

解像度変換に適した可逆画像圧縮形式の検討

北郷正輝[†] 萩原一郎[†](正会員)[†] 東京工業大学

あらまし 現在、デジタル画像の入出力デバイスの解像度は多岐に渡っており、それぞれの入出力デバイスの性能を活かすには画像の解像度変換（拡大/縮小）が必要である。本研究では、画像を開数表現し、関数データを保存する画像情報の圧縮形式“Function Image Compression (FIC)”を提案する。FIC は画像を開数表現するため、解像度変換は関数のリサンプリング間隔で決定され、解像度変換ごとの画像劣化はない。つまり、画像補間と画像圧縮を同時にできる、解像度変換に適した画像圧縮形式といえる。ここでは FIC の性能を、画像の拡大精度、解像度変換における計算時間、画像圧縮率の三点から評価し、提案手法の有効性、実用性を示した。

キーワード：可逆圧縮、解像度変換、関数表現、シフト線形補間

Summary As the need to use various devices increases, we need to produce the compressed image at different resolutions. The development of an image compression format that compression and resolution conversion can be performed simultaneously is desired. In this research, we suggest a lossless image compression format called “Function Image Compression (FIC)”. Since FIC has function representation of an image data, resolution conversion can be easily performed because resampling is easy once it performs function representation. We show the results of zooming images, calculation costs for resolution conversion, and image compression ratios for the evaluation of FIC system.

Key words: Lossless compression, Resolution conversion, Function representation, Shifted linear interpolation

1. はじめに

情報のデジタル化、ネットワーク化が進むにつれ、デジタルデータとしての画像情報は広く普及している。デジタル画像の入出力デバイスには、PC のディスプレイ、デジタルカメラ、携帯電話など様々な解像度のもがあり、多様なデバイス間で画像を受け渡す機会が増えている。それぞれの入出力デバイスの性能を活かすには画像の解像度変換（拡大/縮小）が必要である。

現在、画像の解像度変換にはニアレストネイバ法、バイリニア法、バイキュービック法が使われている^{1),2)}。ニアレストネイバ法、バイリニア法は処理が速く容易に利用できるが、精度については十分といえない。ニアレスト

ネイバ法は原画像のコントラストは保たれるがジャギー（ギザつき）が目立つ、バイリニア法は平滑化の効果が得られジャギーは発生しないがボケた拡大画像になってしまう、という問題点がある。また、バイキュービック法は三次関数補間を行うため、精度は良いが処理が遅いという問題点がある。

現存する画像圧縮形式には JPEG 形式、TIFF 形式、PNG 形式などがある^{3),4)}。これらの形式はデータをピクセルごとに保存しており、解像度変換ごとに画像補間法を用いて計算する必要が生じる。特に、JPEG 形式は非可逆圧縮なので解像度変換をして画像を保存する度に画像は劣化する。従って、従来の画像圧縮形式では、画像は解像度変換ごとに補間法に依存し、また、保存することに圧縮形式に依存して劣化する。すなわち、従来の画像圧縮形式は画像を様々な解像度のデバイスで扱うには適していないといえる。多様なデバイスで画像を扱うには、画像の解像度変換と画像圧縮を繰り返しても劣

“Study on a Lossless Image Compression Format Suitable for Resolution Conversion” by Masaki KITAGO and Ichiro HAGIWARA (Member) (Tokyo Institute of Technology Graduate School of Science and Engineering).

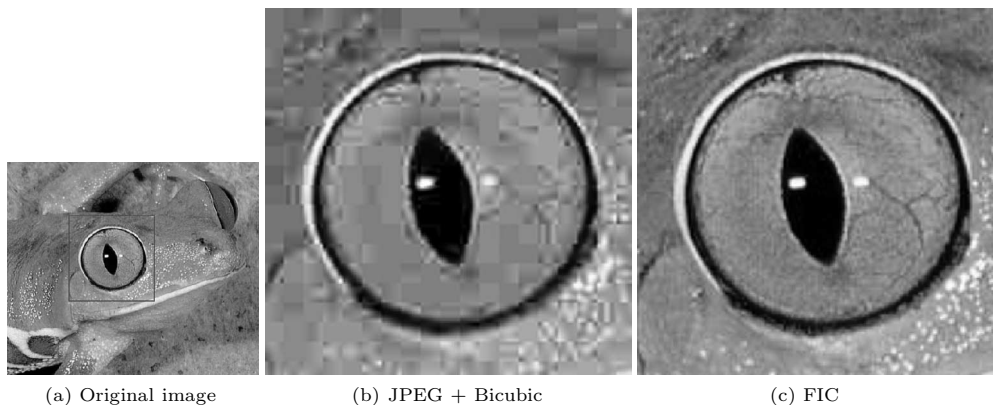


図1 解像度変換

Fig.1 Resolution conversion

化の少ない画像圧縮形式が必要である。

本研究では、画像を関数表現し、関数データを保存する画像情報の圧縮形式“Function Image Compression (FIC)”を提案する。画像が関数で表現されるため、解像度変換は関数のリサンプリング間隔で決定され、解像度変換ごとの画像劣化はない。つまり、画像補間と画像圧縮を同時にできる、解像度変換に適した画像圧縮形式といえる。ここでは、高詳細な拡大画像を生成する目的から、圧縮過程で画像劣化のない可逆圧縮のFICを開発する。

