適用例

Fig.2 に示す SIDBA の girl(256 x 256 画素,256 階調) を原画像とした。ここでは 4 x 4 の画素群を切り出してそれを 1 つの 16 次元ベクトルとして、コードブックを作成して量子化を行った。

処理の流れは次の通りである。

原画像→直流分離→前処理→HQ→後処理→直流加算→再生画像

この場合直流分離とは、この場合 4 x 4 画素中の最小値を見つけそれをそれぞれの画素から差し引く処理を言う。前処理として 4 x 4 画素内の左上の方が、輝度が高くなるように変換した。この様に処理して再生した画像を Fig.3 に示す。また比較のために LBG 法で量子化して再生した画像を Fig.4 に示す。



Fig 2. 原画



Fig 3. HQ (候補数 586)



Fig 4. LBG (候補数 512)

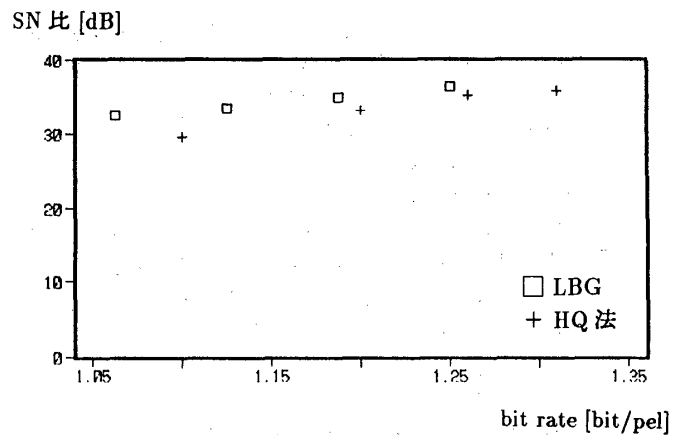


Fig 5. 符号化特性

ここで提案した HQ 法の特徴は、評価基準量となるものに対して非常に敏感なことである。この場合、歪を評価基準量としているので歪が大きいところで分割され、量子化歪を少なくしている。また歪はそれぞれの超立方体内のベクトルの個数と関係しているため、出現確率を考慮した分割となっている。しかしその反面一様な分布に対しては弱くなってしまい、さらにエッジとなるベクトルは他のベクトルに比べ種類が多く、しかも少数なので候補数ある程度以上確保しないとエッジはうまく再現できないこととなる。Fig.5にビットレートとSN比の関係をグラフで表した。これを見ると候補数を減少させると極端にSN比が悪化する。逆に言えば候補数ある程度確保するとよい結果が得られることが判明した。尚ここでSN比は次式を用いた。

$$SN = 20 \log_{10}(255/R.M.S.) \quad (R.M.S.: \text{平均自乗誤差の平方根})$$

#### 4. むすび

ここに提案した HQ の手法は、LBG アルゴリズムと同様に多段化、可変ブロック長化等と組み合わせれば低ビットレートでSN比が良くても高速にコードブックを作成することが期待される。また前述した候補数の問題も、ハフマン符号化等を用いることにより解決すると考えられる。最後に御討論いただいた本学マルチメディア・ラボの諸氏に謝意を表す。

#### 参考文献

- (1) Y.Linde, A.Buzo, R.M.Gray : "An algorithm for Vector Quantizer Design", IEEE Trans, Vol.COM-28, No.1, pp.84-95 (1980).
- (2) R.M.Gray : "Vector Quantization", IEEE ASSP MAGAZINE, pp.5-29, April (1984).
- (3) SHU-Xiang Li, H.Loew. : "The Quadcode and Its Arithmetic", Comm. of the ACM., Vol.30, No.7, pp.621-626 (1987).
- (4) 有本 : "Rate-Distortion 理論", 信学誌, Vol.61, No.5-No.9 (1978).
- (5) 白井, 今井 : "適応的ベクトル量子化を用いた画像符号化の一方式", 信学技報, PRU86-114 (1986).
- (6) 野々村, 佐藤, 山下 : "ベクトル量子化とブロック符号化を用いた画像の高効率符号化", 信学誌, Vol.J71-D No.10, pp.2010-2017 (1988).