

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4325877号
(P4325877)

(45) 発行日 平成21年9月2日(2009.9.2)

(24) 登録日 平成21年6月19日(2009.6.19)

(51) Int.Cl.		F I			
G06F	17/16	(2006.01)	G06F	17/16	K
G06T	1/00	(2006.01)	G06T	1/00	315
G06F	17/30	(2006.01)	G06F	17/30	220Z
			G06F	17/30	350C

請求項の数 6 (全 39 頁)

(21) 出願番号	特願2006-514122 (P2006-514122)	(73) 特許権者	503360115
(86) (22) 出願日	平成17年6月1日(2005.6.1)		独立行政法人科学技術振興機構
(86) 国際出願番号	PCT/JP2005/010084		埼玉県川口市本町4丁目1番8号
(87) 国際公開番号	W02005/119507	(74) 代理人	100078282
(87) 国際公開日	平成17年12月15日(2005.12.15)		弁理士 山本 秀策
審査請求日	平成18年12月1日(2006.12.1)	(74) 代理人	100062409
(31) 優先権主張番号	特願2004-166437 (P2004-166437)		弁理士 安村 高明
(32) 優先日	平成16年6月3日(2004.6.3)	(74) 代理人	100113413
(33) 優先権主張国	日本国(JP)		弁理士 森下 夏樹
		(72) 発明者	中村 佳正
			大阪府豊中市緑丘1丁目21-25
		(72) 発明者	岩▲崎▼ 雅史
			大阪府松原市上田1-2-21アバイラ松原107

最終頁に続く

(54) 【発明の名称】 行列の高速高精度特異値分解方法、プログラムおよび装置

(57) 【特許請求の範囲】

【請求項7】

物体の複数の2次元画像から3次元画像を復元する方法であって、

2次元画像 j ($j = 1, \dots, m$, m は3以上の整数) における特徴点 i ($i = 1, \dots, n$, n は2以上の整数) の座標 d_{ij} (x_{ij}, y_{ij}) を抽出するステップと、
該特徴点の2次元座標 d_{ij} (x_{ij}, y_{ij}) より、該特徴点の3次元座標 s_i (X_i, Y_i, Z_i) および2次元座標から3次元座標への変換を表す行列 M を計算するステップと

を包含し、

該特徴点の3次元座標 s_i (X_i, Y_i, Z_i) および2次元座標から3次元座標への変換を表す行列 M を計算するステップは、行列 D に対して上2重対角化を行い、行列 D の上2重対角行列 B を求めるステップであって、行列 D は、

【数 1】

$$D = \begin{bmatrix} x_1^1 & \cdots & x_i^1 & \cdots & x_n^1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^j & \cdots & x_i^j & \cdots & x_n^j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^m & \cdots & x_i^m & \cdots & x_n^m \\ y_1^1 & \cdots & y_i^1 & \cdots & y_n^1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y_1^j & \cdots & y_i^j & \cdots & y_n^j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y_1^m & \cdots & y_i^m & \cdots & y_n^m \end{bmatrix}$$

10

と定義される、ステップと、

20

行列 D の特異値として行列 B の特異値 λ_j ($\lambda_1 > \lambda_2 > \cdots > \lambda_r > 0$ 、 r は、行列 D のランクに等しい) を求めるステップと、

λ_1 、 λ_2 および λ_3 に対する行列 D の特異ベクトルを求めるステップと、

$M = M' C$ なる行列 C に対して、 $E = C C^T$ を満たす行列 E を計算するステップであって、 $M' = L' (\Lambda')^{1/2}$ 、 Λ' は λ_1 、 λ_2 および λ_3 を対角要素に持ちそれ以外の要素が 0 である 3×3 行列、 L' は λ_1 、 λ_2 および λ_3 に対応する D の特異ベクトルを左から順に並べた行列である、ステップと、

行列 E から、行列 C を計算するステップと、

行列 C から、該 3 次元座標 s_i (X_i, Y_i, Z_i) および該変換を表す行列 M を計算するステップと

30

を含み、

該 λ_1 、 λ_2 および λ_3 に対する行列 D の特異ベクトルを求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、方法。

【請求項 8】

与えられた文書に含まれる情報であって、与えられたキーワードに関連する情報を検索する方法であって、

該キーワードに対応する質問ベクトル q を受け取るステップと、

該文書に対応する索引語文書行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求めるステップと、

40

行列 D の特異値として行列 B の特異値 λ_j ($\lambda_1 > \lambda_2 > \cdots > \lambda_r > 0$ 、 r は、行列 D のランクに等しい) を求めるステップと、

$k < r$ なる k を選択するステップと、

行列 D の擬似行列 D_k を計算するステップであって、行列 D_k は、 $\lambda_1, \lambda_2, \cdots, \lambda_k$ を対角要素としそれ以外の要素が 0 である行列 $\lambda_k, \lambda_1, \lambda_2, \cdots, \lambda_k$ に対応する特異ベクトルのみを左から順に並べた左右直交行列 U_k, V_k を用いて、 $D_k = U_k \Lambda_k V_k^T$ と定義される、ステップと、

行列 D_k と質問ベクトル q との類似度を計算するステップと、

該計算された類似度を基準に、検索結果を出力するステップと

50

を包含し、

該行列 D_k の左右直交行列 U_k, V_k を求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - j^2 I (j = 1, 2, \dots, k)$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、方法。

【請求項10】

物体の複数の2次元画像から3次元画像を復元する画像復元処理をコンピュータに実行させるプログラムであって、

該画像復元処理は、

2次元画像 $j (j = 1, \dots, m, m$ は3以上の整数) における特徴点 $i (i = 1, \dots, n, n$ は2以上の整数) の座標 $d_{ij} (x_{ij}, y_{ij})$ を抽出するステップと

10

、
該特徴点の2次元座標 $d_{ij} (x_{ij}, y_{ij})$ より、該特徴点の3次元座標 $s_i (X_i, Y_i, Z_i)$ および2次元座標から3次元座標への変換を表す行列 M を計算するステップと