従来手法とFICの解像度変換の比較を図1に示す。(a)は原画像、(b)、(c)は原画像の眼(150×150 [pixel])をそれぞれの手法で3倍(450×450 [pixel])に拡大した画像である。(b)はJPEGで画像圧縮した後、バイキュービック法で画像補間を行い、(c)はFICで画像圧縮、補間を共に行った。(b)はJPEG圧縮の際、ブロックノイズが生じてしまい、バイキュービック法による拡大でノイズが顕著に現れてしまっている。対して(c)の拡大画像は、画像の特徴を良く捉え、エッジなど詳細部分の拡大も自然に行えている。FICは、可逆圧縮のため圧縮時における画像劣化がなく、関数表現により解像度変換ごとの画像劣化がないためである。

本稿の構成として、2章では、画像補間手法、可逆圧縮手法、可逆画像圧縮形式、解像度変換を目的とした画像圧縮形式について関連研究を述べる。3章では、FICの理論について、関数表現の手法、可逆圧縮のアルゴリズムを示す。4章では、FICの性能を、画像の拡大精度、解像度変換における計算時間、画像圧縮率の三点から評価する。

2. 関連研究

一般に画像補間法には、ニアレストネイバ法、バイリニア法、バイキュービック法が使われている^{1),2)}。より高次の多項式やB-spline関数を用いた手法^{3),6)}も提案されているが、良い拡大精度が期待される反面、計算量の増大や、オーバーフィッティングにより不自然な結果を示す可能性がある、という問題点もある。また、線形補間法の基底関数がある一定量シフトさせることにより、三次関数補間と同等の精度を線形補間と同等の計算量で実現するシフト線形補間法^{7),8)}が提案されている。これらの手法は、計算量の面で利点があり、関数情報のフォーマット化などの実用面に向いていると考えられる。しかし、原画素に対して関数をフィッティングするため、原画素間にエッジがある場合などには対応することができず、エッジ部分にジャギー、ボケが発生する原因となる。

画像拡大におけるこれらの問題点を改善するため、数多くの手法が提案されている。エッジ情報を考慮して拡大精度を高める手法^{9),10)}、レベルセット法を用いて反復的に等輝度線を再構成する手法^{11),12)}、三角形メッシュを再構成してエッジに適合する手法^{13),14)}、超解像手法としてベイズ推定を用いた手法¹⁵⁾などが挙げられる。

可逆圧縮手法については、LZ法の一種であるLZW法や、LZ法とハフマン法を組み合わせたLHAやZIPが一般に普及している¹⁶⁾。近年では、算術圧縮法と同等の圧縮率を、より高速に実現できるRange Coder法¹⁷⁾、データを圧縮に効率の良い配列に変換するBlock-Sorting法¹⁸⁾、データを確率構造として捉えることで圧縮するPrediction by Partial Matching (PPM)法^{19)~21)}など、より効果的な圧縮法に注目が集まっている。可逆画像圧縮形式においては、JPEG-LS^{22),23)}が代表的である。次世代画

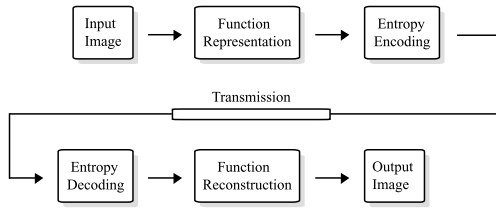


図 2 FIC システム構成
Fig. 2 The structure of FIC system

像圧縮形式として規格化された JPEG2000^{(24),(25)} には Lossless 機能が標準採用されている。

解像度変換を目的とした画像圧縮形式では、ウェーブレット変換の多重解像度解析を用いて画像圧縮する JPEG2000^{(24),(25)} が代表的である。また(株)セラーテムテクノロジーが開発した、画像の輝度変化ベクトルを圧縮する、画像ズームに適した画像圧縮形式 Pixel Live がある⁽²⁶⁾。JPEG2000 はウェーブレット変換の多重解像度解析を用いているため、画像のサムネイルなど縮小処理に優れているが、高倍率の拡大への対応は困難である。対して、Pixel Live 形式は 1200% までの高倍率な拡大が可能であるが、圧縮率は原画像の約 2/3 程度と圧縮面で問題が残る。

