

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2008-42335  
(P2008-42335A)

(43) 公開日 平成20年2月21日(2008.2.21)

(51) Int.Cl.			F I			テーマコード (参考)		
HO4N	1/41	(2006.01)	HO4N	1/41	B	5B057		
GO6T	9/00	(2006.01)	GO6T	9/00		5C078		
HO4N	1/413	(2006.01)	HO4N	1/413	D			

審査請求 未請求 請求項の数 12 O L (全 28 頁)

(21) 出願番号 特願2006-211219 (P2006-211219)  
(22) 出願日 平成18年8月2日(2006.8.2)

特許法第30条第1項適用申請有り 2006年2月27日~3月1日 社団法人 情報処理学会発行の「情報処理学会研究報告 情処研報 Vol. 2006, No. 20」に発表

(71) 出願人 503360115  
独立行政法人科学技術振興機構  
埼玉県川口市本町4丁目1番8号  
(71) 出願人 503027931  
学校法人同志社  
京都府京都市上京区今出川通烏丸東入玄武町601  
(74) 代理人 100078282  
弁理士 山本 秀策  
(74) 代理人 100062409  
弁理士 安村 高明  
(74) 代理人 100113413  
弁理士 森下 夏樹

最終頁に続く

(54) 【発明の名称】 画像特徴抽出方法および画像圧縮方法

(57) 【要約】

【課題】同一の圧縮率で常にJPEGよりも美しい圧縮を実現することが可能な画像特徴抽出方法を提供すること。

【解決手段】本発明の画像特徴抽出方法は、与えられた画像に対して $k^2$ 分割化処理を少なくとも1回実行することにより、該与えられた画像を多分割画像に変換するステップを包含し、該 $k^2$ 分割化処理は、a) 画像行列 $X$ に基づいて、行列 $T$ を作成するステップと、b) 行列 $T$ の特異値を計算するステップと、c)  $\min_j | \lambda_j - \lambda_{j-1} | > \epsilon$  が成立するか否かを判定するステップと、ステップc)の判定結果が「No」である場合には、拡大行列 $T$ の特異値を計算する処理を行った後、ステップc)に戻るステップと、e) ステップc)の判定結果が「Yes」である場合には、 $T = U S V^T$ となる $U$ を求めるステップと、f) 行列 $T_1 = U^T T$ を求めるステップと、g) 行列 $T_1$ に基づいて、画像行列 $X_1$ を作成するステップとを包含する。

【選択図】図5

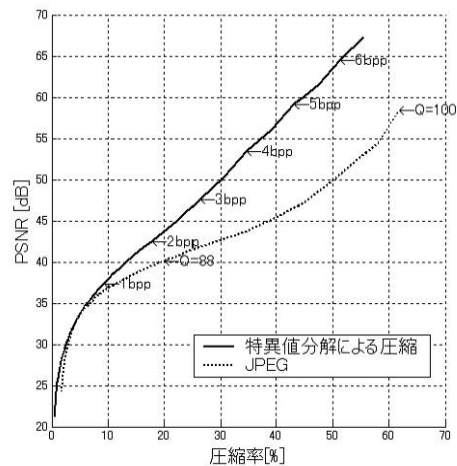


図5：原画像に対するファイルサイズ比と圧縮画像のPSNR値（原画像：Lena）

## 【特許請求の範囲】

## 【請求項 1】

画像の特徴を抽出する画像特徴抽出方法であって、

与えられた画像に対して  $k^2$  ( $k$  は 2 以上の任意の整数) 分割化処理を少なくとも 1 回実行することにより、該与えられた画像を多分割画像に変換するステップを包含し、

該  $k^2$  分割化処理は、

a) 画像行列  $X$  に基づいて、行列  $T$  を作成するステップと、

b) 行列  $T$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を計算するステップであって、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  である、ステップと、

c)  $\min_j | \lambda_j - \lambda_{j-1} | > \epsilon$  が成立するか否かを判定するステップであって、 $\epsilon$  はマシンプレシジョン以上の定数である、ステップと、

d) ステップ c) の判定結果が「No」である場合には、拡大行列  $T$  の特異値を計算する処理を行った後、ステップ c) に戻るステップと、

e) ステップ c) の判定結果が「Yes」である場合には、 $T = U S V^T$  となる  $U$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ ,  $U$  は直交行列、 $V$  は直交行列である、ステップと、

f) 行列  $T_1 = U^T T$  を求めるステップと、

g) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップと

を包含し、

ステップ d) における拡大行列  $T$  の特異値を計算する処理は、

画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズを有するふちと行列  $T$  とに基づいて、拡大行列  $T$  を作成するステップと、

拡大行列  $T$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を計算するステップであって、

$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  である、ステップと

を包含する、画像特徴抽出方法。

## 【請求項 2】

画像の特徴を抽出する画像特徴抽出方法であって、

与えられた画像に対して  $k^2$  ( $k$  は 2 以上の任意の整数) 分割化処理を少なくとも 1 回実行することにより、該与えられた画像を多分割画像に変換するステップを包含し、

該  $k^2$  分割化処理は、

a) 画像行列  $X$  に基づいて、行列  $T$  を作成するステップと、

b) 行列  $T$  の特異値分解  $T = U S V^T$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ ,  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  は、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップと、

c)  $\min_j | \lambda_j - \lambda_{j-1} | > \epsilon$  が成立するか否かを判定するステップであって、 $\epsilon$  はマシンプレシジョン以上の定数である、ステップと、

d) ステップ c) の判定結果が「No」である場合には、拡大行列  $T$  の特異値分解を実行する処理を行った後、ステップ c) に戻るステップと、

e) ステップ c) の判定結果が「Yes」である場合には、行列  $T_1 = U^T T$  を求めるステップと、

f) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップと

を包含し、

ステップ d) における拡大行列  $T$  の特異値分解を実行する処理は、

画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズを有するふちと行列  $T$  とに基づいて、拡大行列  $T$  を作成するステップと、

拡大行列  $T$  の特異値分解  $T = U S V^T$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ ,  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  は、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップと

を包含する、画像特徴抽出方法。

## 【請求項 3】

を包含する、画像特徴抽出方法。

前記画像は、グレースケール画像またはカラー画像である、請求項 1 または 2 に記載の画像特徴抽出方法。

【請求項 4】

前記 T および前記 T の特異値分解が浮動小数点演算で行われる、請求項 1 または 2 に記載の画像特徴抽出方法。

【請求項 5】

前記 T および前記 T の特異値分解が整数演算で行われる、請求項 1 または 2 に記載の画像特徴抽出方法。

【請求項 6】

前記  $k^2$  分割化処理とウェーブレット変換などの既知の  $k^2$  分割化処理とを併用することにより、前記与えられた画像行列 X が多分割画像に変換される、請求項 1 または 2 に記載の画像特徴抽出方法。

10

【請求項 7】

画像を圧縮する画像圧縮方法であって、

与えられた画像に対して  $k^2$  ( $k$  は 2 以上の任意の整数) 分割化処理を少なくとも 1 回実行することにより、該与えられた画像を多分割画像に変換するステップと、

該多分割画像に対してデータ圧縮処理を行うことにより、圧縮画像を作成するステップと

を包含し、

該  $k^2$  分割化処理は、

20

a) 画像行列 X に基づいて、行列 T を作成するステップと、

b) 行列 T の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を計算するステップであって、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  である、ステップと、

c)  $\min_j | \lambda_j - \lambda_{j-1} | > \epsilon$  が成立するか否かを判定するステップであって、 $\epsilon$  はマシン精度以上の定数である、ステップと、

d) ステップ c) の判定結果が「No」である場合には、拡大行列 T の特異値を計算する処理を行った後、ステップ c) に戻るステップと、

e) ステップ c) の判定結果が「Yes」である場合には、 $T = U S V^T$  となる U を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ , U は直交行列、V は直交行列である、ステップと、

30

f) 行列  $T_1 = U^T T$  を求めるステップと、

g) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップと

を包含し、

ステップ d) における拡大行列 T の特異値を計算する処理は、

画像行列 X の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズを有するふちと行列 T とに基づいて、拡大行列 T を作成するステップと、

拡大行列 T の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を計算するステップであって、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  である、ステップと

を包含する、画像圧縮方法。

【請求項 8】

40

画像を圧縮する画像圧縮方法であって、

与えられた画像に対して  $k^2$  ( $k$  は 2 以上の任意の整数) 分割化処理を少なくとも 1 回実行することにより、該与えられた画像を多分割画像に変換するステップと、

該多分割画像に対してデータ圧縮処理を行うことにより、圧縮画像を作成するステップと

を包含し、

該  $k^2$  分割化処理は、

a) 画像行列 X に基づいて、行列 T を作成するステップと、

b) 行列 T の特異値分解  $T = U S V^T$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ ,  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  は、 $\lambda_1 \geq \lambda_2 \geq \dots$

50

$k^2$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップと、  
 c)  $\min_j |j - j - 1| >$  が成立するか否かを判定するステップであって、  
 はマシン精度以上の定数である、ステップと、  
 d) ステップ c) の判定結果が「No」である場合には、拡大行列  $T$  の特異値分解を  
 実行する処理を行った後、ステップ c) に戻るステップと、  
 e) ステップ c) の判定結果が「Yes」である場合には、行列  $T_1 = U^T T$  を求める  
 ステップと、  
 f) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップと  
 を包含し、  
 ステップ d) における拡大行列  $T$  の特異値分解を実行する処理は、  
 画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズ  
 を有するふちと行列  $T$  とに基づいて、拡大行列  $T$  を作成するステップと、  
 拡大行列  $T$  の特異値分解  $T = U S V^T$  を求めるステップであって、 $S = \text{diag} ($   
 $1, 2, \dots, k^2)$ 、 $1, 2, \dots, k^2$  は、 $1, 2, \dots,$   
 $k^2$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップと  
 を包含する、画像圧縮方法。

10

## 【請求項 9】

前記画像は、グレースケール画像またはカラー画像である、請求項 7 または 8 に記載の  
 画像圧縮方法。

20

## 【請求項 10】

前記  $T$  および前記  $T$  の特異値分解が浮動小数点演算で行われる、請求項 7 または 8 に  
 記載の画像圧縮方法。

## 【請求項 11】

前記  $T$  および前記  $T$  の特異値分解が整数演算で行われる、請求項 7 または 8 に記載の  
 画像圧縮方法。

## 【請求項 12】

前記  $k^2$  分割化処理とウェーブレット変換などの既知の  $k^2$  分割化処理とを併用すること  
 により、前記与えられた画像行列  $X$  が多分割画像に変換される、請求項 7 または 8 に記  
 載の画像圧縮方法。

30

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本発明は、同一の圧縮率で常に J P E G よりも美しい圧縮を実現することが可能な画像  
 圧縮方法およびそれに適した画像特徴抽出方法に関する。