を包含し、

該特徴点の3次元座標 $s_i (X_i, Y_i, Z_i)$ および2次元座標から3次元座標への変換を表す行列 M を計算するステップは、

行列 D に対して上2重対角化を行い、行列 D の上2重対角行列 B を求めるステップであって、行列 D は、

20

【数2】

$$D = \begin{bmatrix} x_1^1 & \dots & x_i^1 & \dots & x_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ x_1^j & \dots & x_i^j & \dots & x_n^j \\ \dots & \dots & \dots & \dots & \dots \\ x_1^m & \dots & x_i^m & \dots & x_n^m \\ y_1^1 & \dots & y_i^1 & \dots & y_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ y_1^j & \dots & y_i^j & \dots & y_n^j \\ \dots & \dots & \dots & \dots & \dots \\ y_1^m & \dots & y_i^m & \dots & y_n^m \end{bmatrix}$$

30

と定義される、ステップと、

40

行列 D の特異値として行列 B の特異値 $\lambda_j (j = 1, 2, \dots, r, r > 0, r$ は、行列 D のランクに等しい) を求めるステップと、

λ_1, λ_2 および λ_3 に対する行列 D の特異ベクトルを求めるステップと、

$M = M' C$ なる行列 C に対して、 $E = C C^T$ を満たす行列 E を計算するステップであって、 $M' = L' (\lambda_j)^{1/2}$ 、 L' は λ_1, λ_2 および λ_3 を対角要素に持ちそれ以外の要素が0である 3×3 行列、 L' は λ_1, λ_2 および λ_3 に対応する D の特異ベクトルを左から順に並べた行列である、ステップと、

行列 E から、行列 C を計算するステップと、

行列 C から、該3次元座標 $s_i (X_i, Y_i, Z_i)$ および該変換を表す行列 M を計算するステップと

50

を含み、

該 λ_1, λ_2 および λ_3 に対する行列 D の特異ベクトルを求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、プログラム。

【請求項 11】

与えられた文書に含まれる情報であって、与えられたキーワードに関連する情報を検索する文書検索処理をコンピュータに実行させるプログラムであって、

該文書検索処理は、

該キーワードに対応する質問ベクトル q を受け取るステップと、

該文書に対応する索引語文書行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求めるステップと、

行列 D の特異値として行列 B の特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$ 、 r は、行列 D のランクに等しい) を求めるステップと、

$k < r$ なる k を選択するステップと、

行列 D の擬似行列 D_k を計算するステップであって、行列 D_k は、 $\lambda_1, \lambda_2, \dots, \lambda_k$ を対角要素としそれ以外の要素が 0 である行列 D_k 、 $\lambda_1, \lambda_2, \dots, \lambda_k$ に対応する特異ベクトルのみを左から順に並べた左右直交行列 U_k, V_k を用いて、 $D_k = U_k D_k V_k^T$ と定義される、ステップと、

行列 D_k と質問ベクトル q との類似度を計算するステップと、

該計算された類似度を基準に、検索結果を出力するステップと

を包含し、

該行列 D_k の左右直交行列 U_k, V_k を求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ ($j = 1, 2, \dots, k$) に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、プログラム。

【請求項 13】

物体の複数の 2 次元画像を 3 次元画像に復元する装置であって、

m 枚 (m は 3 以上の整数) の 2 次元画像を受け取る手段と、

2 次元画像 j ($j = 1, \dots, m$) における特徴点 i ($i = 1, \dots, n$ 、 n は 2 以上の整数) の座標 d_{ij} (x_{ij}, y_{ij}) を抽出する手段と、

該特徴点の 2 次元座標 d_{ij} (x_{ij}, y_{ij}) より、該特徴点の 3 次元座標 s_i (X_i, Y_i, Z_i) および 2 次元座標から 3 次元座標への変換を表す行列 M を計算する手段と

を備え、

該特徴点の 3 次元座標 s_i (X_i, Y_i, Z_i) および 2 次元座標から 3 次元座標への変換を表す行列 M を計算する手段は、

行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求める手段であって、行列 D は、

10

20

30

【数3】

$$D = \begin{bmatrix} x_1^1 & \dots & x_i^1 & \dots & x_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ x_1^j & \dots & x_i^j & \dots & x_n^j \\ \dots & \dots & \dots & \dots & \dots \\ x_1^m & \dots & x_i^m & \dots & x_n^m \\ y_1^1 & \dots & y_i^1 & \dots & y_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ y_1^j & \dots & y_i^j & \dots & y_n^j \\ \dots & \dots & \dots & \dots & \dots \\ y_1^m & \dots & y_i^m & \dots & y_n^m \end{bmatrix}$$

10

と定義される、手段と、

20

行列Dの特異値として行列Bの特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$, r は、行列Dのランクに等しい)を求める手段と、

λ_1, λ_2 および λ_3 に対する行列Dの特異ベクトルを求める手段と、

$M = M' C$ なる行列Cに対して、 $E = C C^T$ を満たす行列Eを計算する手段であって、 $M' = L' (\Lambda')^{1/2}$ 、 Λ' は λ_1, λ_2 および λ_3 を対角要素に持ちそれ以外の要素が0である 3×3 行列、 L' は λ_1, λ_2 および λ_3 に対応するDの特異ベクトルを左から順に並べた行列である、手段と、

行列Eから、行列Cを計算する手段と、

行列Cから、該3次元座標 $s_i (X_i, Y_i, Z_i)$ および該変換を表す行列Mを計算する手段と

30

を含み、

該 λ_1, λ_2 および λ_3 に対する行列Dの特異ベクトルを求める手段は、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ に対してツイステッド分解を実行することにより、行列 $B^T B$ を対角化する手段を含み、 I は単位行列である、装置。

