時空間ヒルベルト走査を用いた適応的線形予測符号化による 動画像可逆圧縮法

A Lossless Video Compression Method Based on Adaptive Linear Predictive Coding Using Spatio-temporal Hilbert Scanning

学生会員 坂東 幸浩[†],

横山 貴政[†], 正会員 鎌田 清一郎^{††}

Yukihiro Bando[†], Takamasa Yokoyama[†] and Sei-ichiro Kamata^{††}

Abstract In video processing, compression techniques are required to improve the efficiency of transmission and storage of video data. Data compression methods can be classified into two basically different categories: lossy compression methods and lossless compression methods. Lossless compression methods are required for medical image processing and for compressions broadcast materials. We developed a novel video lossless compression method using the Hilbert curve, by mapping the three-dimensional data to one-dimensional data along the curve and then applying adaptive linear prediction coding to the one-dimensional data. Experimental results show that better compression ratio is obtained with our method than with losslessJPEG, although the processing time for both methods is almost equal.

キーワード: ヒルベルト走査, 時空間領域, 動画像可逆圧縮, 適応的線形予測

1. ま え が き

動画像圧縮は動画像の効率的な伝送・蓄積のためには不可欠な技術であり、圧縮による情報欠損の観点から非可逆圧縮と可逆圧縮とに大別される。このうち可逆圧縮は、医療画像・放送編集用のVTRのように高画質化の要求が強い応用において求められている。

動画像の可逆圧縮法として、太田ら¹⁴⁾はDCTによる可逆符号化方式を検討している。また、小川ら¹³⁾は、各シンボルの生起確率分布をパラメトリックに表現し、フレーム毎にマルコフモデルを切替えるエントロピ符号化を提案している。これら両手法はフレーム間の相関を利用した手法であり、色データ間の相関は利用されていない。色データ間の相関を利用したものとして、Memonら¹⁰⁾の手法がある。しかし、この手法では、異なる色データからの予測と動き補償による予測のうち、予測残差の小さな方を選択する処理が必要である。このため、計算量が大きなもの

となり、処理時間に関して問題を残す。

動画像は静止画像(フレーム)の時系列として表現される。 $x-y-t$ 空間において $x-y$ 平面に平行なフレームが t 軸方向に配置され、 $x-y-t$ 空間内の領域を形成する。本論文では、この $x-y-t$ 空間の部分領域を時空間領域と呼ぶ。動画像にはフレーム内の空間的な相関と時間軸方向の時間的な相関が存在し、これを利用して情報の圧縮が行われる。時空間領域における圧縮手法では、動画像データを時空間領域に存在する3次元データと見なし、複数のフレームを一括して処理する。例として、P.Willeminらの時空間領域の再帰的な2分割処理に基づく領域分割符号化¹⁷⁾、M.S.Lazarらによるレンジブロックとドメインブロックとの間のマッチングを時空間領域内の分割領域に対して行ったフラクタル符号化⁸⁾、変換符号化に関する試みとしてY.Chanらの3次元DCT³⁾、C.I.Podilchukらの3次元サブバンド符号化¹⁵⁾などがある。ただし、これらはいずれも非可逆圧縮のための手法である。

本論文では、時空間領域における可逆圧縮手法を提案する。提案手法は、時空間領域内の3次元データを1次元データへと展開し、得られた1次元データに対して線形予測を行う符号化処理である。このように、多次元空間内のデータを1次元データへと展開し、1対1の対応をとることを走査と呼ぶ。提案手法では、3次元データの走査にヒルベルト曲線⁶⁾を利用する。ヒルベルト曲線は2次元以上の連続な領域のすべての点を一度だけ通る空間充てん曲

2000年8月25日、映像情報メディア学会年次大会で発表

2000年7月3日受付, 2000年12月15日再受付, 2001年1月10日採録

[†]九州大学 大学院 システム情報科学府

(〒812-8581 福岡市東区箱崎 6-10-1, TEL 092-642-3886)

^{††}九州大学 大学院 システム情報科学研究院

(〒812-8581 福岡市東区箱崎 6-10-1, TEL 092-642-4070)

[†]Graduate School of Information Science and Electrical Engineering, Kyushu University

(6-10-1, Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan)

^{††}Faculty of Information Science and Electrical Engineering, Kyushu University

(6-10-1, Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan)