## 【背景技術】

## 【0002】

ハードウェアの進歩に伴い、コンピュータやデジタルカメラ・プリンタなどの周辺機器  
 で扱うことができる画像は、今や超精密アナログカメラの画質レベルに匹敵する。当然、  
 画像がもつデータ量も増大するため、細部にわたる忠実さよりも表示速度が重視される場  
 面では画像圧縮は必要不可欠な技術である。もちろん、画像圧縮によりデータ量を削減す  
 るだけでなく、原画像との違いは人間の視覚では、できる限り確認できないのが理想であ  
 る。同時に、記憶領域を小さくできれば、メモリやハードディスクなどの計算機資源が有  
 効に活用できる。

40

## 【0003】

Web サイトでは、一般的に G I F や J P E G、P N G といった形式で圧縮された画像  
 が利用されている。これらは、B M P 画像のような非圧縮画像に比べてファイルサイズが  
 格段に抑えられるため、高いアクセス性が求められる Web サイトには適している。また  
 、ネットワークを介して画像データを送受信する際も画像圧縮技術は重宝されることが多  
 い。

## 【0004】

50

画像は、コンピュータ内部でピクセルごとに色の濃淡が数値化される。例えば、縦  $m$  ピクセル、横  $n$  ピクセルの 256 階調グレースケール画像  $X$  は、成分に  $[0, 255]$  の整数値をもつ  $m \times n$  行列として表現される。

【0005】

【数1】

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1,n} \\ x_{21} & x_{22} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{pmatrix}.$$

ただし、 $x_{i,j}$  は  $[0, 255]$  の整数値である。

【0006】

以下では画像と行列を同一視して表す。RGB カラー画像は、赤 (Red) ・ 緑 (Green) ・ 青 (Blue) の3色について濃淡情報をそれぞれグレースケール画像と同様の行列形式で保持している。よって、簡単化のため、以下、グレースケール画像を対象とする。

【0007】

画像  $X$  に対して2次元離散ウェーブレット変換、または、ブロック化 + 特異値分解を1度実行すれば、左上  $(m/2) \times (n/2)$  ピクセルには画像  $X$  を  $1/2$  倍したような画像  $X_1$  が生成される。右上には縦方向、左下には横方向、右下には斜め方向のエッジを抽出したような画像が現れる。同様に、 $X_1$  の4分割、すなわち、 $X_1$  (縮小近似画像  $X_2$ ) + (縦方向のエッジ抽出画像) + (横方向のエッジ抽出画像) + (斜め方向のエッジ抽出画像)、 $X_2$  を4分割、 $\cdots$ 、 $X_k$  を4分割というように分割を繰り返せば多分割画像が作成できるが、これが画像圧縮における基本的な工程となる [1]。画像の多分割化後は、SPIHT [5] などでコーディングすればよい。なお、離散コサイン変換による画像圧縮でも  $X$  のブロック化は伴うが、ブロック特異値分解のような画像の4分割は行われない。量子化やハフマン符号によるコーディングが施され、データ量も小さく拡大縮小にも強い圧縮画像が得られるが、原画像を復元できない不可逆変換ということには注意が必要である [2]。

【0008】

ブロック特異値分解による多分割アルゴリズムが、Kakarala - Ogunbona [3] によって提案されている。このアルゴリズムと離散ウェーブレット変換を併用したハイブリット型アルゴリズムも報告されている [1]。指紋のような特別な性質をもつ画像では、JPEG や JPEG 2000 として実用化されている離散コサイン変換や離散ウェーブレット変換よりも、ハイブリット型アルゴリズムによって、より自然な画質で画像圧縮できることも知られている。ブロック特異値分解による画像圧縮には様々な可能性が秘められており、さらなる数値的な検証がなされるべきである。

【非特許文献1】 [1] Ashino, R., Morimoto, A., Nagase, M., and Vaillancourt, R.: Image compression with multiresolution singular value decomposition and other methods, Math. Comput. Model., Vol. 41, pp. 773-790 (2005)

【非特許文献2】 [2] 越智宏, 黒田秀夫: JPEG & MPEG 図解でわかる画像圧縮技術, 日本実業出版社 (2006)

【非特許文献3】 [3] Kakarala, R., and Ogunbona, O. P.: Signal analysis using a multiresolution form of the singular value decomposition, IEEE Trans. Image Process., Vol. 10, pp. 724-735 (2001)

【非特許文献4】 [4] Iwasaki, M., and Nakamura, Y.: Ac

10

20

30

40

50

curate computation of singular values in terms of shifted integrable schemes, (submitted)

【非特許文献5】[5] Said, A., and Pearlman, A.W.: A new fast and efficient image codec based on set partitioning in hierarchical trees, IEEE Trans. on Circuits and Systems for Video Technology, Vol. 6, pp. 243-250 (1996)

【非特許文献6】[6] 高田雅美, 木村欣司, 岩崎雅史, 中村佳正: 高速特異値分解のためのライブラリ開発, 投稿中

【非特許文献7】[7] Parlett, B.N., and Marques, O.A.: An implementation of the dqds algorithm (positive case), Lin. Alg. Appl., Vol. 309, pp. 217-259 (2000)

【発明の開示】

【発明が解決しようとする課題】

【0009】

本明細書では、まず、Kakarala-Ogunbonaアルゴリズムで多分割する際、特異値分解の対象となる行列の特異値の分布を調べる。画像によっては特異値が互いに重複および近接するが、そのような場合、既存の特異値分解法では、精度よく特異ベクトルが計算できるとは限らない[6]。[6]は倍精度演算による実験結果であるが、単精度や整数型演算では特異ベクトルを高い精度で求めることはより一層困難となる。

【0010】

本発明では、ブロック特異値分解において特異値の近接度を下げるアルゴリズムを新たに定式化する。本発明の目的の1つは、同一の圧縮率で常にJPEGよりも美しい圧縮を実現することが可能な画像圧縮方法およびそれに適した画像特徴抽出方法を提供することにある。

【課題を解決するための手段】

【0011】

本発明の方法は、画像の特徴を抽出する画像特徴抽出方法であって、与えられた画像に対して $k^2$  ( $k$ は2以上の任意の整数)分割化処理を少なくとも1回実行することにより、該与えられた画像を多分割画像に変換するステップを包含し、該 $k^2$ 分割化処理は、a) 画像行列 $X$ に基づいて、行列 $T$ を作成するステップと、b) 行列 $T$ の特異値 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ を計算するステップであって、 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ である、ステップと、c)  $\min_j |\sigma_j - \sigma_{j-1}| > \epsilon$  が成立するか否かを判定するステップであって、 $\epsilon$ はマシンプシロン以上の定数である、ステップと、d) ステップc)の判定結果が「No」である場合には、拡大行列 $T$ の特異値を計算する処理を行った後、ステップc)に戻るステップと、e) ステップc)の判定結果が「Yes」である場合には、 $T = USV^T$ となる $U$ を求めるステップであって、 $S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{k^2})$ ,  $U$ は直交行列、 $V$ は直交行列である、ステップと、f) 行列 $T_1 = U^T T$ を求めるステップと、g) 行列 $T_1$ に基づいて、画像行列 $X_1$ を作成するステップとを包含し、ステップd)における拡大行列 $T$ の特異値を計算する処理は、画像行列 $X$ の少なくとも一辺の少なくとも一部に付けられる少なくとも $k \times k$ のサイズを有するふちと行列 $T$ とに基づいて、拡大行列 $T$ を作成するステップと、拡大行列 $T$ の特異値 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ を計算するステップであって、 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ である、ステップとを包含し、これにより、上記目的が達成される。

【0012】

本発明の方法は、画像の特徴を抽出する画像特徴抽出方法であって、与えられた画像に対して $k^2$  ( $k$ は2以上の任意の整数)分割化処理を少なくとも1回実行することにより

、該与えられた画像を多分割画像に変換するステップを包含し、該  $k^2$  分割化処理は、a) 画像行列  $X$  に基づいて、行列  $T$  を作成するステップと、b) 行列  $T$  の特異値分解  $T = USV^T$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ 、 $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  は、 $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップと、c)  $\min_j |j - j - 1| >$  が成立するか否かを判定するステップであって、 $\lambda$  はマシンイpsilon以上の定数である、ステップと、d) ステップc)の判定結果が「No」である場合には、拡大行列  $T$  の特異値分解を実行する処理を行った後、ステップc)に戻るステップと、e) ステップc)の判定結果が「Yes」である場合には、行列  $T_1 = U^T T$  を求めるステップと、f) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップとを包含し、ステップd)における拡大行列  $T$  の特異値分解を実行する処理は、画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズを有するふちと行列  $T$  とに基づいて、拡大行列  $T$  を作成するステップと、拡大行列  $T$  の特異値分解  $T = USV^T$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ 、 $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  は、 $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップとを包含し、これにより、上記目的が達成される。

10

【0013】

前記画像は、グレースケール画像またはカラー画像であってもよい。

【0014】

前記  $T$  および前記  $T$  の特異値分解が浮動小数点演算で行われてもよい。

20

【0015】

前記  $T$  および前記  $T$  の特異値分解が整数演算で行われてもよい。

【0016】

前記  $k^2$  分割化処理とウェーブレット変換などの既知の  $k^2$  分割化処理とを併用することにより、前記与えられた画像行列  $X$  が多分割画像に変換されてもよい。

【0017】

本発明の方法は、画像を圧縮する画像圧縮方法であって、与えられた画像に対して  $k^2$  ( $k$  は2以上の任意の整数) 分割化処理を少なくとも1回実行することにより、該与えられた画像を多分割画像に変換するステップと、該多分割画像に対してデータ圧縮処理を行うことにより、圧縮画像を作成するステップとを包含し、該  $k^2$  分割化処理は、a) 画像行列  $X$  に基づいて、行列  $T$  を作成するステップと、b) 行列  $T$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を計算するステップであって、 $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  である、ステップと、c)  $\min_j |j - j - 1| >$  が成立するか否かを判定するステップであって、 $\lambda$  はマシンイpsilon以上の定数である、ステップと、d) ステップc)の判定結果が「No」である場合には、拡大行列  $T$  の特異値を計算する処理を行った後、ステップc)に戻るステップと、e) ステップc)の判定結果が「Yes」である場合には、 $T = USV^T$  となる  $U$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ 、 $U$  は直交行列、 $V$  は直交行列である、ステップと、f) 行列  $T_1 = U^T T$  を求めるステップと、g) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップとを包含し、ステップd)における拡大行列  $T$  の特異値を計算する処理は、画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズを有するふちと行列  $T$  とに基づいて、拡大行列  $T$  を作成するステップと、拡大行列  $T$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  を計算するステップであって、 $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  である、ステップとを包含し、これにより、上記目的が達成される。