本研究では、高倍率の拡大精度が良く、短時間で解像度変換ができ、圧縮率が高い画像圧縮形式の開発を目的とする。

3. FIC 理論

本章では、FIC 理論について述べる。FIC のシステム構成図を図 2 に示す。FIC システムは関数表現部とエントロピー符号化部の二つの処理から構成される。関数表現はシフト線形補間法^{(7),(8)} を基に関数を生成し、エントロピー符号化には PPM 法⁽¹⁹⁾⁻⁽²¹⁾ を用いる。3.1 節では、シフト線形補間法を概説し、FIC 関数表現への適応手法について述べる。3.2 節では、関数情報を圧縮に適した係数列に変換し、PPM 法で圧縮するまでのアルゴリズムを示す。

3.1 シフト線形補間法

本節では、シフト線形補間法を概説し、FIC 関数表現への適応手法について述べる。

ある関数 $f(x)$ を標準化間隔 T で離散化して得られるサンプル値列を $f[n] = f(nT)$ とする。このとき、線形補間法で得られる連続関数 $f_L(x)$ は、

$$f_L(x) = \sum_{n \in \mathbb{Z}} f[n] \varphi\left(\frac{x}{T} - n\right), \quad (1)$$

と表され、図 3 の破線で示されるように、単純にサンプル値列を直線で結んで得られる関数となる。基底関数 $\varphi(x)$

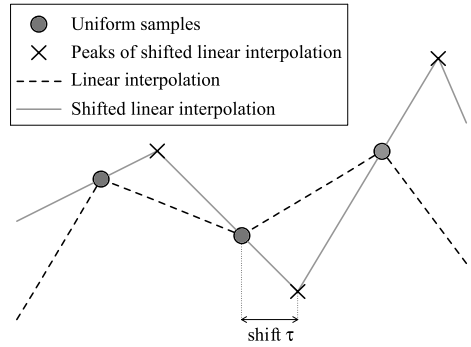


図 3 シフト線形補間
Fig. 3 Shifted linear interpolation

は以下のように表される。

$$\varphi(x) = \begin{cases} 1 - |x|, & |x| \leq 1 \\ 0, & |x| > 1. \end{cases} \quad (2)$$

次に、シフト線形補間法で得られる関数 $f_S(x)$ は以下の式で表される。

$$f_S(x) = \sum_{n \in \mathbb{Z}} c[n] \varphi\left(\frac{x}{T} - n - \tau\right), \quad (3)$$

ここで、 τ はシフトパラメータである。係数列 $\{c[n] | n \in \mathbb{Z}\}$ は再帰的に以下の式で導出される。

$$c[n] = -\frac{\tau}{1-\tau} c[n-1] + \frac{1}{1-\tau} f[n]. \quad (4)$$

シフト線形補間法で得られる関数は図 3 の実線で示される。図 3 では $\tau = 0.4$ を用いており、 τ の値として一般に約 0.21 が推奨されている⁽⁷⁾。

以下に、シフト線形補間法の FIC 関数表現への適応手法について述べる。

まず、図 3 の \times で示される、式 (4) で導出された係数列 $c[n]$ を整数頂点列 $\{d[n] | n \in \mathbb{Z}\}$ に変換する。このとき、シフト線形関数は原画素を完全には通らなくなるが、関数再構成で再び離散化処理することで原画素を完全に復元できる。変換は以下のように表される。

$$d[n] = \lfloor c[n] + 0.5 \rfloor, \quad (5)$$

ここで、 $\lfloor \cdot \rfloor$ はフロア関数 (Floor Function) である。そして、整数頂点列 $d[n]$ の差分値が FIC 係数列 $\{D[n] | n \in \mathbb{Z}\}$ となる。

$$D[n] = d[n] - d[n-1]. \quad (6)$$

また FIC では、 τ は推奨値である 0.21 を採用する。二次元への展開については、横方向の係数のみを符号化し、縦方向の係数は復号化時に計算して保持する。以上の処理で一度関数表現をしてしまえば、解像度変換に必要な係数は単純な線形計算で導出できる。またカラー画像へは、RGB 空間において各色を個別に関数表現することで適用可能である。

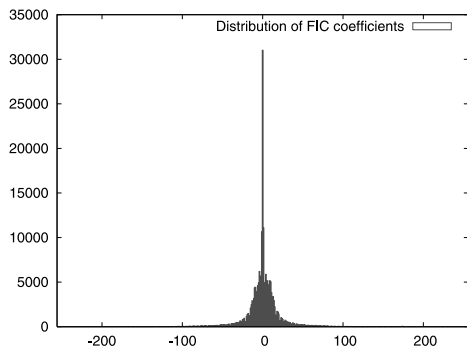


図4 FICの係数分布(処理前)

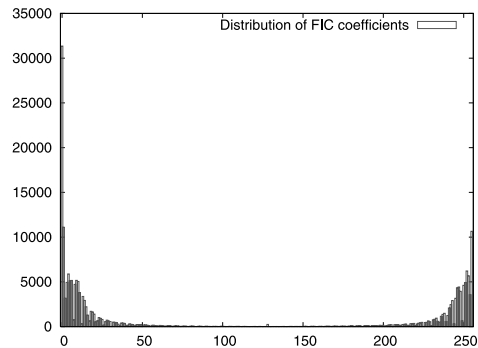
Fig. 4 Distribution of FIC coefficients
(Before processing)