線¹⁰⁾の一種であり、近傍保存性の良さから画像処理を中心に広く利用されている。

先に筆者らは時空間領域における非可逆圧縮手法を提案した²⁾。この非可逆圧縮手法も、今回提案する可逆圧縮手法と同様、時空間領域の3次元データを走査により展開した1次元データに対する処理である。また、DCT演算・動き補償といった複雑な処理を必要としないため、高速処理に適している。したがって、動画データを手手法に共通な1次元データへと展開しておくことにより、高速処理が可能な可逆圧縮・非可逆圧縮を用途に応じて使い分けることが可能となる。

また、本論文では、色データの相関を利用した圧縮法を検討する。画像圧縮における色データの相関の利用は以下の2通りに大別される。ひとつは、色変換を用いる方法⁴⁾⁵⁾である。これは、色データ間のエネルギー比率を偏らせ、圧縮効率の向上を目的としたものである。ただし、可逆圧縮に用いる場合、この色変換は可逆な変換である必要がある。もうひとつが、すでに圧縮の済んだ色データの情報を別の色データの圧縮に用いる方法である。例えば、画像の構造的特徴を全域木により表現し、これを圧縮に利用する場合、R-データに対してのみ全域木を作成し、残るG-データ、B-データの圧縮では、R-データに対する全域木を用いる方法が検討されている⁹⁾。

後者の手法は、後に符号化する色データに対して、処理量を削減できるという特徴がある。一方、これに伴い圧縮効率の劣化は生じるが、色データ間に存在する相関のため、劣化の程度を比較的小さな量に抑えることができる。

本論文で検討する方法は、符号化済の色データの情報を利用するという意味で、後者の部類に属す。しかし、従来法が圧縮効率を多少犠牲にして処理の高速化を図ったのに対し、本手法では、処理の高速化とともに圧縮効率の向上も可能となる。

本論文の構成は、以下のようになる。2章では、時空間領域の3次元データに対する走査法を示し、3章では、走査により得られた1次元データに対する圧縮法を示し、4章では、losslessJPEGとの比較実験の結果を示し、5章で本論文をまとめる。

2. 時空間領域の走査

本手法の対象は x - y - t 空間において x - y 平面に平行なフレーム($\ell_x \times \ell_y$)が t 軸方向に ℓ_t 枚配置された動画データである。この場合、各画素は辺長を ℓ_x, ℓ_y, ℓ_t とする直方体形状の時空間領域 $\{x, y, t \mid 0 \leq x < \ell_x, 0 \leq y < \ell_y, 0 \leq t < \ell_t\}$ の格子点に存在する。ここで、 $\ell_t = n \cdot \Delta_t$ とすると、この時空間領域には t 軸方向の辺長を Δ_t とする n 個の直方体領域($\ell_x \times \ell_y \times \Delta_t$)が含まれる。

一般に、動画データのフレーム($\ell_x \times \ell_y$)は長方形である。このため、長方形領域に対応した2次元擬似ヒルベルト走査¹⁾を各直方体領域($\ell_x \times \ell_y \times \Delta_t$)に対する走査へと拡張する。

図1 長方形領域(12×8)の走査例
An example of scan for a rectangle region (12×8).

```

1   $i \leftarrow 1$ ;
2  for  $j \leftarrow 1$  to  $\ell_x \times \ell_y$ 
3    for  $t \leftarrow 1$  to  $\Delta_t$ 
4       $X[i] \leftarrow HIL_x[j]$ ;
5       $Y[i] \leftarrow HIL_y[j]$ ;
6      if ( $j$  が奇数の場合)
7         $T[i] \leftarrow t$ ;
8      if ( $j$  が偶数の場合)
9         $T[i] \leftarrow \Delta_t + 1 - t$ ;
10      $i \leftarrow i + 1$ ;

```

図2 時空間領域における走査アルゴリズム
An scanning algorithm for spatio-temporal region.

図3 時空間領域の走査
Spatio-temporal Hilbert scan.

張する。例として、図1に長方形領域 $\ell_x = 8, \ell_y = 12$ に対する2次元擬似ヒルベルト走査を示す。

長方形領域($\ell_x \times \ell_y$)に対する2次元擬似ヒルベルト走査において、 j 番目に走査する画素の x 座標および y 座標を各々 $HIL_x[j], HIL_y[j]$ とする。このとき、直方体領域($\ell_x \times \ell_y \times \Delta_t$)に対して、図2に示す計算により求まる走査を用いる。ここで、 $X[i], Y[i], T[i]$ は i 番目に走査する画素の x 座標、 y 座標および t 座標である。この走査は、図3に示すように、2次元擬似ヒルベルト走査を時間軸方向に引き延ばしたものとなり、これを時空間ヒルベルト走査と呼ぶ。

3. 適応的線形予測

本章では、1次元データに対する適応的な線形予測を示す。圧縮対象はRGBカラー動画データとし、R-, G-, B-データ各々に対して時空間領域における以下の処理を行う。まず、2章で述べた時空間ヒルベルト走査を用いてR-, G-, B-データを各々、1次元データ $r[i], g[i], b[i]$

図 4 圧縮処理のブロック図
Block diagram of compression.