30

40

【0018】

本発明の方法は、画像を圧縮する画像圧縮方法であって、与えられた画像に対して  $k^2$  ( $k$  は2以上の任意の整数) 分割化処理を少なくとも1回実行することにより、該与えられた画像を多分割画像に変換するステップと、該多分割画像に対してデータ圧縮処理を行うことにより、圧縮画像を作成するステップとを包含し、該  $k^2$  分割化処理は、a) 画像行列  $X$  に基づいて、行列  $T$  を作成するステップと、b) 行列  $T$  の特異値分解  $T = USV^T$

50

を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_k, \lambda_{k+1}, \lambda_{k+2}, \dots, \lambda_n)$  は、 $\lambda_1, \lambda_2, \dots, \lambda_k$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップと、c)  $\min_j |\lambda_j - \lambda_{j-1}| > \epsilon$  が成立するか否かを判定するステップであって、 $\epsilon$  はマシン精度以上の定数である、ステップと、d) ステップ c) の判定結果が「No」である場合には、拡大行列  $T$  の特異値分解を実行する処理を行った後、ステップ c) に戻るステップと、e) ステップ c) の判定結果が「Yes」である場合には、行列  $T_1 = U^T T$  を求めるステップと、f) 行列  $T_1$  に基づいて、画像行列  $X_1$  を作成するステップとを包含し、ステップ d) における拡大行列  $T$  の特異値分解を実行する処理は、画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられる少なくとも  $k \times k$  のサイズを有するふちと行列  $T$  とに基づいて、拡大行列  $T$  を作成するステップと、拡大行列  $T$  の特異値分解  $T = U S V^T$  を求めるステップであって、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_k, \lambda_{k+1}, \lambda_{k+2}, \dots, \lambda_n)$  は、 $\lambda_1, \lambda_2, \dots, \lambda_k$  を満たす  $T$  の特異値、 $U$  は直交行列、 $V$  は直交行列である、ステップとを包含し、これにより、上記目的が達成される。

【0019】

前記画像は、グレースケール画像またはカラー画像であってもよい。

【0020】

前記  $T$  および前記  $T$  の特異値分解が浮動小数点演算で行われてもよい。

【0021】

前記  $T$  および前記  $T$  の特異値分解が整数演算で行われてもよい。

【0022】

前記  $k^2$  分割化処理とウェーブレット変換などの既知の  $k^2$  分割化処理とを併用することにより、前記与えられた画像行列  $X$  が多分割画像に変換されてもよい。

【発明の効果】

【0023】

本発明によれば、同一の圧縮率で常に JPEG よりも美しい圧縮を実現することが可能な画像圧縮方法およびそれに適した画像特徴抽出方法を提供することができる。

【発明を実施するための最良の形態】

【0024】

以下、図面を参照しながら、本発明の実施の形態を説明する。

【0025】

## 1. はじめに

これまでに画像のデータ量を圧縮する様々な方法が提案されている。ここでは、Kakarala-Ogunbona の画像圧縮アルゴリズムを取り上げる。この方法と離散ウェーブレット変換とを併用することで、すぐれた画像圧縮が可能となる。この方法には行列の多分割 (multiresolution) 特異値分解の計算過程が含まれるが、その数値的検証はこれまで十分には行われてはいない。本明細書では、まず、画像のタイプによって特異値分布が大きく変動すること、ある場合には特異値がクラスタをなすことを数値的に示す。特異値の近接度が高いと既存の特異値分解ルーチンでは必ずしも特異ベクトルが高精度に求められるとは限らず、圧縮画像が原画像と大きく異なってしまう危険性がある。そこで、本発明において、特異値のクラスタを分散させ、特異値相互のギャップを拡大するアルゴリズムを提案し、数値実験によってその効果を明らかにする。

【0026】

2章では、画像を多分割化するための Kakarala-Ogunbona のブロック特異値分解アルゴリズムについて概観する。3章において、縦、横、斜め縞模様、および、ランダム模様の画像に対して Kakarala-Ogunbona アルゴリズムを適用したとき、どのような特異値分布をもつ行列が現れるかを論じる。4章では、ブロック特異値分解アルゴリズムにおいて、特異値相互のギャップを拡大し近接度を下げる効果的な方法について提案し、その効果を数値実験で確認する。



【 0 0 2 7 】

2 . K a k a r a l a - O g u n b o n a の ブ ロ ッ ク 特 異 値 分 解 アルゴリズム

画像行列  $X$  に対して、2次元ウェーブレット変換すれば図1のような4分割画像が得られる。縮小近似画像の4分割化を繰り返すと多分割画像が生成され、SPHITなどでコーディングすれば画像圧縮は完了する。なお、多分割画像は4分割画像に限定されない。縮小近似画像の  $k^2$  分割化を繰り返すことにより多分割画像を生成するようにしてもよい。ここで、 $k$  は2以上の任意の整数である。画像の多分割化は、 $X$  のブロック化 + 特異値分解でも実現できる。なお、離散コサイン変換も  $X$  のブロック化と併用されるが、この場合は多分割画像が生成されない。本章では、画像を多分割するための  $K a k a r a l a - O g u n b o n a$  によって提案されたブロック特異値分解アルゴリズムについて概説する。 $K a k a r a l a - O g u n b o n a$  アルゴリズムは任意のブロックサイズで任意サイズの長方形画像に適用できるが、その場合については [ 1 , 3 ] を参照されたい。また、大きな画像は適当なサイズの複数個の画像に分けて、それぞれについて個別に多分割化してもよい。便宜上、以下ではブロックサイズを2として32ピクセル×32ピクセルの原画像  $X$  から4分割画像  $X_1$  を作成する場合における  $K a k a r a l a - O g u n b o n a$  アルゴリズムを説明する。

10

【 0 0 2 8 】

K a k a r a l a - O g u n b o n a アルゴリズム

20

( 1 )  $32 \times 32$  行列  $X$  を  $2 \times 2$  小行列  $X(k, l)$  (  $1 \leq k \leq 16, 1 \leq l \leq 16$  ) に分割する。

【 0 0 2 9 】

【 数 2 】

$$X = \begin{pmatrix} X(1,1) & \cdots & X(1,16) \\ \vdots & \ddots & \vdots \\ X(16,1) & \cdots & X(16,16) \end{pmatrix}.$$

30

$$X(k, l) \equiv \begin{pmatrix} x_{2k-1, 2l-1} & x_{2k-1, 2l} \\ x_{2k, 2l-1} & x_{2k, 2l} \end{pmatrix}.$$

( 2 )  $X(k, l)$  ごとに  $4 \times 1$  の列ベクトルに変換する。

【 0 0 3 0 】

【 数 3 】

$$X(k, l) \mapsto x(k, l) \equiv \begin{pmatrix} x_{2k-1, 2l-1} \\ x_{2k, 2l-1} \\ x_{2k-1, 2l} \\ x_{2k, 2l} \end{pmatrix}.$$

40

( 3 ) 列ベクトルを並べて  $4 \times 256$  行列  $T$  を作成する。

【 0 0 3 1 】

【 数 4 】

$$T = ( T(1) \ T(2) \ \cdots \ T(16) ).$$

$$T(t) \equiv ( x(1, t) \ x(2, t) \ \cdots \ x(16, t) )$$

( 4 )  $T$  の特異値分解  $T = U S V^T$  を求める。ただし、 $S = \text{diag} ( \sigma_1, \sigma_2, \dots )$

50

3, 4),  $t_k$ ,  $k = 1, 2, 3, 4$  は  $T$  の特異値、 $U$  は  $4 \times 4$  直交行列、 $V$  は  $256 \times 256$  直交行列とする。

【0032】

(5)  $4 \times 256$  行列  $T_1 = U^T T$  を求める。

【0033】

(6)  $T_1$  の  $k$  行目ベクトルを  $16 \times 16$  行列  $X_1(k)$  に変換する。

【0034】

【数5】

$$X_1(k) = \begin{pmatrix} t_1(k,1) & \cdots & t_1(k,241) \\ \vdots & \ddots & \vdots \\ t_1(k,16) & \cdots & t_1(k,256) \end{pmatrix}.$$

10

$t_1(k, j)$ :  $T_1$  の  $(k, j)$  成分.

(7)  $X_1(k)$  を並べて  $32 \times 32$  行列  $X_1$  を作成する。

【0035】

【数6】

$$X_1 = \left( \begin{array}{c|c} X_1(1) & X_1(2) \\ \hline X_1(3) & X_1(4) \end{array} \right).$$

20

(4) で得られる直交行列  $U$  の第 1 列目ベクトルは、(1) でブロック化された  $2 \times 2$  行列  $X(k, 1)$  の 4 成分を平均化するフィルタとなる。(2)、(3) によって  $X(k, 1)$  の成分は  $T$  の列に配置されるので、(5) のように  $T$  の左から  $U^T$  を作用させれば、 $T_1$  の第 1 行目にはブロックに含まれる 4 成分の平均値が並ぶ。よって、(6) のように  $T_1$  の第 1 行目を  $X_1(1)$  に変換すれば、それが  $X$  の縮小近似画像となる。同様に、 $X_1(2)$ 、 $X_1(3)$ 、 $X_1(4)$  はそれぞれ  $X$  の縦、横、斜め方向のエッジが強調された画像と対応する。したがって、(7) のように  $X_1(k)$  を並べると、 $X$  の 4 分割化が完了する。

【0036】

30

### 3. 画像の種類と特異値分布

KakaraIa-Ogunbona アルゴリズムで多分割画像を作成する際、特異値分解の対象となる行列  $T$  は画像ごとに異なり、ブロック特異値分解で求められる特異値  $j$  ( $1, 2, \dots$ ) も画像によって変動する。

【0037】

特異値分解では、特異値のギャップ  $gap_j = |t_j - t_{j+1}|$  は特異ベクトルの精度に関する重要な指標である。すなわち、一般に特異値の分布が密になるほど、特異ベクトルの直交性が悪化する [6]。

【0038】

40

以下では、図 2 の縦、横、斜め縞およびランダム模様の  $32$  ピクセル  $\times$   $32$  ピクセル グレースケール画像を作成して、KakaraIa-Ogunbona アルゴリズムのもとで行列  $X_*$  がもつ特異値の最小ギャップ  $gap_{min} = \min_j |t_j - t_{j+1}|$  がどの程度になるかを調べた。数値実験には CPU: Pentium (登録商標) M1.2GHz、Memory: 504MB をもつ IBM Think Pad 上で Matlab 6.5 を使用した。なお、行列の特異値分解は Matlab の組み込み関数  $svd()$  によって倍精度で求めた。

【0039】

図 2 のテスト画像 (a) ~ 画像 (e) は、それぞれ、縞の濃淡と太さまたは点の濃淡をランダムに変化させて生成した画像である。具体的には、行列の成分  $x_k$ 、 $k = 1, 2,$

50

・・・, 1024を[0, 255]のランダムな整数とし、このような $X_a$ 、 $X_b$ 、 $X_c$ 、 $X_d$ 、 $X_e$ をそれぞれ10000個ずつ準備した。