図5 FICの係数分布(処理後)

Fig. 5 Distribution of FIC coefficients
(After processing)

3.2 エントロピー符号化

本節では、式(6)で導出されたFIC係数列 $D[n]$ を圧縮に適した係数列に変換し、PPM法^{19) - 21)}で圧縮するまでのアルゴリズムを示す。

まず、FIC係数列 $D[n]$ について検討する。Lena(256×256 [pixel])のFIC係数分布を図4に示す。横軸がFIC係数の値であり、縦軸がFIC係数の数である。0に最大ピークがあり、0付近に係数が集中している。そして、係数はほとんど-100から100までに分布していることがわかる。また、 $D[n]$ の差分前の情報 $d[n]$ は、ほとんど0から255内に入っていることが特徴として挙げられる。これらの係数分布の特徴は、Lena以外の画像に対しても一般に成り立つことが確認されている。

FIC係数における幾つかの特徴から、圧縮に適した係数列に変換するアルゴリズムを以下に示す。

符号化処理

1. 1バイト係数リスト, 2バイト係数リストをそれぞれ用意する。
2. $|D[n]| > 255$, または $d[n] < 0$, または $d[n] > 255$ のとき, $D[n]$ を2バイト係数リストに書き込み, 1バイト係数リストに128を書き込む。
3. $D[n] = 128$ のとき, 2バイト係数リストに0を書き込み, 1バイト係数リストに128を書き込む。
4. 2以外で $D[n] < 0$ のとき, $D[n]$ に256を加算し, 1バイト係数リストに書き込む。このとき, 加算した値が128であれば, 2バイト係数リストに0を書き込む。
5. それ以外のとき, $D[n]$ を1バイト係数リストに書き込む。

復号化処理

1. 1バイト係数リストの値が128のとき, 2バイト係数リストを参照する。

- (a) 2バイト係数リストが0以外するとき, 2バイト係数リストの値を $D[n]$ とする。
 - (b) 2バイト係数リストが0のとき, $d[n] > 255$ であれば256を減算し, それ以外はそのままの値を $D[n]$ とする。
2. 1以外するとき, $d[n] > 255$ であれば256を減算し, それ以外はそのままの値を $D[n]$ とする。

以上の処理により, すべて2バイトで表現されていたFIC係数が, 1バイト係数と少数の2バイト係数に変換される。ちなみにLena(256×256 [pixel])では, 処理前の2バイト係数は197,376個, 処理後の1バイト係数は197,376個, 2バイト係数は244個である。1バイト係数の分布を図5に示す。横軸がFIC係数の値であり, 縦軸がFIC係数の数である。0, 255にそれぞれピークがあり0, 255付近に係数が集中している。また, 2バイト係数リストの参照に用いる128に低いピークを確認できる。

結果として, 本アルゴリズムでFIC係数を1バイト係数リストと2バイト係数リストへ効率的に変換することにより, 少ない情報量で画像を関数表現することが可能となる。更に, 1バイト係数リストは係数情報に偏りを持つためエントロピー符号化の圧縮効率が高いと思われる。以下に図5に示される1バイト係数リストと図6に示されるLena(256×256 [pixel])の色分布のエントロピーを比較する。図6は横軸が画素値であり, 縦軸が画素数である。エントロピーは以下の式で表される。

$$\text{entropy} = \sum_{i=1}^N p_i \log_2 \frac{1}{p_i}, \quad (7)$$

ここで, N はシンボル数, p_i は出現確率を示す。エントロピーはそれぞれ, 1バイト係数リストで5.64ビット, Lenaの色分布で7.30ビットを示し, 本アルゴリズムによるエントロピーの向上を確認できた。また, 図6に示される原画素を単純に差分した系列のエントロピーは4.95

ビットを示した。本アルゴリズムと、線形補間法の符号化と等価である単純な差分系列は、共に近隣画素を参照しているため原画素のエントロピーを下げられる。本アルゴリズムより単純な差分系列がエントロピーを下げられる理由として、図 3 に示されるようにシフト線形補間法が線形補間法より傾きを大きくつけるため、差分値が多少分散してしまうことが考えられる。

最後に 1 バイト係数リストと 2 バイト係数リストを合わせて PPM 法でエントロピー符号化することで、FIC

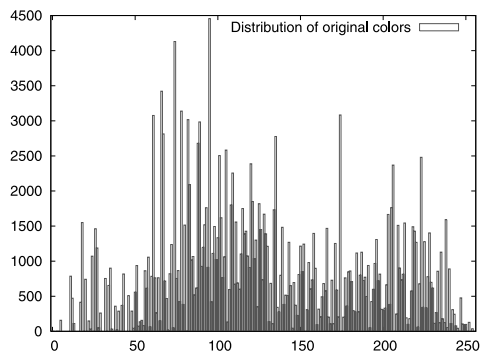


図 6 Lena (256×256) の色分布
Fig. 6 Color distribution of Lena (256×256)