【請求項14】

与えられた文書に含まれる情報であって、与えられたキーワードに関連する情報を検索する装置であって、

該キーワードに対応する質問ベクトルqを受け取る手段と、

該文書に対応する索引語文書行列Dに対して上2重対角化を行い、行列Dの上2重対角行列Bを求める手段と、

40

行列Dの特異値として行列Bの特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$, r は、行列Dのランクに等しい)を求める手段と、

$k < r$ なるkを選択する手段と、

行列Dの擬似行列 D_k を計算する手段であって、行列 D_k は、 $\lambda_1, \lambda_2, \dots, \lambda_k$ を対角要素としそれ以外の要素が0である行列 $\lambda_k, \lambda_1, \lambda_2, \dots, \lambda_k$ に対応する特異ベクトルのみを左から順に並べた左右直交行列 U_k, V_k を用いて、 $D_k = U_k V_k^T$ と定義される、手段と、

行列 D_k と質問ベクトルqとの類似度を計算する手段と、

該計算された類似度を基準に、検索結果を出力する手段と

50

を包含し、

該行列 D_k の左右直交行列 U_k, V_k を求める手段は、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I (j = 1, 2, \dots, k)$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化する手段を含み、 I は単位行列である、装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、任意の行列 A に対して高速かつ高精度に特異値分解を実行する方法、プログラムおよび装置に関し、より詳細には、行列の特異値を計算し、計算された特異値に対する特異ベクトルを計算することによって、高速かつ高精度に特異値分解を実行する方法、プログラムおよび装置に関する。

10

【背景技術】

【0002】

現在、コンピュータ上で特異値分解を行うために用いられる標準的な方法は、国際的な数値計算ライブラリ LAPACK により公開されている DBDSQR ルーチンである。DBDSQR ルーチンは、QR 法に基づいて作られたものであり、前処理、メインループ、後処理の3つの部分に分けることができる。前処理では、特異値の上限および下限を計算し、収束判定に用いる精度を計算する。メインループでは、QR 法を繰り返しながら、除々に行列の分割および減次を行い、最終的に行列サイズが1となった時点でループを終了する。途中で 2×2 行列のブロックが現れたならば別処理を行う。後処理では、特異値として計算された値が負ならば正に直し、必要ならば特異ベクトルをスカラー倍する。特異値を降順に、それに対応するように特異ベクトルも並べ替える。DBDSQR は、計算量が非常に多く、特に大規模問題では計算時間の増大が避けられない。DBDSQR ルーチンでは、特異値および特異ベクトルが同時に計算される。また、LAPACK には、特異値を計算する DLASQ ルーチン、および、計算された特異値を用いて特異ベクトルを計算する際に用いる対角化のための DSTEGR ルーチンがある。DLASQ ルーチンは、高速高精度に特異値を求めることができるが、特異ベクトルを計算できない。DSTEGR ルーチンは、その数値的な欠点から、実際に特異値分解に使用することは難しい。

20

【0003】

LAPACK による DBDSQR ルーチンを例として、従来の特異値分解方法を説明する。DBDSQR ルーチンでは、 $l_1 \times l_2$ の一般行列 A を特異値分解するために、まず、行列 A に対して Householder (ハウスホルダー) 変換を適用する。すなわち、行列 A は、直交行列 U_A, V_A を用いて、

30

【0004】

【数4】

$$U_A^T A V_A = \begin{cases} (B \ 0) & (l_1 \leq l_2) \\ \begin{pmatrix} B \\ 0 \end{pmatrix} & (l_1 \geq l_2) \end{cases}, \quad B = \begin{pmatrix} b_1 & b_2 & & \\ & b_3 & \dots & \\ & & \dots & b_{2m-2} \\ 0 & & & b_{2m-1} \end{pmatrix}$$

40

$$U_A^T U_A = I, \quad V_A^T V_A = I, \quad m = \min\{l_1, l_2\}$$

と表わすことができる。このとき得られる B を、上2重対角行列という。ここで、 A の特異値 = B の特異値であることに注意する。このように、一般行列 A に対する特異値分解問題を上2重対角行列 B に対する特異値分解問題

$$B = U_B \quad V_B^T$$

行列 U_B, V_B は、それぞれ左および右直交行列

$$\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_m) \quad \lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_m \quad 0$$

λ_j は B の特異値

50

に置き換える。

【0005】

次に、行列 $B^T B$ を考える。この行列 $B^T B$ の対角化

$$= V^T B^T B V$$

$$\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_m) \quad \lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_m \quad 0$$

$$V = (v_1, v_2, \dots, v_m)$$

λ_j は $B^T B$ の固有値

v_j は固有値 λ_j に対する固有ベクトル

を行う。ここで、一般に、以下のことが成り立つ。(1) $B^T B$ は対称な3重対角行列で

ある。(2) $B^T B$ の固有値は全て正であり、 B の特異値 λ_j ($\lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_m \quad 0$) は、 $B^T B$ の固有値 λ_j ($\lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_m \quad 0$) の正の平方根に等しい。

(3) $V_B = V$ 、すなわち、 $B^T B$ の固有ベクトル v_j は、 B の右特異ベクトルに等しい。

従って、行列 $B^T B$ の対角化が求まると、(2) より $\lambda_j = \sigma_j^2$ であることから、

σ_j が求まり、さらに(3)より左直交行列 $U_B = B V_B^{-1} = B V^{-1}$ が求まるので、

B が特異値分解される。すなわち、 B の特異値分解を、 $B^T B$ の対角化の問題に置き換えることができる。

この原理は、 m 個全ての特異値および特異ベクトルを求める場合だけでなく、少なくとも1つの特異値および特異ベクトルを求める場合にも適用され得る。

【0006】

以上のように、一般行列 A の特異値分解は、対称な3重対角行列 $B^T B$ の対角化の問題

を含む。

【発明の開示】

【発明が解決しようとする課題】

【0007】

DBDSQRルーチンでは、計算量が非常に多いため、特に大規模問題では、高速に特異値分解を実行することが困難であった。一方、DLASQRルーチンは、高速かつ高精度に特異値を求めることができるが、DSTEGRルーチンは、場合によっては特異ベクトルを低精度で計算するため、DSTEGRルーチンでは、常に高精度に特異値分解を実行することは困難であった。