【0040】

【数7】

$$X_a = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{132} \\ x_{21} & x_{22} & \dots & x_{232} \\ \vdots & \vdots & \ddots & \vdots \\ x_{321} & x_{322} & \dots & x_{3232} \end{pmatrix}, X_b = \begin{pmatrix} x_{11} & x_{11} & \dots & x_{11} \\ x_{22} & x_{22} & \dots & x_{22} \\ \vdots & \vdots & \ddots & \vdots \\ x_{32} & x_{32} & \dots & x_{32} \end{pmatrix}$$

$$X_c = \begin{pmatrix} x_{32} & \dots & x_{2} & x_{1} \\ \vdots & \ddots & \vdots & x_{2} \\ x_{62} & \dots & \dots & \vdots \\ x_{63} & x_{62} & \dots & x_{32} \end{pmatrix}, X_d = \begin{pmatrix} x_{11} & x_{2} & \dots & x_{32} \\ x_{2} & x_{3} & \dots & x_{33} \\ \vdots & \vdots & \ddots & \vdots \\ x_{32} & x_{33} & \dots & x_{63} \end{pmatrix}$$

$$X_e = \begin{pmatrix} x_{11} & x_{2} & \dots & x_{32} \\ x_{33} & x_{34} & \dots & x_{64} \\ \vdots & \vdots & \ddots & \vdots \\ x_{993} & x_{994} & \dots & x_{1024} \end{pmatrix}$$

画像(a)および画像(b)について、それぞれ、10000個の4分割化をしたところ、 $gap_{min}$ の値は同じような傾向を示していて、ほとんどの場合 $10^{-10}$ 以下となった。表1のように、最も特異値が近接する場合で $gap_{min}$ がマシンイプシロン程度の極めて小さい値になることも分かった。一方、画像(c)および画像(d)では、図3のように、ほとんどが $gap_{min} > 1$ であったが、表1に示すように、 $gap_{min}$ がゼロとなり行列 $X_*$ の特異値が重複することもあった。画像(e)では、概ね $gap_{min} > 1$ 、かつ、最も特異値が近接する場合でも $gap_{min} = 0.52$ と他の画像の場合と比較して特異値が分散していることが分かった。

【0041】

【表1】

表1: 5種類の画像10000個をそれぞれ4分割する際、特異値分解の対象となる行列Tがもつ特異値相互の最小ギャップ $gap_{min}$ (10000個の画像ごとに $gap_{min}$ を求めた際の最大値と最小値)

	$gap_{min}$	
	最大値	最小値
(a) 縦縞	2.0E-03	1.4E-16
(b) 横縞	2.2E-03	2.7E-15
(c) 斜め縞I	1.1E-03	0.0E-00
(d) 斜め縞II	9.9E-02	0.0E-00
(e) ランダム	1.5E-02	5.2E-01

倍精度計算において、 $gap_{min} > 10^{-3}$ 程度ならば直交性のよい特異ベクトルが計算できるとされている[6]。この中で、常にこの条件が満たされるのは画像(e)のみである。つまり、画像(a)~画像(d)のような何らかの規則性をもつ画像の場合は、Kakaraala-Ogunbonaアルゴリズムでは行列 $X_*$ を精度よく特異値分解できず、その結果、美しい圧縮画像が得られない可能性がある。

【0042】

#### 4. 拡大画像

原画像Xに何らかの規則性があると、Kakaraala-Ogunbonaアルゴリズム

10

20

30

40

50

ムにおける特異値分解が高精度に実現できるかは定かでない。つまり、原画像  $X$  より生成される行列  $T$  の特異値相互の最小ギャップ  $gap_{min}$  が小さければ、必ずしも美しい 4 分割画像が得られるとは限らない。本章では、原画像  $X$  のまわりにランダム模様のふちを追加することで規則性を崩した  $X$  バーについて検討する。ここで、「 $X$  バー」とは  $X$  の上に水平線を付した表記と同義である。なお、原画像サイズは 32 ピクセル  $\times$  32 ピクセル、Kakarala-Ogunbona アルゴリズムにおけるブロックサイズを  $b = 2$  とする。

【0043】

まず、左側のみに 2 ピクセルのふちが付いた拡大画像を表す  $32 \times 34$  行列

【0044】

【数 8】

$$\bar{X} = \left( \begin{array}{cc|c} d_1 & d_2 & X \\ d_3 & d_4 & \\ \vdots & \vdots & \\ d_{63} & d_{64} & \end{array} \right)$$

について数学的に考察する。ただし、 $d_k$ ,  $k = 1, 2, \dots, 64$  は  $[0, 255]$  のランダムな整数とする。Kakarala-Ogunbona アルゴリズムの (1) ~ (4) において、 $X$  が  $4 \times 256$  行列  $T$  に、 $X$  バーが  $4 \times 272$  行列  $T$  バーに変換されるならば、 $T$  バーは  $T$  とふちの画像に対応する  $4 \times 16$  行列  $D$  によって

【0045】

【数 9】

$$\bar{T} = (D \ T)$$

と表現できる。ただし、 $D$  の成分は  $d_k$ ,  $k = 1, 2, \dots, 64$  からなる。このとき、

【0046】

【数 10】

$$\bar{T}\bar{T}^T = DD^T + TT^T$$

なので、 $D$  を変化させると  $T$  バー ( $T$  バー)<sup>T</sup> は  $TT^T$  と異なる固有値分布をもつことが分かる。つまり、 $D$  をうまく制御すれば、 $T$  バーは近接した特異値をもたない行列にできる。ここで、「 $T$  バー」とは  $T$  の上に水平線を付した表記と同義である。

【0047】

左側に加えて右側、上側、下側に任意サイズのふちを付けた拡大画像  $X$  バーは、Kakarala-Ogunbona アルゴリズムの (1) ~ (4) によって

【0048】

【数 11】

$$\bar{T} = (D_1 \ T_1 \ D_2 \ \dots \ D_{32} \ T_{32} \ D_{33})$$

に変換される。ただし、 $D_k$ ,  $k = 1, 2, \dots, 32$  は上下左右いずれかのふちの画像に対応する行数 4 の行列またはベクトルであり、 $T = (T_1 \ T_2 \ \dots \ T_{32})$  となる。ふちが左側のみの場合と同じように

【0049】

【数 12】

$$\bar{T}\bar{T}^T = DD^T + TT^T$$

$$DD^T \equiv \sum_{k=1}^{32} D_k D_k^T$$

が成り立つ。したがって、ふちあり画像を 4 分割の対象とすれば、 $gap_{min}$  の値が小

10

20

30

40

50

さくなるような行列の特異値分解は避けられる。また、Tバーの行を入れかえても特異値は変化せず、列を入れかえても直交行列Uは変化しないことにも注意したい。

【0050】

通常用いられる特異値分解アルゴリズムでは、原点シフトによって特異値の相対ギャップが拡大されるため、得られる特異値の精度は  $gap_{min}$  の値にほとんど依存しない。dqdsアルゴリズム[7]やmdLvsアルゴリズム[4]のように、高い相対精度をもつ特異値計算法もある。

【0051】

画像圧縮に必要なのは特異値ではなく特異ベクトルのいくつかの成分である。このことから、特異値の相対ギャップを拡大して特異ベクトルを高精度に計算する新しいアルゴリズムが定式化できる。

10

【0052】

なお、拡大画像は、原画像の左側のみにはふちを付けることによって得られるものには限定されない。拡大画像は、原画像のうち少なくとも1つにふちを付けることによって得られるものであってもよい。また、ふちの幅または長さは、2ピクセルには限定されない。ふちの幅および長さは、所定の数のピクセルであり得る。ここで、所定の数は任意の整数であり得る。ふちの幅および長さのピクセル数は、例えば、原画像の性質に応じて設計され得る。また、ふちに対応する各要素  $d_k$  の値は、ランダム値には限定されない。ふちに対応する各要素  $d_k$  の値は、実質的にランダムな値（例えば、擬似ランダム値）であれば同等の効果が得られる。

20

【0053】

以下では、画像Xの高精度な4分割アルゴリズムを提案する。

【0054】

#### 高精度な4分割アルゴリズム

Step 1: Kakarala - Ogunbonaアルゴリズムの(1)~(4)によってXからTに変換する。また、Tバー = Tとする。

【0055】

Step 2: Tバーの特異値を求める。

30

【0056】

$gap_{min}$  ならば  
Tバーの特異値分解  $Tバー = U S V^T$  を求める。

【0057】

$gap_{min} <$  ならば  
Tバー (  $D(Tバー)$  ) としてStep 2に戻る。ただし、Dは、成分が [ 0 , 255 ] のランダムな整数で与えられた  $4 \times n'$  行列、 $n'$  はふちの大きさに対応する。

【0058】

どの特異値分解アルゴリズムを利用するかによって  $\epsilon$  の値を設定すべきであるが、マシンイプシロン付近の小さな値は望ましくない。

40

【0059】

Step 3: ふちの情報を含まないTに左から  $U^T$  を作用させる、つまり、 $T_1 = U^T T$  とする。

【0060】

Step 4: Kakarala - Ogunbonaアルゴリズムの(6)、(7)によって4分割画像  $X_1$  を得る。

【0061】

このアルゴリズムは、特異値に近接しているものがあれば一時的にふちを付けて、特異値分解終了後には直ちにふちを取り除いて4分割画像を作成するという仕組みである。

50

## 【 0 0 6 2 】

この4分割アルゴリズムの効果（特に、Step 2の効果）を実証するための数値実験を行った。原画像は3章と同じく図2の画像（a）～画像（e）、拡大画像はそれぞれに2ピクセルのランダムなふちを上側と左側につけた画像とした。

## 【 0 0 6 3 】

## 【表2】

表2：5種類のふちあり画像10000個をそれぞれ4分割する際、特異値分解の対象となる行列Tがもつ特異値相互の最小ギャップ  $gap_{min}$  の最小値

	$gap_{min}$ の最小値
(a) 縦縞	7.8E - 04
(b) 横縞	1.5E - 03
(c) 斜め縞I	5.8E - 03
(d) 斜め縞II	4.1E - 03
(e) ランダム	3.2E - 01

10

画像（a）および画像（b）の拡大画像それぞれ10000個から生成されるTバーでは、特異値相互のギャップが  $gap_{min} > 10$  となった。行列Tバーの特異値が最も近接する場合でも、 $gap_{min}$  の値は、表2のように、画像（a）で  $7.8 \times 10^{-4}$ 、画像（b）で  $1.5 \times 10^{-3}$  となり、3章に示したふちなしよりも特異値のクラスタが分散されているのが分かる。画像（c）および画像（d）の拡大画像については、多くの場合でTバーの  $gap_{min}$  が0.1以上となり、ふちなし画像と比較すれば行列Tバーの特異値が若干近接するが、高精度に特異値分解するには全く支障のないレベルである。最低でも  $gap_{min}$  は  $O(10^{-3})$  に抑えられ、Tバーが重複特異値をもつ状況は回避できた。画像（e）では、原画像に規則性がないため、ふちを付けても  $gap_{min}$  の値が大きく変化することはなかった。