システムのエントロピー符号化部となる。PPM 法はデータを確立構造として捉えることで圧縮するため、相関の高い系列を生成することで圧縮効率を高めることができる。FIC のカラー画像では、RGB 空間それぞれで生成した FIC 係数の相関が高いことから FIC 係数を R1, G1, B1, R2, G2, B2, ... のように RGB の順に符号化する。

4. 計算機実験

本章では FIC の性能を、画像の拡大精度、解像度変換における計算時間、画像圧縮率の三点から評価する。

ここで利用した計算機の環境は Pentium4 3.2GHz, 2GB RAM, Windows XP である。

4.1 拡大画像評価

本節では、FIC の拡大精度を他手法と比較、評価する。図 7 は、(a) の黒枠で示される Lena の眼 (32×32 [pixel]) を 8 倍 (256×256 [pixel]) に拡大した画像の比較である。(b) はニアレストネイバ法、(c) はバイリニア法、(d) はバイキュービック法、(e) はシフト線形補間法、(f) は FIC でそれぞれ拡大した画像である。(b) のニアレストネイバ法はジャギーが目立ち画像の滑らかさが欠如している。(c) のバイリニア法はジャギーは発生しないが画像にボケが生じてしまっている。それに対し、(d) のバ

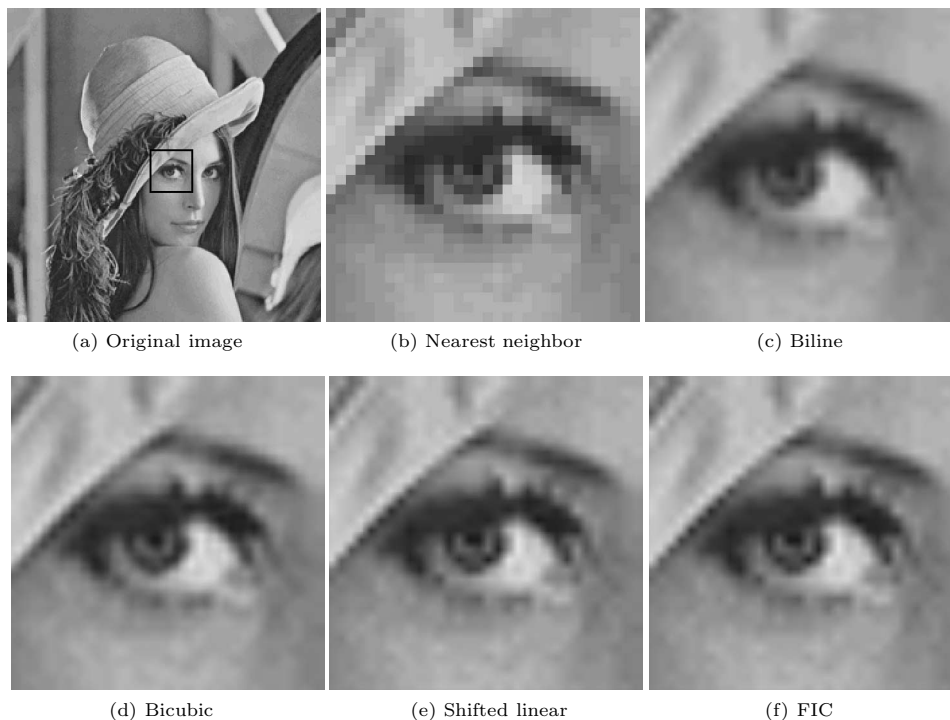


図 7 拡大画像
Fig. 7 Zooming images

表 1 計算時間 [s]
Table 1 CPU-time [s]

Lena (256×256)	Nearest	Bilinear	Bicubic	Shifted	FIC
1024×1024	0.147	0.169	0.275	0.234	0.215
2048×2048	0.566	0.644	1.031	0.801	0.788
Lena (512×512)					
1024×1024	0.197	0.244	0.375	0.322	0.256
2048×2048	0.656	0.766	1.200	1.031	0.894