【0008】

本発明の目的の1つは、対称な3重対角行列 $B^T B$ の対角化を高速かつ高精度で実行することにより、任意の行列 A の特異値分解を高速かつ高精度に実行することを可能にする方法、プログラム、および装置を提供することにある。ここで、行列 B を、行列 A をハウスホルダー変換することで得られる上2重対角行列とする。

【課題を解決するための手段】

【0009】

本発明では、ミウラ型逆変換、sdLVvs変換、rdLVvs変換およびミウラ型変換によるTwisted(ツイステッド)分解を実行することにより、行列 $B^T B$ を対角化する。ここで、行列 B を、行列 A と特異値が同じ上2重対角行列とする。行列 A から行列 B への変換は、例えばハウスホルダー法で実行できる。本発明による行列 $B^T B$ の対角化では、固有値および固有ベクトルを同時に求めるDBDSQRルーチンとは異なり、DSTEGRルーチンのように、まず固有値を計算し、次に計算された固有値を用いて固有ベクトルを求める。

【0010】

本発明により、コンピュータを用いて任意の行列 A に対して特異値分解を実行する方法であって、行列 A に対して上2重対角化を行い、行列 A の上2重対角行列 B を求めるステップと、行列 A の特異値として行列 B の少なくとも1つの特異値 λ_j を求めるステップと、 λ_j に対する行列 A の特異ベクトルを求めるステップとを包含し、該行列 A の特異ベクトルを求めるステップは、ミウラ型逆変換と、sdLVvs変換と、rdLVvs変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ に対してツイステッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、方法が提供され、

これにより上記目的が達成される。

【0011】

前記行列 A の特異ベクトルを求めるステップは、前記行列 $B^T B$ を対角化するステップの後に、逆反復法を実行するステップを含んでもよい。

【0012】

前記行列 A の特異ベクトルを求めるステップは、前記逆反復法を実行するステップの後に、グラムシュミット法を実行するステップを含んでもよい。

【0013】

前記行列 B の特異値 を求めるステップは、d L V s ルーチンを実行するステップを含んでもよい。

10

【0014】

前記行列 B の特異値 を求めるステップは、要求される計算時間および精度に応じて、d L V s ルーチンもしくは D L A S Q ルーチンのどちらを実行するのかを判定するステップを含んでもよい。

【0015】

前記行列 A に対して上 2 重対角化を行い、上 2 重対角行列 B を求めるステップは、上 2 重対角化法（たとえばハウスホルダー法）を実行するステップを含んでもよい。

【0016】

本発明により、物体の複数の 2 次元画像から 3 次元画像を復元する方法であって、2 次元画像 j ($j = 1, \dots, m$ 、 m は 3 以上の整数) における特徴点 i ($i = 1, \dots, n$ 、 n は 2 以上の整数) の座標 d_{ij} (x_{ij}, y_{ij}) を抽出するステップと、該特徴点の 2 次元座標 d_{ij} (x_{ij}, y_{ij}) より、該特徴点の 3 次元座標 s_i (X_i, Y_i, Z_i) および 2 次元座標から 3 次元座標への変換を表す行列 M を計算するステップとを包含し、該特徴点の 3 次元座標 s_i (X_i, Y_i, Z_i) および 2 次元座標から 3 次元座標への変換を表す行列 M を計算するステップは、行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求めるステップであって、行列 D は、

20

【0017】

【数 5】

$$D = \begin{bmatrix} x_1^1 & \cdots & x_i^1 & \cdots & x_n^1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^j & \cdots & x_i^j & \cdots & x_n^j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^m & \cdots & x_i^m & \cdots & x_n^m \\ y_1^1 & \cdots & y_i^1 & \cdots & y_n^1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y_1^j & \cdots & y_i^j & \cdots & y_n^j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y_1^m & \cdots & y_i^m & \cdots & y_n^m \end{bmatrix}$$

30

40

と定義される、ステップと、行列 D の特異値として行列 B の特異値 j (1 2 \cdots 50

・ ・ ・ $r > 0$ 、 r は、行列 D のランクに等しい) を求めるステップと、 λ_1 、 λ_2 および λ_3 に対する行列 D の特異ベクトルを求めるステップと、 $M = M' C$ なる行列 C に対して、 $E = C C^T$ を満たす行列 E を計算するステップであって、 $M' = L' (\lambda_1 \lambda_2 \lambda_3)^{1/2}$ 、 L' は λ_1 、 λ_2 および λ_3 を対角要素に持ちそれ以外の要素が 0 である 3×3 行列、 L' は λ_1 、 λ_2 および λ_3 に対応する D の特異ベクトルを左から順に並べた行列である、ステップと、行列 E から、行列 C を計算するステップと、行列 C から、該 3 次元座標 $s_i (X_i, Y_i, Z_i)$ および該変換を表す行列 M を計算するステップとを含み、該 λ_1 、 λ_2 および λ_3 に対する行列 D の特異ベクトルを求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - j^2 I$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、方法が提供され、これにより上記目的が達成される。

10

【0018】

本発明により、与えられた文書に含まれる情報であって、与えられたキーワードに関連する情報を検索する方法であって、該キーワードに対応する質問ベクトル q を受け取るステップと、該文書に対応する索引語文書行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求めるステップと、行列 D の特異値として行列 B の特異値 $\lambda_j (\lambda_1 \lambda_2 \dots \lambda_r > 0$ 、 r は、行列 D のランクに等しい) を求めるステップと、 $k < r$ なる k を選択するステップと、行列 D の擬似行列 D_k を計算するステップであって、行列 D_k は、 $\lambda_1, \lambda_2, \dots, \lambda_k$ を対角要素としそれ以外の要素が 0 である行列 D_k 、 $\lambda_1, \lambda_2, \dots, \lambda_k$ に対応する特異ベクトルのみを左から順に並べた左右直交行列 U_k, V_k を用いて、 $D_k = U_k D_k V_k^T$ と定義される、ステップと、行列 D_k と質問ベクトル q との類似度を計算するステップと、該計算された類似度を基準に、検索結果を出力するステップとを包含し、該行列 D_k の左右直交行列 U_k, V_k を求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - j^2 I (j = 1, 2, \dots, k)$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、方法が提供され、これにより上記目的が達成される。