20

## 【 0 0 6 4 】

原画像をもとに縮小近似画像と縦、横、斜め方向のエッジ抽出画像の4分割画像を作成する画像圧縮方式において、有用な手段の1つであるKakarala-Ogunbonaのブロック特異値分解について説明した。5種類のテスト画像をKakarala-Ogunbonaアルゴリズムによって4分割化したところ、規則性のある画像についてはアルゴリズムの途中で近接または重複特異値をもつ行列が現れることがあった。特異値相互のギャップが小さくなれば、特異ベクトルの精度が低下するとの報告もあるため、本発明では特異値の相対ギャップを拡大し、クラスタを分散させる4分割アルゴリズムを定式化した。原画像にランダム模様のふちを追加して4分割化するというシンプルなアイデアではあるが、理論的にも実験的にも高精度な特異値分解が困難な状況を回避できることを示した。

30

## 【 0 0 6 5 】

## 5. 画像圧縮

提案した4分割化アルゴリズムとSPHITコーディングを組み合わせる圧縮画像を生成したところ、図5のような結果が得られた。図5の横軸は、原画像に対するファイルサイズ比（左ほど高圧縮）を示し、図5の縦軸は、PSNR値（上ほど美しい圧縮）を示す。図5において、実線は、本発明による画像圧縮の特性を示し、点線は、比較例として公知のJPEGによる画像圧縮の特性を示す。図5から、本発明の画像圧縮によれば、同じ圧縮率で常にJPEGよりも美しい圧縮となることが分かる。

40

## 【 0 0 6 6 】

また、行列Tの特異値がばらつくなれば、特異値分解を倍精度、単精度、整数演算のいずれで求めてもPSNR値はほとんど変わらない（表3参照）。

50

【 0 0 6 7 】

【 表 3 】

表 3 : 多分割画像を倍精度, 単精度, 整数 16bit 演算で求め, SPHIT コーディングしたときに得られる圧縮画像の PSNR 値 (原画像: Lena)

bpp	倍精度	単精度	整数 16bit
0.1	26.76	26.76	26.76
0.5	33.47	33.47	33.47
1.0	37.42	37.42	37.41
1.5	40.2	40.2	40.2

10

もちろん整数演算するのが最も高速で、かつ情報量を削減（より圧縮できる）できるが、近接あるいは重複特異値が存在する場合は倍精度演算のときよりも PSNR 値の悪化する可能性が高い。そのため、特に整数演算で画像圧縮を行う際には、提案アルゴリズムを多分割化に利用すべきであると考えらる。

【 0 0 6 8 】

SPHIT のほかに EZW による符号化もあり、それらをランレングス符号化やシャノン符号（ハフマン符号、算術符号など）により符号化することでデータはさらに圧縮される。また、LZ77 を基本アルゴリズムとする LHA、gzip、bzipp2、ZIP、CAB、7z、RAR、CAB、DGCA、GCA などを併用しても圧縮される。

20

【 0 0 6 9 】

## 6 . 画像圧縮方法の詳細

以下、図 6 ~ 図 16 を参照して、 $l \times m$  画像行列  $X$  で表される画像を圧縮する方法を詳細に説明する。なお、 $l \times m$  画像行列  $X$  で表される画像は、グレースケール画像でもよいし、カラー画像でもよい。

【 0 0 7 0 】

図 6 は、本発明の画像圧縮方法の処理の手順を示すフローチャートである。この処理は、任意のプロセッサによって実行され得る。

30

【 0 0 7 1 】

ステップ S 6 1 : 原画像  $X$  は、多分割画像  $X_M$  に変換される（多分割化： $X \rightarrow X_M$ ）。ここで、原画像  $X$  は、 $l \times m$  行列の形式で表される（以下、 $l \times m$  画像行列  $X$  という）。多分割画像  $X_M$  は、 $k^2$  分割を少なくとも 1 回実行することによって得られる画像であり、行列の形式で表される（以下、多分割画像行列  $X_M$  という）。ここで、 $k$  は 2 以上の任意の整数であり、 $l$ 、 $m$  は  $k$  の任意の倍数である。多分割画像  $X_M$  を得ることは、原画像  $X$  の特徴抽出を行うことにほかならない。ステップ S 6 1 を包含する画像特徴抽出方法もまた本発明の範囲内である。

【 0 0 7 2 】

ステップ S 6 2 : 多分割画像行列  $X_M$  に対してデータ圧縮処理を行うことにより、圧縮画像行列  $X_{c.o.m}$  が得られる（データ圧縮： $X_M \rightarrow X_{c.o.m}$ ）。

40

【 0 0 7 3 】

ここで、ステップ S 6 1 への入力である画像行列  $X$ 、ステップ S 6 1 からの出力であり、かつ、ステップ S 6 2 への入力である多分割画像行列  $X_M$ 、ステップ S 6 2 からの出力である圧縮画像行列  $X_{c.o.m}$  は、いずれも、任意の記憶装置（媒体）に格納され得る。

【 0 0 7 4 】

図 7 は、多分割化処理（図 6 のステップ S 6 1）の手順を示すフローチャートである。

【 0 0 7 5 】

ステップ S 7 2 において、画像行列  $X_n$  を  $k^2$  分割することにより画像行列  $X_{n+1}$  を生成する処理が少なくとも 1 回実行される。ここで、 $n = 0, 1, \dots, n_{max} - 1$

50

であり、 $X_0 = X$ である。

【0076】

ここで、分割回数  $n$ 、画像行列  $X_n$  は、いずれも、任意の記憶装置（媒体）に格納され得る。

【0077】

図8は、画像行列  $X_n$  を  $k^2$  分割することにより画像行列  $X_{n+1}$  を生成する処理（図7のステップS72）の手順を示すフローチャートである。ここでは、 $n = 0$  の場合の処理を説明するが、 $n = 1, \dots, n_{max} - 1$  の場合も同様である。

【0078】

ステップS81：  $l \times m$  画像行列  $X$  に基づいて、 $k^2 \times (lm/k^2)$  行列  $T$  が作成される。このステップは、上述したKakaraala-Ogunbonaアルゴリズムの(1)~(4)に対応する。

10

【0079】

ステップS82：  $T$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  が計算される。ただし、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  である。

【0080】

ステップS83：  $\min_j |\lambda_j - \lambda_{j-1}| > \epsilon$  が成立するか否かが判定される。ここで、 $\epsilon$  はマシンプレシジョン以上の定数を示す。ステップS83における判定結果が「Yes」である場合には、処理はステップS84に進む。ステップS83における判定結果が「No」である場合には、処理はステップS85に進む（ステップS85、S86において、拡大行列  $T$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  が計算される）。

20

【0081】

ステップS84：  $T = USV^T$  となる  $U$  のみが求められる。ただし、 $S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k^2})$ 、 $U$  は直交行列、 $V$  は直交行列とする。

【0082】

ステップS85： 行列  $T$  が拡大行列  $T'$  に変換される。拡大行列  $T'$  は、画像行列  $X$  の少なくとも一辺の少なくとも一部に付けられるふちと行列  $T$  とに基づいて生成される。ふちは、少なくとも  $k$  ピクセル  $\times k$  ピクセルのサイズを有する。ふちに属するピクセルの値はランダム値（もしくは、実質的にランダムな値）である。

【0083】

ステップS86：  $T'$  の特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$  が計算される。ただし、 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{k^2}$  である。この  $T'$  の特異値がステップS83に入力される（ステップS83において、 $\min_j |\lambda_j - \lambda_{j-1}| > \epsilon$  が成立するか否かが再び判定される）。

30

【0084】

その結果、 $\min_j |\lambda_j - \lambda_{j-1}| > \epsilon$  が成立するまで、ステップS85、S86が繰り返される。

【0085】

ステップS87： 行列  $T_1 = U^T T$  が求められる。行列  $T_1$  は、特徴抽出行列と呼ばれる。

40

【0086】

ステップS88： 行列  $T_1$  に基づいて、画像行列  $X_1$  が作成される。画像行列  $X_1$  は、行列  $X$  を  $k^2$  分割することによって得られる行列である。このステップは、上述したKakaraala-Ogunbonaアルゴリズムの(6)~(7)に対応する。

【0087】

ここで、画像行列  $X$ 、行列  $T_1$ 、行列  $S$ 、特異値  $\lambda_1, \lambda_2, \dots, \lambda_{k^2}$ 、拡大行列  $T'$ 、直交行列  $U$ 、 $k^2$  分割画像行列  $X_1$ 、特徴抽出行列  $T_1$  は、いずれも、任意の記憶装置（媒体）に格納され得る。

【0088】

図9は、画像行列  $X_n$  を  $k^2$  分割することにより画像行列  $X_{n+1}$  を生成する処理（図

50



7のステップS72)の別の手順を示すフローチャートである。ここでは、 $n = 0$ の場合の処理を説明するが、 $n = 1, \dots, n_{max} - 1$ の場合も同様である。

【0089】

ステップS91:  $l \times m$ 画像行列Xに基づいて、 $k^2 \times (lm/k^2)$ 行列Tが作成される。このステップは、上述したKakaraala-Ogunbonaアルゴリズムの(1)~(4)に対応する。

【0090】

ステップS92: Tの特異値分解 $T = USV^T$ が求められる。ただし、 $S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{k^2})$ 、 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ は、 $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{k^2}$ を満たすTの特異値、Uは直交行列、Vは直交行列とする。

10

【0091】

ステップS93:  $\min_j | \sigma_j - \sigma_{j-1} | > \epsilon$ が成立するか否かが判定される。ここで、 $\epsilon$ はマシンプシロン以上の定数を示す。ステップS93における判定結果が「Yes」である場合には、処理はステップS94に進む。ステップS93における判定結果が「No」である場合には、処理はステップS95に進む(ステップS95、S96において、拡大行列Tの特異値分解が実行される)。

【0092】

ステップS94: 行列 $T_1 = U^T T$ が求められる。行列 $T_1$ は、特徴抽出行列と呼ばれる。

【0093】

ステップS95: 行列Tが拡大行列 $T'$ に変換される。拡大行列 $T'$ は、画像行列Xの少なくとも一辺の少なくとも一部に付けられるふちと行列Tとに基づいて生成される。ふちは、ふちは、少なくとも $k$ ピクセル $\times k$ ピクセルのサイズを有する。ふちに属するピクセルの値はランダム値である。

20

【0094】

ステップS96:  $T'$ の特異値分解 $T' = USV^T$ が求められる。ただし、 $S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{k^2})$ 、 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ は、 $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{k^2}$ を満たす $T'$ の特異値、Uは直交行列、Vは直交行列とする。このTの特異値がステップS93に入力される(ステップS93において、 $\min_j | \sigma_j - \sigma_{j-1} | > \epsilon$ が成立するか否かが再び判定される)。