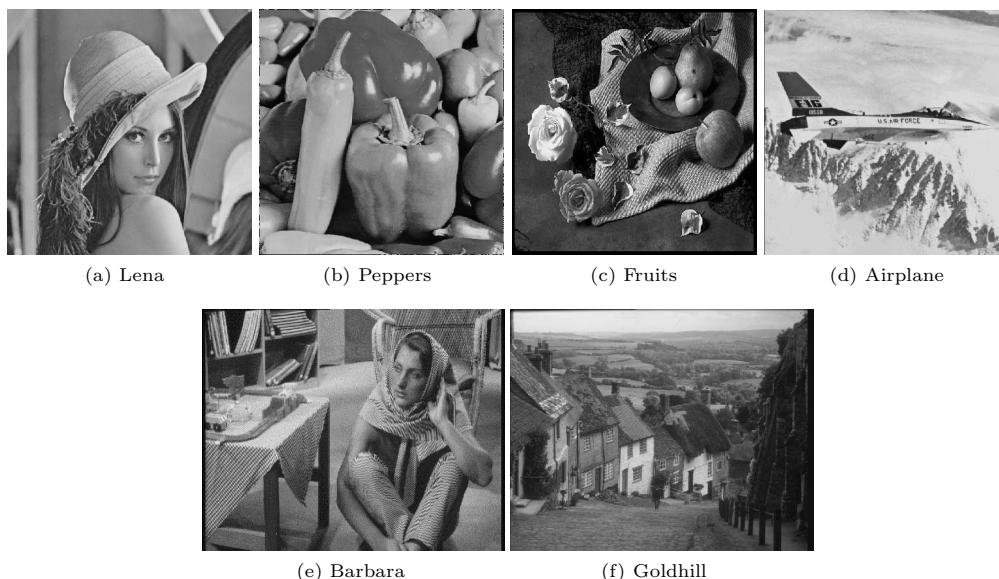


図 8 圧縮率評価に用いる原画像

Fig. 8 Original images for the evaluation of compression ratio

イキュービック法, (e) のシフト線形補間法, (f) の FIC は鮮明な拡大画像を生成している。

また, (e) の線形補間法に対して, (f) の FIC はシフト線形補間関数の係数情報を整数値に離散化しているが, 生成された拡大画像の違いは視認できない。(d) のバイキュービック法と (f) の FIC を比較すると, 画像コントラストにおいて FIC の方が優れているように見える。

つまり FIC による画像拡大は, バイキュービック法と同等の拡大精度を持ち, 画像のコントラストに関してはバイキュービック法より優れた性能を持つことがわかった。

4.2 計算時間評価

本節では, FIC の解像度変換における計算時間を他手法と比較, 評価する。解像度変換における計算時間の比較を表 1 に示す。原画像には Lena (256 × 256 [pixel]) と Lena (512 × 512 [pixel]) を用いて, 1024 × 1024 [pixel] と 2048 × 2048 [pixel] にそれぞれ拡大した。比較する拡大手法は, ニアレストネイバ法, バイリニア法, バイキュービック法, シフト線形補間法, FIC である。すべ

での拡大において, ニアレストネイバ法, バイリニア法, FIC, シフト線形補間法, バイキュービック法の順に計算時間がかかっている。

拡大精度の考察と合わせると, FIC はバイリニア法とバイキュービック法の間段階の計算量で, バイキュービック法と同等の拡大精度を出せるといえる。また, FIC はシフト線形補間法の関数係数を保持しているため, シフト線形補間法と同等の拡大精度をより少ない計算量で生成できている。

4.3 圧縮率評価

本節では, FIC の画像圧縮率を他手法と比較, 評価する。圧縮率の比較に用いた画像を図 8 に示す。圧縮に用いた画像はそれぞれ, (a) Lena, (b) Peppers, (c) Fruits, (d) Airplane, (e) Barbara, (f) Goldhill である。画像解像度は, (a) から (d) が 512 × 512 [pixel] で, (e) と (f) が 720 × 576 [pixel] である。表 2 に図 8 の画像を用いた可逆圧縮の圧縮率の比較を示す。従来の可逆画像圧縮形式として, LZW 圧縮を用いた TIFF, ZIP 圧縮を

表 2 可逆圧縮率 [%]
Table 2 Lossless compression ratio [%]

	Lena (512×512)	Peppers (512×512)	Fruits (512×512)	Airplane (512×512)	Barbara (720×576)	Goldhill (720×576)
TIFF(LZW)	84.4	46.0	87.9	18.6	47.5	69.0
TIFF(ZIP)	71.7	35.3	77.6	17.3	36.5	60.4
PNG	67.7	38.0	71.0	20.8	36.1	65.4
JPEG-LS	56.6	48.6	63.7	45.1	80.7	58.6
JPEG2000	56.6	55.2	47.9	47.3	85.3	54.1
FIC	67.4	34.6	67.8	34.6	66.7	62.0

用いた TIFF, PNG, ロスレス JPEG (JPEG-LS), ロスレス JPEG2000 を用いて FIC と圧縮率の比較を行った。ここで圧縮率は以下のように表される。

$$\text{圧縮率} = \frac{\text{圧縮画像のファイルサイズ}}{\text{原画像のファイルサイズ}} \times 100 [\%]. \quad (8)$$

圧縮手法の違いにより圧縮に向いている画像と不向きな画像に分かれている。LZW 圧縮を用いた TIFF, ZIP 圧縮を用いた TIFF, PNG は Peppers, Airplane, Barbara のような色数が少なく, 色の分布に規則性のある画像の圧縮に向いているが, 色数が多く, 画像に規則性の少ない Lena, Fruits のような画像の圧縮には向いていないことがわかる。