($i = 0, 1, \dots, \ell_x \times \ell_y \times \Delta_t - 1$) へと展開する. 次に, 得られた 1 次元データに対して, 適応的な線形予測を行う. 最後に, 予測残差をハフマン符号化する. この圧縮処理の流れを図 4 に示す. 符号化は $g[i]$, $r[i]$, $b[i]$ の順で行うものとし, まず, $g[i]$ の場合を説明する.

3.1 G-データの符号化

走査により得られた 1 次元データ $g[i]$ に対し, 次のような線形予測を行う.

$$\hat{g}[i] = a_1 g[i-1] + a_2 g[i-f(i)] + a_3 g[i-f(i)-1]. \quad (1)$$

ここで, $g[i-f(i)]$ は $g[i]$ と同じフレーム内の画素のうち最も近いものであり, $f(i)$ は次式で表される.

$$f(i) = 2 \times (i \bmod \Delta_t) + 1.$$

なお, $i \bmod \Delta_t$ は整数 i を整数 Δ_t で割った余りを表す. 図 5 に $\Delta_t = 4$ とした場合の符号化画素 ($g[i]$) と線形予測に用いる参照画素 ($g[i-1]$, $g[i-f(i)]$, $g[i-f(i)-1]$) との位置関係を示す. ここで, $g[i]$ に対応する位置の画素は濃い網かけで表されている. また, 参照画素に対応する位置の画素は薄い網かけで表されており, このうち, $g[i-f(i)]$ に対応する位置の画素はその位置を明示してある.

式 (1) の予測値 $\hat{g}[i]$ に対する予測残差

$$g[i] - \hat{g}[i],$$

はハフマン符号化される. このため, 予測残差のエントロピーの減少が圧縮効率の向上につながる.

一般に画像データは非定常性が強い. そのため, 画像データの局所的な性質に基づいて予測器を変化させた場合, そのような予測器は画像圧縮では有効であることが知られている¹¹⁾¹²⁾.

本論文では, $g[i]$ を分類し予測係数を適応的に選択して符号化する. この予測係数選択のための分類をクラス分離と呼ぶ. また, 同じ予測係数を割り当てられた画素の集合をクラスと呼ぶ. $g[i]$ のクラス分離には, 輝度差

$$e_g[i-1] = |g[i-1] - g[i-2]|,$$

および, 時空間領域でのフレームの順序を示す

$$p(i) = i \bmod \Delta_t, \quad (2)$$

を用いる. このため, クラス分離のためのオーバーヘッド情報は必要ない.

図 6 に $\Delta_t = 4$ とした場合の符号化対象画素 ($g[i]$) とクラス分離に用いる画素 ($g[i-1]$, $g[i-2]$) との位置関係を示す. ここで, $g[i]$ に対応する位置の画素は網かけで表されており, また, $g[i-1]$ および $g[i-2]$ に対応する位置の画素には斜線が施されている. 同図 (a- $p(i)$) ($p(i) = 0, 1, 2, 3$) は符号化対象画素が各々,

$$p(i) = i \bmod 4,$$

となるフレーム内に含まれる場合に対応する.

図 6 からわかるように, 各 (a- $p(i)$) ($p(i) = 0, 1, 2, 3$) において, $e_g[i-1]$ の値の意味するところは異なる. 同図 (a-2), (a-3) における $e_g[i-1]$ はフレーム間における時間軸方向の差分値を表している. 一方, 同図 (a-1) における $e_g[i-1]$ はフレーム内における空間方向の差分値を表し, 同図 (a-0) における $e_g[i-1]$ は符号化対象画素とは異なる位置の時間軸方向の差分値を表している. したがって, 同じ輝度差 $e_g[i-1]$ であっても, フレーム位置を表す $p(i)$ が異なれば, 別のクラスに分類する必要があると考える. なお, 次章にて示すシミュレーションでは, $\Delta_t = 4$ とし, 表 1 に示す 8 種類のクラスへと分離した. ただし, 同表における Th は $e_g[i-1]$ に等しい.