30

【0095】

その結果、 $\min_j | \sigma_j - \sigma_{j-1} | > \epsilon$ が成立するまで、ステップS95、S96が繰り返される。

【0096】

ステップS97: 行列 $T_1$ に基づいて、画像行列 $X_1$ が作成される。画像行列 $X_1$ は、行列Xを $k^2$ 分割することによって得られる行列である。このステップは、上述したKakaraala-Ogunbonaアルゴリズムの(6)~(7)に対応する。

【0097】

ここで、画像行列X、行列 $T_1$ 、行列S、特異値 $\sigma_1, \sigma_2, \dots, \sigma_{k^2}$ 、拡大行列 $T'$ 、直交行列U、 $k^2$ 分割画像行列 $X_1$ 、特徴抽出行列 $T_1$ は、いずれも、任意の記憶装置(媒体)に格納され得る。

40

【0098】

なお、Tおよび $T'$ の特異値分解は、浮動小数点演算で行われてもよい。あるいは、Tおよび前記 $T'$ の特異値分解は、整数演算で行われてもよい。

【0099】

図10は、 $l \times m$ 画像行列Xに基づいて $k^2 \times (lm/k^2)$ 行列Tを作成する処理(図8のステップS81および図9のステップS91)を模式的に示す。

【0100】

図11は、行列Tを拡大行列 $T'$ に変換する処理(図8のステップS85および図9のステップ95)を模式的に示す。

50

## 【0101】

図12は、行列 $T_1$ に基づいて画像行列 $X_1$ を作成する処理(図8のステップS88および図9のステップS97)を模式的に示す。

## 【0102】

なお、図8のステップS85および図9のステップS95において、画像行列 $X$ のふちは、画像行列 $X$ のすべての辺に付けられる必要はなく、画像行列 $X$ の少なくとも一辺に付けられてもよい。さらに、画像行列 $X$ のふちは、画像行列 $X$ の少なくとも一辺の全部に付けられる必要はなく、画像行列 $X$ の少なくとも一辺の少なくとも一部に付けられてもよい。

## 【0103】

図13は、画像行列 $X$ が $l \times m$ 行列である場合において、画像行列 $X$ に付けられるふちのパターンのバリエーションを示す。図13において黒塗り部分がふちに相当する。 $k^2$ 分割化が行われる場合には、少なくとも $k$ ピクセル $\times k$ ピクセルのサイズを有するふちが画像行列 $X$ に付けられればよい。

## 【0104】

なお、画像行列 $X$ に付けるふちに属するピクセルの値をどのような値にするかは重要であるが、画像行列 $X$ のどの位置にふちを付けるかは重要ではない。図14の矢印に示されるようにふちの位置を移動させたとしても同じ結果になるからである。ふちに属するピクセルの値は、例えば、ランダム値である。あるいは、ふちに属するピクセルの値は、実質的にランダムな値(例えば、擬似ランダム値)であってもよい。

## 【0105】

なお、図6に示されるステップS61において行われる多分割化処理は、図8に示される $k^2$ 分割化処理とウェーブレット変換などの既知の $k^2$ 分割化処理とを併用することによって行われてもよいし、図9に示される $k^2$ 分割化処理とウェーブレット変換などの既知の $k^2$ 分割化処理とを併用することによって行われてもよい。

## 【0106】

なお、図6に示されるステップS62において行われるデータ圧縮処理は、分割された領域の画像の類似性を利用して行われることが好ましい。上述したように、 $k^2$ 分割化を $n$ 回繰り返すと、大きさは異なるが類似した特徴が抽出される $n$ 箇所の領域が $(k^2 - 1)$ 組現れる。例えば、図15は、4分割化を3回繰り返した場合において、類似した特徴が抽出される領域を同一の丸付き数字で表したものである。図15の例では、3つの領域(1)の画像が類似した画像となり、3つの領域(2)の画像が類似した画像となり、3つの領域(3)の画像が類似した画像となる。このような領域間の画像の類似性を利用することにより、多分割画像行列 $X_M$ を単純に保存する場合に比べて必要とされるデータ量を削減することができる。このような領域間の画像の類似性を利用する方法としては、例えば、画素値の差分値を保存するようにしてもよいし、本来16桁の倍精度数で保存しなければならないが、類似性が高ければ単純に0を保存するようにしてもよい。このようなデータ圧縮処理は、例えば、SPHITやEZWによって実現することができる。

## 【0107】

さらに、ランレングス符号化やシャノン符号(ハフマン符号, 算術符号など)により符号化することでデータを圧縮するようにしてもよい。また、LZ77を基本アルゴリズムとするLHA、gzip、bz2、ZIP、CAB、7z、RAR、CAB、DGA、GCAなどを併用してデータを圧縮するようにしてもよい。

## 【0108】

さらに、上述した実施形態では、 $l \times m$ 画像行列 $X$ ( $k$ は2以上の任意の整数、 $l$ 、 $m$ は $k$ の任意の倍数)を用いて、 $k$ の倍数である縦横サイズを有する画像の特徴を抽出する例を説明したが、本発明はこれに限定されない。 $k$ の倍数でない縦横サイズを有する画像の特徴を抽出する場合には、例えば、画像行列の下側または左側に適切な数の行または列を追加することにより、 $k$ の倍数である縦横サイズを有する画像を作成し、その画像の特徴を抽出するにすればよい。追加する行または列に属するピクセルの値は、例えば、

10

20

30

40

50

0である。このようにして特徴抽出された画像から追加した行または列に対応する部分を取り除くことにより、所望の結果を得ることができる。あるいは、適切な数の行または列を追加する代わりに、適切な数の行または列を削除することによって、 $k$ の倍数である縦横サイズを有する画像を作成し、その画像の特徴を抽出するにすればよい。

【0109】

さらに、上述した実施形態では、画像行列 $X$ を $k^2$ 分割化する例を説明したが、画像行列 $X$ をいくつかの領域に分割して、各領域について独立に $k^2$ 分割化を行うようにしてもよい。この場合、各領域における分割数は同一である必要はなく異なってもよい。例えば、図16に示されるように、画像行列 $X$ を8つの領域(領域(1)~領域(8))に分割して、領域(1)について $2^2$ 分割化、領域(2)について $3^2$ 分割化・・・を行っ

10

【0110】

以上のように、本発明の好ましい実施形態を用いて本発明を例示してきたが、本発明は、この実施形態に限定して解釈されるべきものではない。本発明は、特許請求の範囲によってのみその範囲が解釈されるべきであることが理解される。当業者は、本発明の具体的な好ましい実施形態の記載から、本発明の記載および技術常識に基づいて等価な範囲を実施することができるということが理解される。

【産業上の利用可能性】

【0111】

本発明は、同一の圧縮率で常にJPGよりも美しい圧縮を実現することが可能な画像圧縮方法およびそれに適した画像特徴抽出方法等を提供するものとして有用である。

20

【図面の簡単な説明】

【0112】

【図1】(a)は原画像の一例を示す図、(b)は4分割画像の一例を示す図

【図2】5種類のテスト画像(32ピクセル×32ピクセル)を示す図であって、(a)は縦縞、(b)は横縞、(c)は斜め縞I、(d)は斜め縞II、(e)はランダムを示す。

【図3】5種類の画像10000個をそれぞれ4分割する際、特異値分解の対象となる行列 $T$ がもつ特異値相互の最小ギャップ $gap_{min}$ を示す図

【図4】4種類のふちあり画像10000個をそれぞれ分割する際、特異値分解の対象となる行列 $T$ がもつ特異値相互の最小ギャップ $gap_{min}$ を示す図

30

【図5】原画像に対するファイルサイズ比と圧縮画像のPSNR値との関係を示す図

【図6】本発明の画像圧縮方法の処理の手順を示すフローチャート

【図7】多分割化処理(図6のステップS61)の手順を示すフローチャートである。

【図8】画像行列 $X_n$ を $k^2$ 分割することにより画像行列 $X_{n+1}$ を生成する処理(図7のステップS72)の手順を示すフローチャート

【図9】画像行列 $X_n$ を $k^2$ 分割することにより画像行列 $X_{n+1}$ を生成する処理(図7のステップS72)の別の手順を示すフローチャート

【図10】 $1 \times m$ 画像行列 $X$ に基づいて $k^2 \times (1m/k^2)$ 行列 $T$ を作成する処理(図8のステップS81および図9のステップS91)を模式的に示す図