20

【0019】

本発明により、任意の行列 A に対して特異値分解を実行する特異値分解処理をコンピュータに実行させるプログラムであって、該特異値分解処理は、行列 A に対して上 2 重対角化を行い、行列 A の上 2 重対角行列 B を求めるステップと、行列 A の特異値として行列 B の少なくとも 1 つの特異値 λ_j を求めるステップと、 λ_j に対する行列 A の特異ベクトルを求めるステップとを包含し、該行列 A の特異ベクトルを求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - j^2 I$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、 I は単位行列である、プログラムが提供され、これにより上記目的が達成される。

30

【0020】

本発明により、物体の複数の 2 次元画像から 3 次元画像を復元する画像復元処理をコンピュータに実行させるプログラムであって、該画像復元処理は、2 次元画像 $j (j = 1, \dots, m, m$ は 3 以上の整数) における特徴点 $i (i = 1, \dots, n, n$ は 2 以上の整数) の座標 $d_{ij} (x_{ij}, y_{ij})$ を抽出するステップと、該特徴点の 2 次元座標 $d_{ij} (x_{ij}, y_{ij})$ より、該特徴点の 3 次元座標 $s_i (X_i, Y_i, Z_i)$ および 2 次元座標から 3 次元座標への変換を表す行列 M を計算するステップとを包含し、該特徴点の 3 次元座標 $s_i (X_i, Y_i, Z_i)$ および 2 次元座標から 3 次元座標への変換を表す行列 M を計算するステップは、行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求めるステップであって、行列 D は、

40

【0021】

【数6】

$$D = \begin{bmatrix} x_1^1 & \dots & x_i^1 & \dots & x_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ x_1^j & \dots & x_i^j & \dots & x_n^j \\ \dots & \dots & \dots & \dots & \dots \\ x_1^m & \dots & x_i^m & \dots & x_n^m \\ y_1^1 & \dots & y_i^1 & \dots & y_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ y_1^j & \dots & y_i^j & \dots & y_n^j \\ \dots & \dots & \dots & \dots & \dots \\ y_1^m & \dots & y_i^m & \dots & y_n^m \end{bmatrix}$$

10

20

と定義される、ステップと、行列Dの特異値として行列Bの特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$, r は、行列Dのランクに等しい)を求めるステップと、 λ_1, λ_2 および λ_3 に対する行列Dの特異ベクトルを求めるステップと、 $M = M' C$ なる行列Cに対して、 $E = C C^T$ を満たす行列Eを計算するステップであって、 $M' = L' (\Lambda')^{1/2}$ 、 Λ' は λ_1, λ_2 および λ_3 を対角要素に持ちそれ以外の要素が0である 3×3 行列、 L' は λ_1, λ_2 および λ_3 に対応するDの特異ベクトルを左から順に並べた行列である、ステップと、行列Eから、行列Cを計算するステップと、行列Cから、該3次元座標 s_i (X_i, Y_i, Z_i) および該変換を表す行列Mを計算するステップとを含み、該 λ_1, λ_2 および λ_3 に対する行列Dの特異ベクトルを求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化するステップを含み、Iは単位行列である、プログラムが提供され、これにより上記目的が達成される。

30

【0022】

本発明により、与えられた文書に含まれる情報であって、与えられたキーワードに関連する情報を検索する文書検索処理をコンピュータに実行させるプログラムであって、該文書検索処理は、該キーワードに対応する質問ベクトルqを受け取るステップと、該文書に対応する索引語文書行列Dに対して上2重対角化を行い、行列Dの上2重対角行列Bを求めるステップと、行列Dの特異値として行列Bの特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$, r は、行列Dのランクに等しい)を求めるステップと、 $k < r$ なるkを選択するステップと、行列Dの擬似行列 D_k を計算するステップであって、行列 D_k は、 $\lambda_1, \lambda_2, \dots, \lambda_k$ を対角要素としそれ以外の要素が0である行列 Λ_k 、 $\lambda_1, \lambda_2, \dots, \lambda_k$ に対応する特異ベクトルのみを左から順に並べた左右直交行列 U_k, V_k を用いて、 $D_k = U_k \Lambda_k V_k^T$ と定義される、ステップと、行列 D_k と質問ベクトルqとの類似度を計算するステップと、該計算された類似度を基準に、検索結果を出力するステップとを包含し、該行列 D_k の左右直交行列 U_k, V_k を求めるステップは、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ ($j = 1, 2, \dots, k$) に対してツイスティッド分解を実行することにより、行

40

50

列 $B^T B$ を対角化するステップを含み、 I は単位行列である、プログラムが提供され、これにより上記目的が達成される。

【0023】

本発明により、任意の行列 A に対して特異値分解を実行する装置であって、行列 A を入力として受け取る手段と、行列 A に対して上 2 重対角化を行い、行列 A の上 2 重対角行列 B を計算する手段と、行列 A の特異値として行列 B の少なくとも 1 つの特異値を計算する手段と、に対する行列 A の特異ベクトルを求める手段とを備え、該行列 A の特異ベクトルを求める手段は、ミウラ型逆変換、 $s d L V v s$ 変換および $r d L V v s$ 変換、ならびにミウラ型変換を用いて行列 $B^T B - \lambda^2 I$ に対してツイステッド分解を実行することにより、行列 $B^T B$ を対角化する手段を含み、 I は単位行列である、装置が提供され、