3.2 R-データ, B-データの符号化

本節では, 色データの相関を利用した $r[i]$, $b[i]$ に対する符号化処理を述べる. 以下, $r[i]$ の場合について示すが, $b[i]$ の場合も同様に行うことができる.

$r[i]$ の符号化においても, 線形予測, および, クラス毎の係数は $g[i]$ と同じものを用いる. しかし, クラス分離の方法が G-データとは異なる. G-データでは, $g[i]$ のクラス分離に

$$e_g[i-1] = |g[i-1] - g[i-2]|,$$

図 5 予測に用いる参照画素
Reference pixels for prediction.

図 6 クラス分離に用いる参照画素
Reference pixels for class separation.

表 1 コンテキスト毎の予測係数
Predict coefficients in each class.

	$p(i) = 0, 1$			$p(i) = 2, 3$		
	a_1	a_2	a_3	a_1	a_2	a_3
$Th = 0$	0.75	0.25	0	0.25	0.75	0
$Th = 1$	0.6	0.4	0	0.25	0.75	0
$Th \leq 4$	0.4	0.4	0.2	0.4	0.4	0.2
$Th \geq 5$	0.2	0.7	0.1	0.7	0.1	0.2

表 2 符号量の比較 (bpp)
Comparison of bit rate (bpp).

	Method1	Method2	Method3	JPEG
MissAmerica	4.09	4.09	4.01	4.10
Claire	2.96	3.03	2.80	3.13
CarPhone	4.33	4.29	4.11	4.41

を用いた。一方、G-データの後に符号化する R-データ、B-データでは $r[i]$ のクラス分離に

$$e_g[i] = |g[i] - g[i-1]|,$$

を用いる。

$e_r[i]$ のクラス分離に $e_g[i]$ でなく $e_r[i-1]$ を用いる理由は、処理速度、および圧縮効率に関する性能向上のためである。まず、新たに $e_r[i]$ を計算するの比べて、すでに計算の済んだ $e_g[i]$ を用いることで、その分の計算が不要となり、処理時間が短縮される。さらに、こうした処理の高速化を図るため、従来法が圧縮効率を多少犠牲にしたのに対し、本手法は、処理の高速化とともに圧縮効率の向上も実現できる。圧縮効率の向上は、次の (3) 式が成り立つことから説明できる。同式の証明は付録 B に示す。なお、 $E[\cdot]$ は平均値を表す。

$$E[(e_r[i] - e_r[i-1])^2] > E[(e_r[i] - e_g[i])^2]. \quad (3)$$

これは、 $e_g[i]$ と $e_r[i-1]$ を比べた場合、 $e_r[i]$ との相関は $e_g[i]$ の方が強いことを示している。つまり、空間的な相関よりも色データ間の相関の方が強いということである。このため、 $e_r[i-1]$ よりも $e_g[i]$ を用いた方が、 $e_r[i]$ のクラス分離をより正確に行うことができ、圧縮効率の向上につながる。

表 3 圧縮処理時間の比較 (なお、処理時間は 120 フレームに要した時間)

Comparison of run time (each run time is a total time required for 120 frames).

圧縮手法	平均処理時間
Method1	101.78
Method2	79.45
Method3	76.27
JPEG	81.02

4. 実験

4.1 実験条件

提案手法の有効性、問題点を検討するために計算機シミュレーションによる圧縮実験を行った。実験において用いた画像は RGB カラー動画画像 (360×288(画素/frame)) “MissAmerica”, “Claire”, “Carphone” の第 1 フレーム～第 120 フレームであり、用いた計算機の CPU は Pentium 200MHz (5.10 SPECint95, 4.18 SPECfp95) である。2章で示した走査法に関しては $\Delta_t = 4$ とし、360×288×4 の領域を走査対象とした。

4.2 LosslessJPEG との比較

他の圧縮手法として、JPEG の lossless mode である losslessJPEG⁷⁾ を取り上げ、圧縮率・処理時間に関して比較検討した。静止画像符号化の国際標準方式 JPEG には、非可逆符号化を行う lossy mode と可逆符号化に対応した lossless mode がある。lossy mode が離散コサイ