40

【図11】行列 $T$ を拡大行列 $T'$ に変換する処理(図8のステップS85および図9のステップ95)を模式的に示す。

【図12】行列 $T_1$ に基づいて画像行列 $X_1$ を作成する処理(図8のステップS88および図9のステップ97)を模式的に示す図

【図13】 $1 \times m$ 画像行列に付けられるふちのパターンのバリエーションを示す図

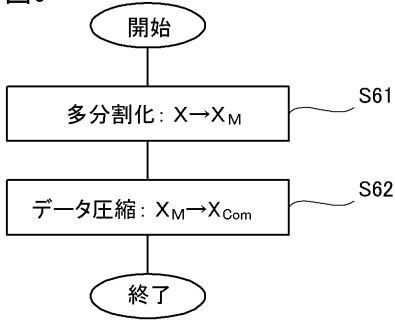
【図14】画像行列に付けられるふちの位置を移動した場合を説明するための図

【図15】類似した特徴が抽出される分割領域の一例を示す図

【図16】画像行列を複数の領域に分割して、各領域について独立に $k^2$ 分割化を行う場合を説明するための図

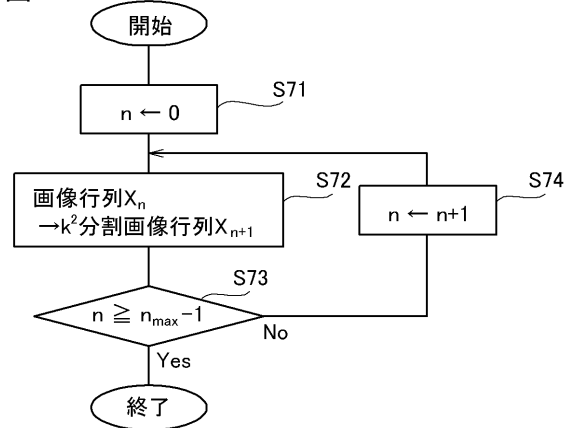
【図6】

図6



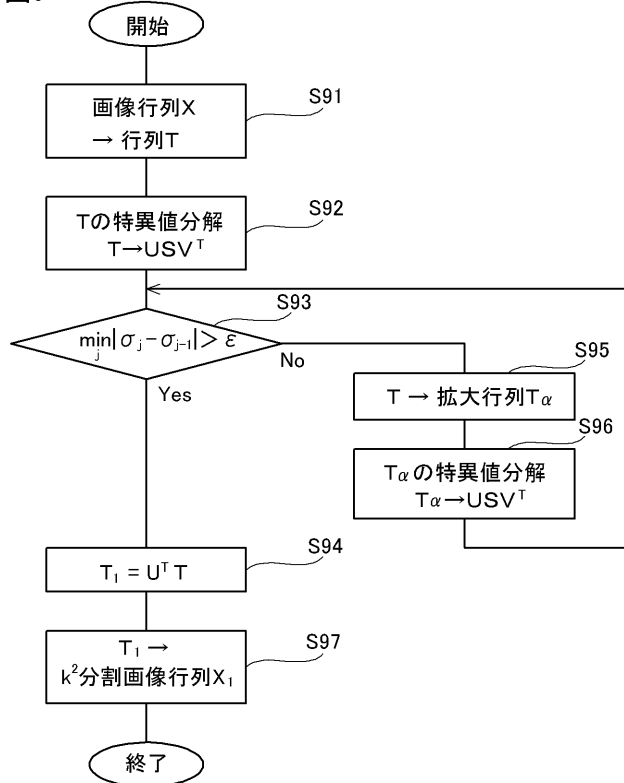
【図7】

図7



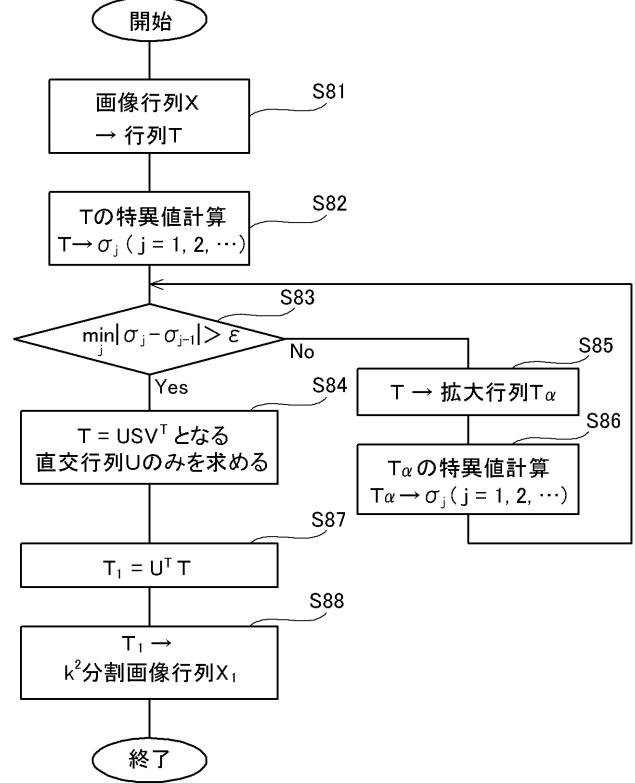
【図9】

図9



【図8】

図8



【 図 1 】



(a) 原画像

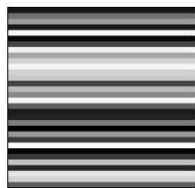


(b) 4分割画像

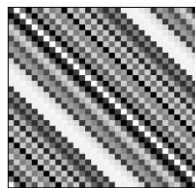
【 図 2 】



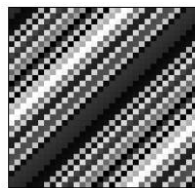
(a) 縦縞



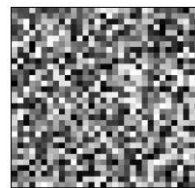
(b) 横縞



(c) 斜め縞 I



(d) 斜め縞 II



(e) ランダム

【 図 3 】

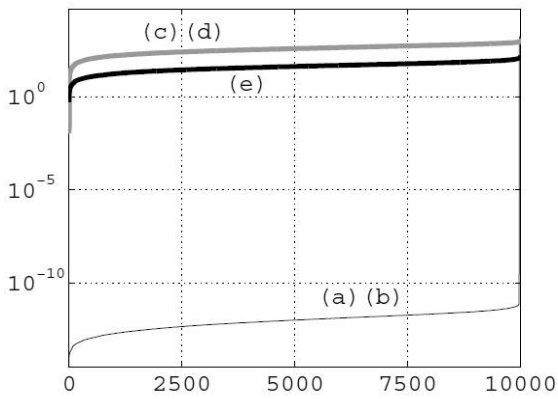


図 3 : 5 種類の画像 10000 個をそれぞれ 4 分割する際, 特異値分解の対象となる行列  $T$  がもつ特異値相互の最小ギャップ  $gap_{min}$  (横軸: 近接順位, 縦軸:  $gap_{min}$ )

【 図 4 】

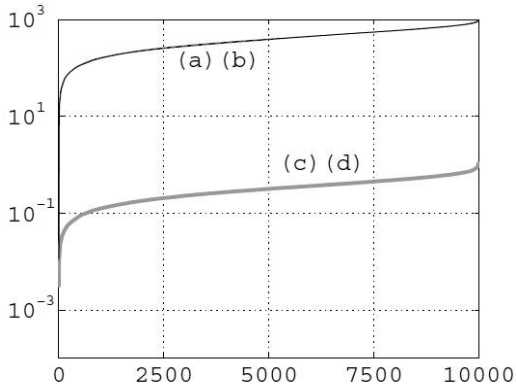


図 4 : 4 種類のふちあり画像 10000 個をそれぞれ 4 分割する際、特異値分解の対象となる行列  $T$  がもつ特異値相互の最小ギャップ  $gap_{min}$  (横軸：近接順位, 縦軸： $gap_{min}$ )

【 図 5 】

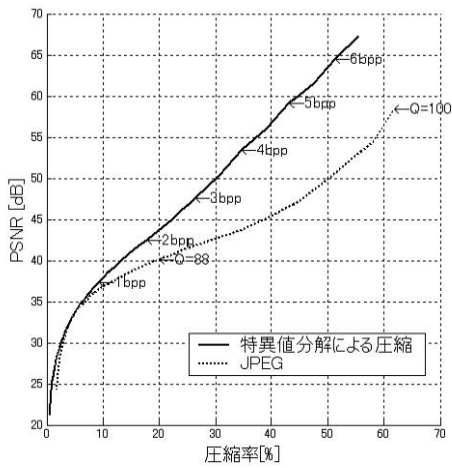


図 5 : 原画像に対するファイルサイズ比と圧縮画像の PSNR 値 (原画像：Lena)

【 図 1 0 】

$$X = \begin{bmatrix} \begin{array}{|c|c|} \hline X(1) & X(k_1+1) \\ \hline \end{array} & \dots & \begin{array}{|c|} \hline X(k_2k_1-k_1+1) \\ \hline \end{array} \\ \begin{array}{|c|c|} \hline X(2) & X(k_1+2) \\ \hline \end{array} & \dots & \begin{array}{|c|} \hline X(k_2k_1-k_1+2) \\ \hline \end{array} \\ \vdots & \vdots & \vdots \\ \begin{array}{|c|c|} \hline X(k_1) & X(2k_1) \\ \hline \end{array} & \dots & \begin{array}{|c|} \hline X(k_2k_1) \\ \hline \end{array} \end{bmatrix}$$

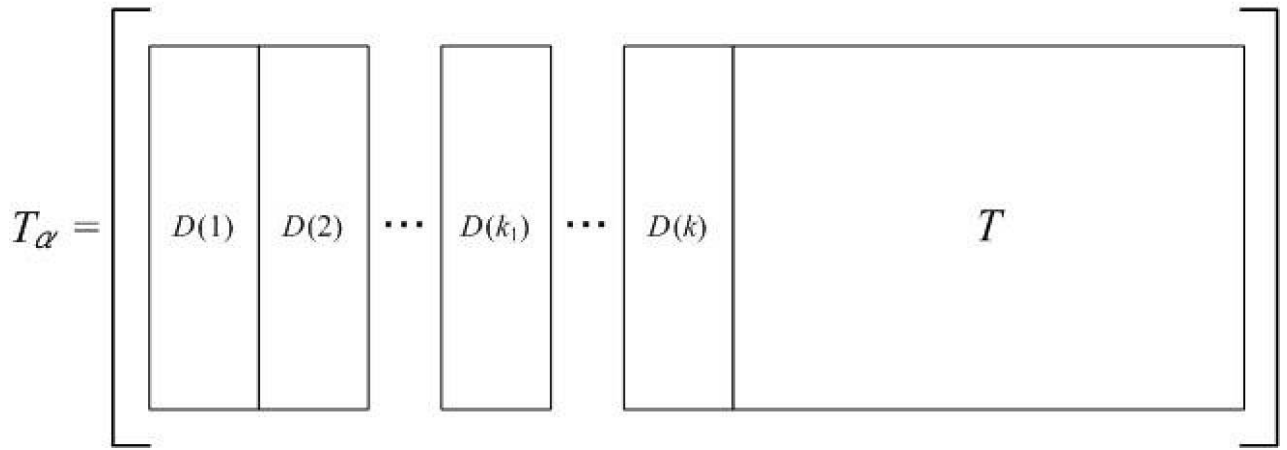
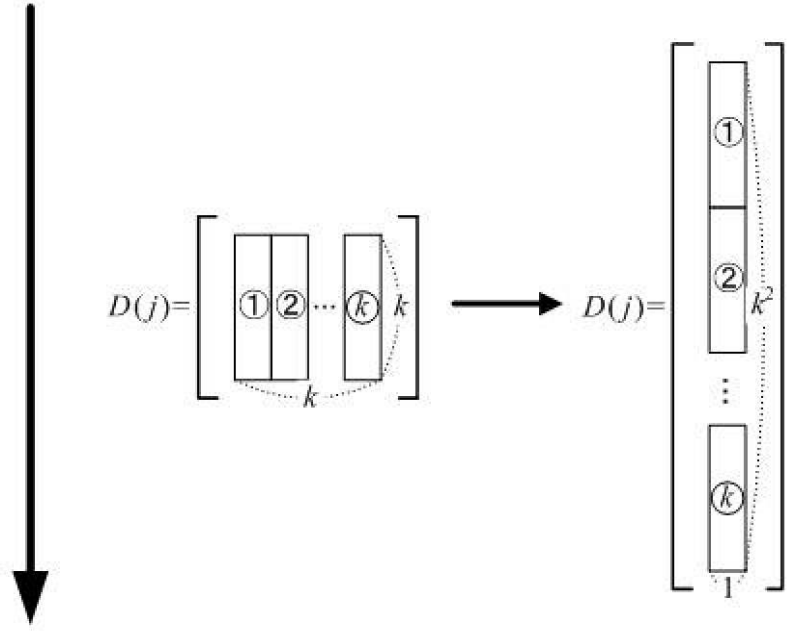
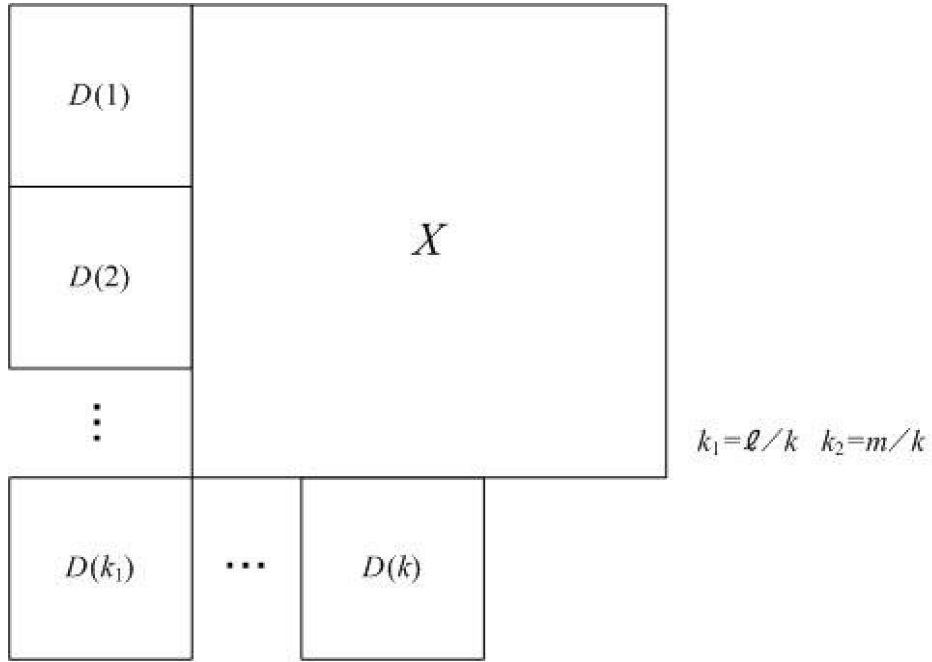
$k_1 = \ell/k \quad k_2 = m/k$