10

【0024】

本発明により、物体の複数の 2 次元画像を 3 次元画像に復元する装置であって、 m 枚 (m は 3 以上の整数) の 2 次元画像を受け取る手段と、2 次元画像 j ($j = 1, \dots, m$) における特徴点 i ($i = 1, \dots, n$ 、 n は 2 以上の整数) の座標 d_{ij} (x_{ij}, y_{ij}) を抽出する手段と、該特徴点の 2 次元座標 d_{ij} (x_{ij}, y_{ij}) より、該特徴点の 3 次元座標 s_i (X_i, Y_i, Z_i) および 2 次元座標から 3 次元座標への変換を表す行列 M を計算する手段とを備え、該特徴点の 3 次元座標 s_i (X_i, Y_i, Z_i) および 2 次元座標から 3 次元座標への変換を表す行列 M を計算する手段は、行列 D に対して上 2 重対角化を行い、行列 D の上 2 重対角行列 B を求める手段であって、行列 D は、

20

【0025】

【数 7】

$$D = \begin{bmatrix} x_1^1 & \dots & x_i^1 & \dots & x_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ x_1^j & \dots & x_i^j & \dots & x_n^j \\ \dots & \dots & \dots & \dots & \dots \\ x_1^m & \dots & x_i^m & \dots & x_n^m \\ y_1^1 & \dots & y_i^1 & \dots & y_n^1 \\ \dots & \dots & \dots & \dots & \dots \\ y_1^j & \dots & y_i^j & \dots & y_n^j \\ \dots & \dots & \dots & \dots & \dots \\ y_1^m & \dots & y_i^m & \dots & y_n^m \end{bmatrix}$$

30

40

と定義される、手段と、行列 D の特異値として行列 B の特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$ 、 r は、行列 D のランクに等しい) を求める手段と、 λ_1, λ_2 および λ_3 に対する行列 D の特異ベクトルを求める手段と、 $M = M' C$ なる行列 C に対して、 $E = C C^T$ を満たす行列 E を計算する手段であって、 $M' = L' (\lambda_j)^{1/2}$ 、 L' は λ_1, λ_2 および λ_3 を対角要素に持ちそれ以外の要素が 0 である 3×3 行列、 L' は λ_1, λ_2 および λ_3 に対応する D の特異ベクトルを左から順に並べた行列である、手段と、行列 E から、行列 C を計算する手段と、行列 C から、該 3 次元座標 s_i (X_i, Y_i, Z_i) お

50

よび該変換を表す行列Mを計算する手段とを含み、該 λ_1 、 λ_2 および λ_3 に対する行列Dの特異ベクトルを求める手段は、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化する手段を含み、Iは単位行列である、装置が提供され、これにより上記目的が達成される。

【0026】

本発明により、与えられた文書に含まれる情報であって、与えられたキーワードに関連する情報を検索する装置であって、該キーワードに対応する質問ベクトルqを受け取る手段と、該文書に対応する索引語文書行列Dに対して上2重対角化を行い、行列Dの上2重対角行列Bを求める手段と、行列Dの特異値として行列Bの特異値 λ_j ($\lambda_1, \lambda_2, \dots, \lambda_r > 0$, rは、行列Dのランクに等しい)を求める手段と、 $k < r$ なるkを選択する手段と、行列Dの擬似行列 D_k を計算する手段であって、行列 D_k は、 $\lambda_1, \lambda_2, \dots, \lambda_k$ を対角要素としそれ以外の要素が0である行列 D_k 、 $\lambda_1, \lambda_2, \dots, \lambda_k$ に対応する特異ベクトルのみを左から順に並べた左右直交行列 U_k, V_k を用いて、 $D_k = U_k D_k V_k^T$ と定義される、手段と、行列 D_k と質問ベクトルqとの類似度を計算する手段と、該計算された類似度を基準に、検索結果を出力する手段とを包含し、該行列 D_k の左右直交行列 U_k, V_k を求める手段は、ミウラ型逆変換と、 $s d L V v s$ 変換と、 $r d L V v s$ 変換と、ミウラ型変換とを用いて行列 $B^T B - \lambda_j^2 I$ ($j = 1, 2, \dots, k$) に対してツイスティッド分解を実行することにより、行列 $B^T B$ を対角化する手段を含み、Iは単位行列である、装置が提供され、これにより上記目的が達成される。

10
20

【発明の効果】

【0027】

本発明によれば、ミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換によるツイスティッド分解を実行することにより、行列 $B^T B$ が対角化される。それにより、D S T E G R ルーチンと比較した場合、精度の点ではるかに優れ、実用化に耐え得る固有値計算を実現し、高精度の特異値分解が可能になる。また、逆反復法、Gram-Schmidt (グラムシュミット) 法、本発明において提供する $d L V s$ ルーチンのうちの少なくとも1つを併用することにより、D B D S Q R と比較して精度の点では同等になることもあるが、速度の点では圧倒的に高速である特異値分解を実現することができる。さらに、D B D S Q R ルーチンと比較すると、固有値および固有ベクトルを同時の求めるのか、または、計算された固有値を用いて固有ベクトルを求めるのかの計算フローの違いにより、以下で説明する2次元から3次元への画像復元問題および文書検索問題等においては、計算時間をさらに短縮することができる。

30

【図面の簡単な説明】

【0028】

【図1】図1は、本発明により提供される特異値分解のためのI-SVDルーチンの処理の手順を示すフローチャート

【図2】図2は、Parlettらの手法による $s q d s$ 変換および $r p q d s$ 変換の処理の手順を示すフローチャート

40

【図3】図3は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換の処理の手順を示すフローチャート

【図4】図4は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換の処理の手順を示すフローチャート

【図5】図5は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換の処理の手順を示すフローチャート。

【図6】図6は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換の処理の手順を示すフローチャート

【図7】図7は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換の処理の手順を示すフローチャート

50

【図 8】図 8 は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換の処理の手順を示すフローチャート

【図 9】図 9 は、本発明において誤差の低減をもたらす、Parlett の手法と本発明による方法との計算経路の違いを示す図

【図 10】図 10 は、本発明による特異値分解装置の 1 つの実施形態であるコンピュータを示す図

【図 11】図 11 は、本発明による特異値分解方法を利用した物体の複数の 2 次元画像を 3 次元画像へ復元する画像処理の 1 つの実施形態を示すフローチャート