ン変換を用いるのに対し、lossless mode では2次元の線形予測による符号化を行う。これが losslessJPEG である。用いる線形予測には、7通りの予測式が用意されており、その中で符号量が最小となるものを画像毎に選択している。予測次数は最大で3次であり、符号化処理は低次の線形予測を用いた簡単な処理となっている。ただし、本手法のように画素単位で適応的に予測係数を選択することはなく、選択された予測式は画像内のすべての画素に対して用いられる。また、予測残差の符号化にはハフマン符号あるいは算術符号が用いられる。なお、比較実験では、ハフマン符号を用いた losslessJPEG シミュレータ (ftp://ftp.cs.cornell.edu/pub/multimed/ljpg.tar.Z) を利用した。同シミュレータに添付のドキュメントによれば、画像符号化に用いた場合、汎用的な圧縮ツールである compress や gzip よりも losslessJPEG の方が優れた圧縮効率を示したとの報告がある。

このように、losslessJPEG は本手法と同様に低次の線形予測とハフマン符号を組み合わせた手法であるため、本手法の比較対象とした。

実験結果として、各手法を用いた場合の符号量、および、圧縮処理に要した時間をそれぞれ、表2、表3に示す。ここで、Method1～3は次のような違いがある。Method1はクラス分離を行わず、予測誤差の二乗和を最小にする予測係数を用いた場合である。この方法では、 $4(= \Delta_t)$ フレーム毎に予測係数を更新するための計算が必要となる。なお、この予測係数の計算法を付録Aに示す。Method2は時空間領域でのフレームの順序を表す $p(i)$ も考慮に入れ、 $Th = e_g[i-1]$ として、表1のように8種類のクラスに分離した場合であり、予測係数も同表の値を用いた。Method3は色データ間の相関も考慮した方法である。G-データに対しては、Method2と同様である。一方、R-データ、B-データに対しては、 $Th = e_g[i]$ としてクラス分離を行った。上述の Method1～Method3 の相違点を表4に示す。

また、動きを含めた評価を行うために、各手法の4フレーム毎の符号量の推移を図7に示す。ここで示した縦軸の値は4フレーム毎に算出した1画素あたりの平均符号量である。

4.3 他の走査方法との比較

ラスタ走査において、 j 番目に走査する画素の x 座標および y 座標を各々 $RAS_x[j]$, $RAS_y[j]$ とする。図2において $HIL_x[j]$, $HIL_y[j]$ の部分を $RAS_x[j]$, $RAS_y[j]$ に換えた走査をここでは、時空間ラスタ走査と呼ぶことにする。各走査により得られた1次元データに対して、(1)式による線形予測を行った結果を表5に示す。なお、予測係数は付録Aに示す計算法により求めた。時空間ヒルベルト走査を用いた場合は、4.2節で示した Method1 に一致する。図7における Raster は表5の時空間ラスタ走査に対応する結果である。ここで、走査法の違いを明確にする

図7 フレーム毎の符号量の推移
Changes of bit-rate every 4 frames.

ために、フレーム毎の符号量の推移を図9に示す。なお、図中の Method1, Raster は表5における時空間ヒルベルト走査、時空間ラスタ走査に各々対応する。

符号量に関しては、時空間ヒルベルト走査を用いた方が少ない符号量で済んでいる。これは図5(a-3)の場合に、 $g[i-f(i)-1]$ として参照される画素の違いによると考え

表 4 Method1 ~ Method3 の比較
Comparison of Method1 ~ Method3.

	Method1	Method2	Method3
クラス数	1	8	8
予測係数	正規方程式の解として算出	8通りの中から適応的に選択	同左
クラス分離の基準 (G-データ)	-	$e_g[i-1]$	同左
クラス分離の基準 (R-データ)	-	$e_r[i-1]$	$e_g[i]$
クラス分離の基準 (B-データ)	-	$e_b[i-1]$	$e_g[i]$

図 8 走査毎の符号化画素と参照画素
Coding pixels and referring pixels for each scanning.

表 5 走査法の比較 (bpp)
Comparison of scanning method (bpp).