$$X(j) = \begin{bmatrix} \begin{array}{|c|c|c|} \hline \textcircled{1} & \textcircled{2} & \dots & \textcircled{k} \\ \hline \end{array} & k \end{bmatrix} \longrightarrow T(j) = \begin{bmatrix} \begin{array}{|c|} \hline \textcircled{1} \\ \hline \end{array} & k^2 \\ \vdots & \vdots \\ \begin{array}{|c|} \hline \textcircled{k} \\ \hline \end{array} & 1 \end{bmatrix}$$

$$T = \begin{bmatrix} \begin{array}{|c|c|} \hline T(1) & T(2) \\ \hline \end{array} & \dots & \begin{array}{|c|} \hline T(k_1) \\ \hline \end{array} & \dots & \dots & \begin{array}{|c|} \hline T(k_2k_1) \\ \hline \end{array} \end{bmatrix}$$

【 図 1 1 】





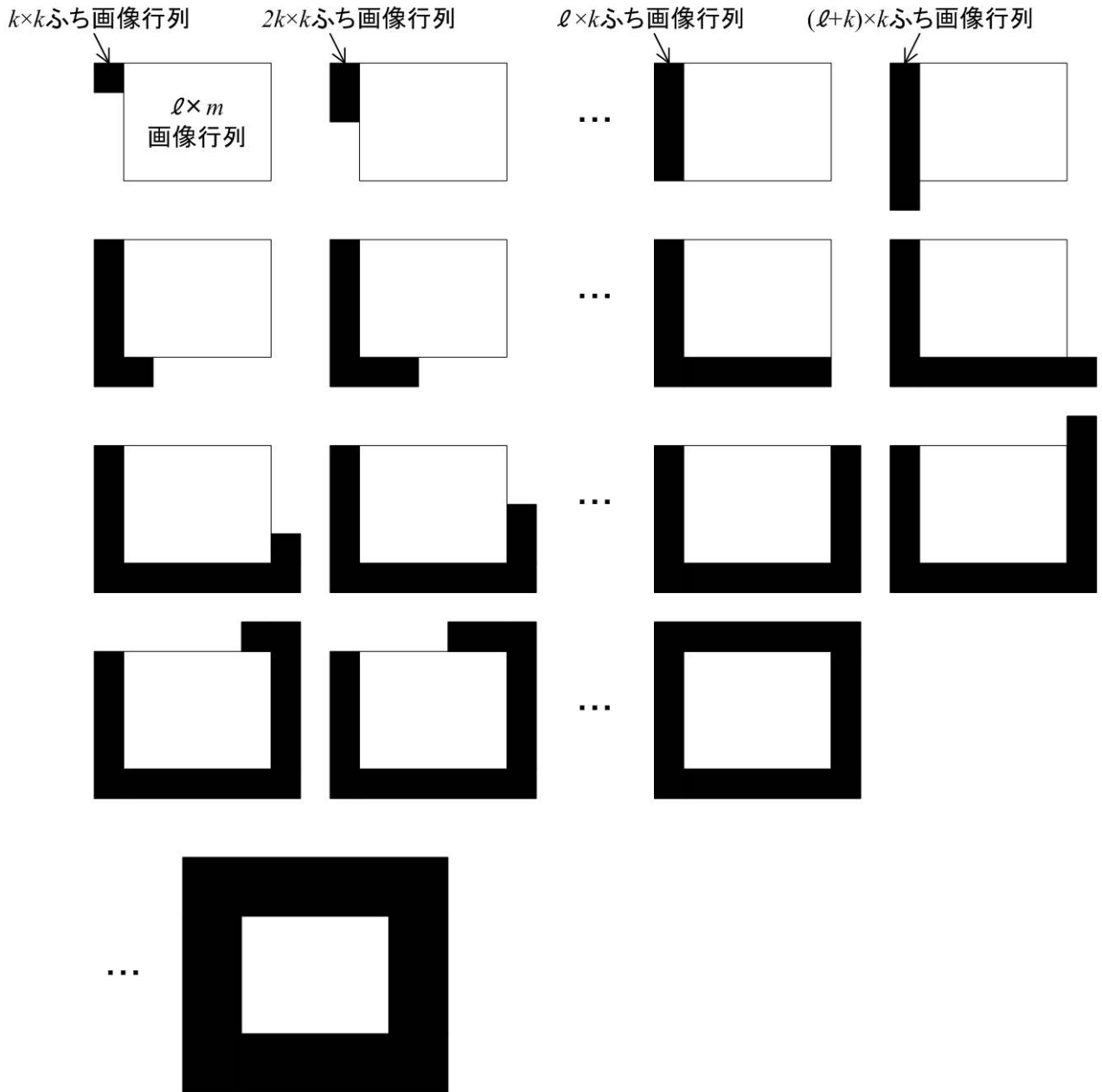
【 図 1 2 】

$$T_1 = \begin{bmatrix} T_1(1) \\ T_1(2) \\ \vdots \\ T_1(k) \\ \vdots \\ T_1(k^2) \end{bmatrix} \quad k_1 = \ell/k \quad k_2 = m/k$$

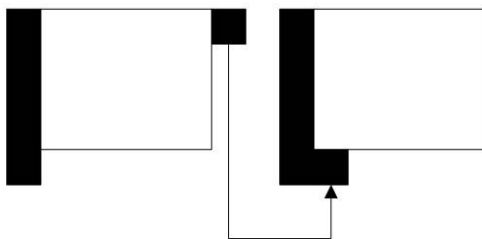
$$(T_1(j))^T = \begin{bmatrix} \textcircled{1} \\ \vdots \\ \textcircled{k_2} \end{bmatrix} \xrightarrow{\quad} X_1(j) = \begin{bmatrix} \textcircled{1} & \textcircled{2} & \dots & \textcircled{k_2} & k_1 \end{bmatrix}$$

$$X_1 = \begin{bmatrix} X_1(1) & X_1(k+1) & \dots & X_1(k^2-k+1) \\ X_1(2) & X_1(k+2) & \dots & X_1(k^2-k+2) \\ \vdots & \vdots & & \vdots \\ X_1(k) & X_1(2k) & \dots & X_1(k^2) \end{bmatrix}$$

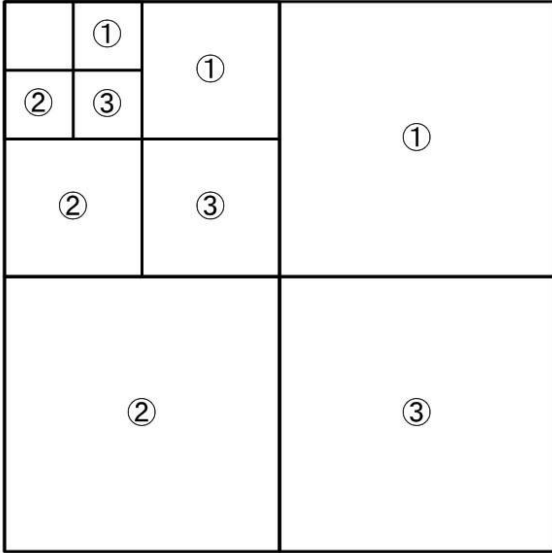
【 図 1 3 】



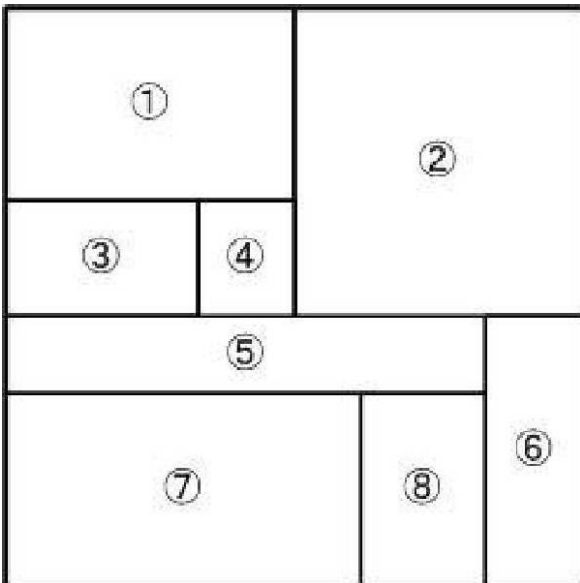
【 図 1 4 】



【 図 1 5 】



【 図 1 6 】



---

フロントページの続き

(72)発明者 中村 佳正

京都府京都市左京区吉田本町 国立大学法人京都大学内

(72)発明者 岩崎 雅史

京都府京都市左京区吉田本町 国立大学法人京都大学内

(72)発明者 小幡 雅彦

京都府京都市左京区吉田本町 国立大学法人京都大学内

(72)発明者 近藤 弘一

京都府京田辺市多々羅都谷 1 - 3 学校法人同志社 同志社大学内

(72)発明者 笹田 昇平

京都府京田辺市多々羅都谷 1 - 3 学校法人同志社 同志社大学内

Fターム(参考) 5B057 CA01 CA08 CA12 CA16 CB18 CG05 CG07 CH01 DA12 DB02  
DB09  
5C078 AA09 BA53 CA02 DA01