【図 12】図 12 は、本発明による特異値分解装置を利用した文書検索方法の 1 つの実施形態を示すフローチャート

10

【符号の説明】

【0029】

10 コンピュータ

1001 CPU

1002 メモリ

1003 入力インターフェース部

1004 出力インターフェース部

1005 バス

【発明を実施するための最良の形態】

【0030】

20

以下、図面を参照しながら、本発明の実施形態を説明する。

【0031】

1. 特異値分解アルゴリズム I - SVD ルーチン

図 1 を参照する。図 1 は、本発明により提供される特異値分解のための I - SVD ルーチンの処理の手順を示す。図 1 の処理は、コンピュータプログラムの形式で提供され得る。

【0032】

ステップ 102 において、一般行列 A に対してハウスホルダー法を用いて上 2 重対角化を行う。このとき利用される上 2 重対角化方法は、ハウスホルダー法であってもよいし、他の上 2 重対角化方法であってもよい。これにより、上 2 重対角行列 B が得られる。次に、行列 B の特異値 σ_j ($\sigma_1, \sigma_2, \dots, \sigma_m, 0$ 、 m は、行列 $B^T B$ の行列サイズに等しい) を計算する。ここで、本発明により提供される特異値計算ルーチン $d L V s$ および $L A P A C K$ により提供される $D L A S Q$ ルーチンを利用し得る。なお、行列 B の特異値は、行列 A の特異値に等しく、かつ、行列 $B^T B$ の固有値の正の平方根 $\sigma_j^{1/2}$ に等しい。 $d L V s$ ルーチンは、B の特異値 σ_j を高精度で計算する。 $d L V s$ ルーチンの詳細は、後述される。 $d L V s$ で求められる特異値は、従来の $L A P C K$ ルーチン ($D B D S Q R$ 、 $D L A S Q$ 等) と比較して、最も高精度である。速度に関しては、ほとんどの場合 $D L A S Q$ ルーチンには劣るが、信頼性の高い $D B D S Q R$ ルーチンの約 2 分の 1 の時間で計算できる (CPU が *Itanium 2* の場合は $d L V s$ ルーチンが最速)。そこで、ステップ 104 において、高精度性が必要とされるかどうかを判定する。最も高精度な特異値分解を行いたい場合には、ステップ 106 において $d L V s$ を、高精度性および高速性を両立したい場合には、ステップ 108 において従来の $D L A S Q$ ルーチンを、特異値計算ルーチンとして用いるのがよい。さらに、 $D L A S Q$ ルーチンが有限回の演算で停止するかは定かでないが、 $d L V s$ ルーチンは収束することが保証されているので、信頼性を重視するならば、 $d L V s$ ルーチンを用いるのがよい。

30

40

【0033】

次に、ステップ 110 において、計算された B の特異値 σ_j (すなわち、A の特異値に等しい) を用いて、1.1 において説明される $s d L V v s$ 変換および $r d L V v s$ 変換によるツイステッド分解により、 $B^T B$ の対角化を行う。本発明による行列 $B^T B$ の対角化ルーチンの詳細は、1.1 で説明する。さらに、ステップ 112 において、以下の 1.

50

1にて詳細が示される簡略化連立1次方程式

【0034】

【数8】

$$N_{k_0}^T v_j = e_{k_0}, \quad e_{k_0} \equiv (\underbrace{0, \dots, 0}_{(k_0-1)\text{個}}, 1, 0, \dots, 0)^T$$

を求めると、行列Bの特異ベクトル v_j が得られる(すなわち、 $B^T B$ の対角化行列Vが得られる)。すなわち、 $B^T B$ が対角化される。ステップ114において、 $\text{diag}(d_1, d_2, \dots, d_m)$ (ただし、 $d_1 \geq d_2 \geq \dots \geq d_m > 0$)とすると、行列Bの右直交行列 $V_B = V$ であるので、行列Bの左直交行列 $U_B = B V_B^{-1} = B V^{-1}$ から U_B を得る。すなわち、行列Bが特異値分解される。ここでは、m個全ての特異値および特異ベクトルを計算したが、この原理は、少なくとも1つの特異値および特異ベクトルを計算する場合にも適用され得る。

10

【0035】

このとき得られる左右直交行列、すなわち特異ベクトル計算の結果は、DBDSQRルーチンによって得られる結果と比較して精度の点で劣る。そこで、ステップ116において、逆反復法を用いることにより、DBDSQRルーチンと同程度の精度を得ることができる。さらなる高精度性が必要とされる場合は、ステップ120において、グラムシュミット法による再直交化を行えば、DBDSQRよりも高精度での計算を実現し得る。I-SVDルーチンでは、速度に関しては、表2から、DBDSQRと比較して大幅に短縮されることが理解される。また、表1から理解されるように、I-SVDルーチンは、DSTEGRによる極度の精度悪化がみられる場合でも、高精度に特異値分解することができる。なお、表1、2の誤差および計算時間に関するデータは、逆反復法を用いているが、グラムシュミット法は用いていない場合の結果である。また、ステップ116および120を実行する順序は逆であってもよいし、あるいは、両ステップは、省略してもよい。

20

【0036】

このように、図1に示される各ステップの機能は、ソフトウェア(例えば、プログラム)によって実現される。しかし、本発明は、これに限定されない。図1に示される各ステップの機能をハードウェア(例えば、回路、ボード、半導体チップ)によって実現してもよいし、ソフトウェアとハードウェアとの組み合わせによって実現してもよい。