	時空間ヒルベルト走査	時空間ラスタ走査
MissAmerica	4.09	4.11
Claire	2.98	3.07
CarPhone	4.32	4.36

る。この画素は符号化対象画素 $g[i]$ と同一フレームに存在し、走査線に沿って 2 画素分離れた距離にある画素である。一例を図 8 に示す。同一フレーム内の走査線に沿って 1 画素以上離れた画素を参照する場合、時空間ラスタ走査に比べて時空間ヒルベルト走査の方が空間的に近い画素を参照することになる。このことが、時空間ヒルベルト走査の方が符号量を少なくできた原因と考える。

このとき、計算時間は走査法によらず一定となる。これは以下の理由による。対象となる動画のフレームサイズは既知なので、走査情報 ($HIL_x[j]$, $HIL_y[j]$), および ($RAS_x[j]$, $RAS_y[j]$) を符号化に先立ち計算しておくことができる。このため、いずれの走査の場合も、符号化の際の 1 次元系列への展開はあらかじめ求めた走査情報を参照する処理となり、走査法の違いによる計算量の差は生じない。

4.4 考 察

表 2 より、Method1 ~ Method3, losslessJPEG の中で、Method3 が符号量を最も低減できる手法であることを確認した。Method1 と Method3 の比較から、クラス毎に適応的な予測を行うことにより、符号量を低減できることがわかる。クラス分離に関する Method2 と Method3 の比較結果は、異なる色データであっても空間的に近い傍の情報の方がクラス分離には有効であることを示唆している。また、表 3 より、Method3 は Method1 のような 4 フレーム毎の予測係数の計算が不要であるため、処理時間も Method1 に比べて少なくて済むことがわかる。

(a) MissAmerica

(b) Claire

(c) CarPhone

図 9 走査法の比較 (フレーム毎の符号量の推移)
Comparison of scanning method (Changes of bit-rate every 4 frames).

さらに、Method3 と losslessJPEG との比較では、両者は、ほぼ同程度の処理時間でありながら、Method3 はよ

り少ない符号量を実現できている。これは、時間方向の相関、および色データ間の相関を利用していることが効果的に働いているためと考えられる。

5. む す び

本論文では、動画像の可逆圧縮手法として、時空間領域における圧縮手法を提案した。提案手法は、時空間ヒルベルト走査により得られた1次元データに対し、適応的な線形予測を行うものである。時空間領域の走査を行い、1次元データへと展開するのは先に提案した非可逆圧縮手法と同様である。そのため、得られた1次元データに対しては、高速処理が可能な可逆圧縮・非可逆圧縮を目的別に使い分けることが可能となる。今後は、1次元化された圧縮後の画像データに対して、検索・編集等の処理を検討していく予定である。

日頃より、さまざまな形で御指導頂く本学 迫江博昭教授に深く感謝します。本研究の一部は文部省科学研究費補助金(特別研究員奨励費 No.12000423)の援助を受けた。ここに感謝の意を表します。

〔文 献〕

- 1) 坂東幸浩, 鎌田清一郎: “長方形領域を充てんする擬似ヒルベルト走査法”, 信学論, **J80-D-II**, 10, pp. 2864-2867 (Oct. 1997)
- 2) 坂東幸浩, 西修功, 鎌田清一郎: “ヒルベルト走査を用いた時空間領域分割による高速動画像圧縮法”, 映情学誌, **53**, 4, pp. 559-564 (Apr. 1999)
- 3) Y.Chan and W.Sui: “Variable Temporal-Length 3-D Discrete Cosine Transform Coding”, IEEE Trans. Image Processing, **IP-6**, 5, pp.758-763 (May 1997)
- 4) 福岡慎治, 岩橋政宏, 神林紀嘉: “可逆の色変換を用いた色彩画像の可逆符号化”, 信学論, **J81-D-II**, 11, pp. 2680-2684 (Nov. 1998)
- 5) M.J.Gormish, E.L.Schwartz, A.Keith, M.Boliek, and A.Zandi: “Lossless and Nealy Lossless Compression for High Quality Images”, SPIE, San Jose, CA, **3025**, pp. 62-70 (Feb. 1997)
- 6) D.Hilbert: “Über die stetige abbildung einer linie auf ein flächenstück”, Math. Ann., Vol. 38, pp. 459-460 (1891)
- 7) ISO/IEC 10918-1, “Digital Compression and Coding Continuous-tone Still Images: Requirements and Guidelines” (Feb. 1994)
- 8) M.S.Lazar and L.T.Brunton: “Fractal Block Coding of Digital Video”, IEEE Trans. Circ. and Syst., **CAS-4**, 3 pp. 297-308 (June 1994)
- 9) N.D.Memon and K.Sayood: “Lossless Compression of RGB Color Images”, Optical Engineering, **34**, 6, pp. 1711-1717 (June 1995)
- 10) N.D.Memon and K.Sayood: “Lossless Compression of Video Sequences”, IEEE Trans. Commun., **COM-44**, 10, pp. 1340-1345 (Oct. 1996)
- 11) H.G.Musmann, P.Persch and H.J.Grallert: “Advances in Picture Coding”, Proc. IEEE, **PROC-73**, pp. 523-548 (Apr. 1985)
- 12) A.N.Netravali and B.G.Haskell: “Digital Pictures: Representation, Compression, and Standards 2nd edition”, Plenum Publishing Corporation (1994)
- 13) 小川一人, 加藤隆, 上原年博, 和泉吉則, 田中豊: “マルコフモデルを用いた動画像のロスレス符号化”, 映情学誌, **53**, 4, pp. 583-590 (Apr. 1999)
- 14) 太田睦, 大網亮磨: “ロスレス DCT 画像符号化方式”, テレビ誌, **50**, 8, pp. 1162-1171 (Aug. 1996)
- 15) C.I.Podilchuk, N.S.Jayant and N.Farvardin: “Three-Dimensional Subband Coding of Video”, IEEE Trans. Image Processing, **IP-4**, 4, pp.125-139 (Feb. 1995)
- 16) H.ザーガン著, 鎌田清一郎 訳: “空間充填曲線とフラクタル”, シュプリンガー・フェアラーク東京 (1998)
- 17) P.Willemin, T.R.Redd and M.Kunt: “Image Sequence Coding by Split And Merge”, IEEE Trans. Commun., **COM-39**, 12, p-p. 1845-1855 (Dec. 1991)