JPEG-LS, JPEG2000 は Lena, Fruits, Airplane, Goldhill のような色が滑らかに変化する画像の圧縮に向いており, Barbara のように色の変化が大きい画像では圧縮率が下がっている。対して, FIC は画像の向き, 不向きが少なく常に安定した圧縮率を示している。Peppers においては, 従来手法と比較して最も良い圧縮効率を示している。全体的に圧縮率はそれ程高くないが, 他手法と比較しても遜色のない圧縮性能といえる。

5. ま と め

本研究では, 解像度変換に適した画像圧縮形式の開発を目的として, 画像を開数表現し開数データを保存する画像圧縮形式 “Function Image Compression (FIC)” を提案した。FIC は画像を開数表現するため, 解像度変換ごとの画像劣化はない。つまり, 画像補間と画像圧縮を同時にできる, 解像度変換に適した画像圧縮形式といえる。

FIC の性能評価では, 拡大精度については, バイキュービック法と同等の拡大精度を持ち, 画像のコントラストに関してはバイキュービック法より優れた性能を持つことが示された。解像度変換における計算時間については, バイリニア法とバイキュービック法の中間程の計算量で解像度変換ができることを示した。そして画像圧縮率に

ついては, 解像度変換を目的としていない従来の画像圧縮形式と比較しても遜色のない圧縮性能を示した。

つまり FIC は, 高倍率の拡大に対応し, 解像度変換が高速で, 圧縮性能の良い画像圧縮形式といえる。解像度変換に適した画像圧縮形式として, 十分実用に耐えうる性能を持つと思われる。

6. 将 来 課 題

FIC で用いるシフト線形補間法は一次関数補間であるが, 三次関数補間を行うバイキュービック法と比較しても, 画像のコントラストにおいて優れた結果を示した。しかしエッジ部分のジャギーに関しては, 特に斜めのエッジに対してバイキュービック法よりジャギーが目立つという問題点も確認された。この問題は, シフト線形補間法が一次関数を利用していること, シフトした画素値が適切な範囲 (0 から 255) に収まらない場合があることから生じていると思われる。画像のコントラストとジャギーはトレードオフの関係ではあるが, 拡大時に FIC 開数表現のエッジ情報を考慮するアルゴリズムを加えることで画像の拡大精度を改善したいと考えている。

また, 画像圧縮率においては, 3.2 節で提案したアルゴリズムに受け渡す前の処理で, 係数列 $D[n]$ の配列の仕方によって圧縮率に多少ばらつきが出ることが確認された。今後は配列方法を検討し, テンプレート化など更に圧縮効率を高めるアルゴリズムの開発を考えている。また, シフトした画素値列 $d[n]$, あるいはその差分系列 $D[n]$ を符号化対象とすることによる情報エントロピーへの影響についての検討をテスト画像を増やし行う。

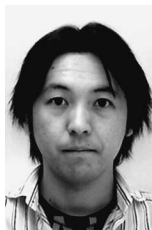
参 考 文 献

- 1) P. Thévenaz, T. Blu, and M. Unser: “Image Interpolation and Resampling”, Handbook of Medical Imaging, Processing and Analysis, I.N. Bankman, Ed., Academic Press, San Diego, CA, USA, pp.393-420 (2000).
- 2) P. Thévenaz, T. Blu, and M. Unser: “Interpolation Revisited”, IEEE Transaction on Medical Imaging, Vol.19, No.7, pp.739-758 Jul. (2000).
- 3) R.C. Gonzalez and R.E. Woods: Digital Image Pro-