〈付 録〉

A. 予測誤差の二乗和を最小にする係数

(1) 式の線形予測において、予測残差の二乗和を最小化する予測係数の計算法を示す。1組の係数(a_1, a_2, a_3)に対する二乗誤差を

$$\rho = \sum_{i=0}^{\ell_x \times \ell_y \times \Delta_t - 1} (g[i] - \hat{g}[i])^2,$$

とおくと、 ρ は (1) 式の係数 a_1, a_2, a_3 の関数である。二乗誤差 ρ を最小化する係数は次の連立方程式を満たす。

$$\begin{cases} \frac{\partial \rho}{\partial a_1} = 0, \\ \frac{\partial \rho}{\partial a_2} = 0, \\ \frac{\partial \rho}{\partial a_3} = 0. \end{cases}$$

これを解いて a_1, a_2, a_3 は以下のように求まる。

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = U^{-1}b.$$

ここで、 U^{-1} , b は

$$v(m_1, m_2) = \sum_{i=\min(m_1, m_2)}^{\ell_x \times \ell_y \times \Delta_t - 1} g[i - m_1]g[i - m_2],$$

$$\min(m_1, m_2) = \begin{cases} m_1 & (m_1 < m_2), \\ m_2 & (\text{上記以外の場合}), \end{cases}$$

を用いて次式のように表される。

$$b = \begin{bmatrix} v(0, 1) \\ v(0, f(i)) \\ v(0, f(i) + 1) \end{bmatrix},$$

$$U = \begin{bmatrix} v(1, 1), & v(1, f(i)), & v(1, f(i) + 1) \\ v(1, f(i)), & v(f(i), f(i)), & v(f(i), f(i) + 1) \\ v(1, f(i) + 1), & v(f(i), f(i) + 1), & v(f(i) + 1, f(i) + 1) \end{bmatrix}.$$

B. (3) 式の証明

以下、 $e_g[i]$, $e_r[i]$ について平均、分散を各々

$$\begin{cases} \bar{e}_r = E[e_r[i]], \\ \bar{e}_g = E[e_g[i]], \\ V_r = E[(e_r[i] - \bar{e}_r)^2], \\ V_g = E[(e_g[i] - \bar{e}_g)^2], \end{cases}$$

とし、 $e_r[i]$ と $e_r[i-1]$ との間の相関係数を

$$\gamma_{rr} = \frac{E[(e_r[i] - \bar{e}_r)(e_r[i-1] - \bar{e}_r)]}{V_r},$$

$e_r[i]$ と $e_g[i]$ との間の相関係数を

付・表 1 各画像の統計量
Statistical values for each image.

(a) CarPhone				
色データ	$\bar{e}_c$	V_c	γ_{cc}	γ_{cg}
G	4.93	88.3	0.352	—
R	4.54	89.7	0.362	0.955
B	4.85	93.7	0.359	0.929
平均	4.77	90.6	0.361	0.942
(b) Claire				
色データ	$\bar{e}_c$	V_c	γ_{cc}	γ_{cg}
G	2.04	88.28	0.352	—
R	4.54	89.7	0.362	0.955
B	4.85	93.7	0.359	0.929
平均	4.77	90.6	0.361	0.942
(c) Missamerica				
色データ	$\bar{e}_c$	V_c	γ_{cc}	γ_{cg}
G	5.21	29.7	0.230	—
R	3.17	17.1	0.342	0.415
B	3.76	23.1	0.321	0.623
平均	4.05	23.3	0.332	0.519

$$\gamma_{rg} = \frac{E[(e_g[i] - \bar{e}_g)(e_r[i] - \bar{e}_r)]}{\sqrt{V_r} \sqrt{V_g}},$$

とおく. このとき, $E[(e_r[i] - \bar{e}_r)(e_r[i-1] - \bar{e}_r)]$ および $E[(e_r[i] - \bar{e}_r)(e_g[i] - \bar{e}_g)]$ は次のようになる.

$$\begin{aligned} & E[(e_r[i] - \bar{e}_r)(e_r[i-1] - \bar{e}_r)] \\ &= E[\{(e_r[i] - \bar{e}_r) - (e_r[i-1] - \bar{e}_r)\}^2] \\ &= E[(e_r[i] - \bar{e}_r)^2 + E[(e_r[i-1] - \bar{e}_r)^2] \\ &\quad - 2E[(e_r[i] - \bar{e}_r)(e_r[i-1] - \bar{e}_r)]] \\ &= 2E[(e_r[i] - \bar{e}_r)^2] - 2E[(e_r[i] - \bar{e}_r)(e_r[i-1] - \bar{e}_r)] \\ &= 2(1 - \gamma_{rr})V_r, \end{aligned}$$

$$\begin{aligned} & E[(e_r[i] - \bar{e}_r)(e_g[i] - \bar{e}_g)] \\ &= E[\{(e_r[i] - \bar{e}_r) - (e_g[i] - \bar{e}_g) + (\bar{e}_r - \bar{e}_g)\}^2] \\ &= V_r + V_g + (\bar{e}_r - \bar{e}_g)^2 - 2\gamma_{rg}\sqrt{V_r}\sqrt{V_g} \\ &\quad + 2(\bar{e}_r - \bar{e}_g)E[(e_r[i] - \bar{e}_r)] - 2(\bar{e}_r - \bar{e}_g)E[(e_g[i] - \bar{e}_g)] \\ &= V_r + V_g + (\bar{e}_r - \bar{e}_g)^2 - 2\gamma_{rg}\sqrt{V_r}\sqrt{V_g} \\ &\quad \text{ここで } \bar{e}_r \simeq \bar{e}_g, V_r \simeq V_g \text{ と仮定すると,} \\ &\simeq 2(1 - \gamma_{rg})V_r. \end{aligned}$$

付・表 1 に, 各画像の統計量を色データ毎に示す. なお, 表中の e_c , V_c , γ_{cc} , γ_{cg} の添字 c は色データの成分を表しており, R-データの場合では, $c = r$ とした e_r , V_r , γ_{rr} , γ_{rg} に対応する. この結果, $E[(e_r[i] - \bar{e}_r)(e_g[i] - \bar{e}_g)]$ の最後の式変形で用いた仮定が妥当であること, および, 相関係数は

$$\gamma_{rr} < \gamma_{rg},$$

の関係にあることが確認される. この結果, (3) 式が示された. $\square$

ばんどう ゆきひろ
坂東 幸浩 1996 年, 九州大学工学部情報工学科卒業. 1998 年, 同大学院修士課程修了. 現在, 同大学院博士後期課程在学中. 画像処理の研究に従事. 平成 12 年度日本学術振興会特別研究員, 学生会員.

よこやま たかまさ
横山 貴政 2000 年, 九州大学工学部情報工学科卒業. 同年, 同大学院修士課程入学. 現在, 同大学院修士課程在学中. 画像処理の研究に従事.

かまた せいじろう
鎌田清一郎 1983 年, 九州工業大学工学部情報工学科卒業. 1985 年, 九州大学大学院修士課程修了. 同年, 日本電気(株)入社. 1988 年, 九州工業大学情報工学科助手. 1996 年, 九大大学院システム情報科学研究科助教授. 1990 年, 米国州立メイン大学客員研究員. 画像表現, パターン認識, 画像処理等の研究に従事. 工学博士. 正会員.