- cessing Second Edition. Prentice-Hall, New Jersey, USA (2002).
- 4) G.K. Wallace: "The JPEG Still Picture Compression Standard", Communications of the ACM, Vol.34, No.4, pp.30-44 Apr. (1991).
 - 5) M. Unser: "Splines: A Perfect Fit for Signal and Image Processing", IEEE Signal Processing Magazine, Vol.16, No.6, pp.22-38 Nov. (1999).
 - 6) M. Unser: "Splines: A Perfect Fit for Medical Imaging", In Proceedings of the SPIE International Symposium on Medical Imaging: Image Processing (MI'02), Vol.4684, Part I, pp.225-236, San Diego, CA, USA Feb. (2002).
 - 7) T. Blu, P. Thévenaz, and M. Unser: "How a Simple Shift Can Significantly Improve the Performance of Linear Interpolation", In Proceedings of the 2002 IEEE International Conference on Image Processing (ICIP'02), pp.III. 377-III. 380, Rochester, NY, USA (Sep. 2002).
 - 8) 市毛弘一, T. Blu, and M. Unser: "スプライン関数とそのデジタル信号処理・画像処理への応用に関する研究", The Telecommunications Advancement Foundation, Vol.18, No.7, pp.358-365 Jan. (2003).
 - 9) S. Battiato, G. Gallo, and F. Stanco: "A New Edge-Adaptive Zooming Algorithm for Digital Images", In Proceedings of 4th IASTED International Conference Signal Processing and Communications (SPC 2000), pp.144-149, Marbella, Spain Sep. (2000).
 - 10) X. Li and M.T. Orchard: "New Edge-Directed Interpolation", IEEE Transactions on Image Processing, Vol.10, No.10, pp.1521-1527 Oct. (2001).
 - 11) B.S. Morse and D. Schwartzwald: "Isophote-Based Interpolation", In Proceedings of the 1998 IEEE International Conference on Image Processing (ICIP'98), Vol.3, pp.227-231, Chicago, Illinois, USA, Oct. (1998).
 - 12) B.S. Morse and D. Schwartzwald: "Image Magnification Using Level-Set Reconstruction", In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01), Vol.1, pp.333-340, Kauai, HI, USA Dec. (2001).
 - 13) X. Yu, B.S. Morse, and T.W. Sederberg: "Image Reconstruction Using Data-Dependent Triangulation", IEEE Computer Graphics and Applications, Vol.21, No.3, pp.62-68 May/June. (2001).
 - 14) D. Su and P. Willis: "Image Interpolation by Pixel Level Data-Dependent Triangulation", Computer Graphics Forum, Vol.23, No.2, pp.189-201 Jun. (2004).
 - 15) J. Sun, N.N. Zheng, H. Tao, and H.Y. Shum: "Image Hallucination with Primal Sketch Priors", In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03), Vol.2, pp.729-736, Madison, WI, USA, Jun. (2003).
 - 16) 奥村晴彦, 山崎 敏: LHA と ZIP (株) ソフトバンクパブリッシング, 東京 (2003).
 - 17) G.N.N. Martin: "Range encoding: an algorithm for removing redundancy from a digitised message", Video & Data Recording Conference, Southampton, Jul. (1979).
 - 18) M. Burrows and D.J. Wheeler: "A Block-sorting Lossless Data Compression Algorithm", Digital SRC Research Report, No.124, Digital Systems Research Center, Palo Alto, CA, USA, May (1994).
 - 19) J.G. Cleary and I.H. Witten: "Data Compression Using Adaptive Coding and Partial String Matching", IEEE Transactions on Communications, Vol.32, No.4, pp.396-402, Apr. (1984).
 - 20) D. Shkarin: "PPM: one step to practicality", In Proceedings of the IEEE Data Compression Conference (DCC'02), pp.202-211, Snowbird, Utah, USA, Apr. (2002).
 - 21) M. Drinic, D. Kirovski, and M. Potkonjak: "PPM Model Cleaning", In Proceedings of the IEEE Data Compression Conference (DCC'03), pp.163-172, Snowbird, Utah, USA, Mar. (2003).
 - 22) M.J. Weinberger, G. Seroussi, and G. Sapiro: "LOCO-I: A Low Complexity, Context-Based, Lossless Image Compression Algorithm", In Proceedings of the IEEE Data Compression Conference (DCC'96), pp.140-149, Snowbird, Utah, USA, Mar. (1996).
 - 23) M.J. Weinberger and G. Seroussi: "The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS", IEEE Transactions on Image Processing, Vol.9, pp.1309-1324, Aug. (2000).
 - 24) M.W. Marcellin, M.J. Gormish, A. Bilgin, and M.P. Boliek: "An Overview of JPEG-2000", Data Compression Conference 2000, pp.523-544 (2000).
 - 25) A.N. Skodras, C.A. Christopoulos, and T. Ebrahimi: "JPEG2000: The Upcoming Still Image Compression Standard", In Proceedings of the 11th Portuguese Conference on Pattern Recognition, pp.359-366, Porto, Portugal, May (2000).
 - 26) DAM 推進協議会: ブロードバンド時代の新画像フォーマット VFZ. 日経 BP 出版センター, 東京 (2001).

(2005年4月7日受付)

北 郷 正 輝



2002年,東京工業大学工学部機械科学科卒. 2004年,東京工業大学大学院理工学研究科機械物理工学専攻修士課程了. 現在,同大学院博士課程在学中. 2005年8月より University of California, Irvine, Computer Graphics Lab にて Surface Reconstruction, Geometry Compression などの研究に従事. 工修. ACM 学生会員.

萩 原 一 郎 (正会員)



1970年,京都大学工学部数理工学科卒. 1972年,京都大学工学研究科数理工学専攻修士課程修了. 同年4月から1996年3月まで,日産自動車. 1996年4月以降,東京工業大学教授. 現在,東京工業大学教授大学院理工学研究科機械物理工学専攻教授. 応用数理学会理事・学会誌編集長. 応用数理にかかわる研究に興味. 米国機械学会, 日本機械学会, 自動車技術会各フェロー.