【0037】

30

以上により、I-SVDルーチンを用いると、従来技術と比較して高速かつ高精度の特異値分解が実行される。

【0038】

表1は、本発明による行列 $B^T B$ 対角化ルーチンを実行した後に、逆反復法を実行したときの従来技術および本発明の誤差を比較した表である。

【0039】

【表1】

表1: 誤差評価

	I-SVD (dLVsを含む)	DBDSQR	DSTEGR
$\ B - U \Sigma V^T\ _\infty$	1.69E-13	1.31E-13	4.40E+04
$\ U^T U - I\ _\infty$	2.09E-13	1.26E-13	—
$\ V^T V - I\ _\infty$	4.16E-13	1.21E-13	2.04E+04

40

表2は、本発明による行列 $B^T B$ 対角化ルーチンを実行した後に、逆反復法を実行したときの従来技術および本発明の計算時間を比較した表である。

【0040】

【表 2】

表2:計算時間(秒)

行列サイズ	I-SVD (dLVsを含む)	DBDSQR	DSTEGR
1000	1.12	56.85	0.20
2000	4.88	1175.34	0.84
3000	11.84	4879.39	1.78

また、1.3に、本発明の方法において、DSTEGRルーチンと比較して、誤差が低減される理由を説明する。

10

【0041】

1.1.本発明による行列 $B^T B$ の対角化ルーチン

DBDSQRは $B^T B$ の固有値 λ_j とともに対応する固有ベクトル v_j を求めることができるので、他のルーチンを併用しなくても $B^T B$ の対角化 (B の特異値分解) が得られる。しかしながら、DLASQおよびdLVsは上述したように固有ベクトル v_j を計算する機能はない。よって、別の固有ベクトル計算ルーチンが必要となる。ただし、 $B^T B$ の固有値 λ_j (B の特異値 σ_j) が求まっているものとする。

【0042】

まず、DSTEGRによる固有ベクトルの計算方法を示す。Parlettらは以下の(1)、(2)、(3)の手順で固有ベクトル計算ができることを示している。

20

(1) stationary qd with shift変換 (stqds変換) および reverse-time progressive qd with shift変換 (rpqds変換) を用いた $(B^{(0)})^T B^{(0)} - \lambda_j I$ のCholesky (コレスキー) 分解 $(B^{(\pm 1)})^T B^{(\pm 1)} = (B^{(0)})^T B^{(0)} - \lambda_j I$ を求める。ただし、 $B^{(0)} = B$ であり、 $B^{(+1)}$ および $B^{(-1)}$ はそれぞれ上2重対角および下2重対角行列とする。すなわち

【0043】

【数9】

$$B^{(\pm 1)} \equiv \begin{pmatrix} b_1^{(\pm 1)} & b_2^{(\pm 1)} & & & \\ & b_3^{(\pm 1)} & \dots & & \\ & & \dots & b_{2m-2}^{(\pm 1)} & \\ & & & & b_{2m-1}^{(\pm 1)} \end{pmatrix}$$

$$B^{(-1)} \equiv \begin{pmatrix} b_1^{(-1)} & & & & \\ b_2^{(-1)} & b_3^{(-1)} & & & \\ & \dots & \dots & & \\ & & & b_{2m-2}^{(-1)} & b_{2m-1}^{(-1)} \end{pmatrix}$$

30

$b_{2k-1}^{(n)} \equiv \xi_k^{(n)} \sqrt{q_k^{(n)}}, b_{2k}^{(n)} \equiv \eta_k^{(n)} \sqrt{e_k^{(n)}}$

40

$(\xi_k^{(n)})^2 = 1, (\eta_k^{(n)})^2 = 1$

$\xi_k^{(1)} \eta_k^{(1)} = \xi_k^{(0)} \eta_k^{(0)}, \xi_{k+1}^{(-1)} \eta_k^{(-1)} = \xi_k^{(0)} \eta_k^{(0)}$

となる。すなわち、 $\{q_k^{(\pm 1)}, e_k^{(\pm 1)}\}$ を計算する。

(2) コレスキー分解より $(B^{(0)})^T B^{(0)} - \lambda_k I$ のツイステッド分解

【0044】

【数10】

$N_{k0} D_{k0} N_{k0}^T = (B^{(0)})^T B^{(0)} - \lambda_k I$

を形成する。ただし

50

【 0 0 4 5 】

【数 1 1】

$$N_k \equiv \begin{pmatrix} 1 & & & & & \\ L_1^{(1)} & 1 & & & & \\ & & \ddots & & & \\ & & & L_{k-1}^{(1)} & 1 & U_k^{(-1)} \\ & & & & 1 & \ddots \\ & & & & & \ddots & U_{m-1}^{(-1)} \\ & & & & & & & 1 \end{pmatrix}$$

10

$$D_k \equiv \text{diag}(q_1^{(1)}, \dots, q_{k-1}^{(1)}, 1, q_k^{(-1)}, \dots, q_m^{(-1)}),$$

$$L_k^{(1)} \equiv \sqrt{e_k^{(1)}/q_k^{(1)}}, \quad U_k^{(-1)} \equiv \sqrt{e_k^{(-1)}/q_{k+1}^{(-1)}}$$

とし、 $q_k^{(1)} + q_k^{(-1)} - (e_{k-1}^{(0)} + q_k^{(0)} - \dots)$ が最小になる k を k_0 とする。

(3) 簡約化された連立 1 次方程式

【 0 0 4 6 】

【数 1 2】

20

$$N_{k_0}^\top v_j = e_{k_0}, \quad e_{k_0} \equiv \underbrace{(0, \dots, 0, 1, 0, \dots, 0)^\top}_{(k_0-1)\text{個}}$$

を解く。

【 0 0 4 7 】

図 2 は、Parlett らの手法で最も核となる (1) の処理手順を示す。(1) で求めたいのは、 $\{q_k^{(n)}, e_k^{(n)}\}$ 、 $n = \pm 1$ であるが、そのための変換として stationary qd with shift (stqds) 変換

$$q_k^{(1)} + e_{k-1}^{(1)} = q_k^{(0)} + e_{k-1}^{(0)} - \dots_j, \quad k = 1, 2, \dots, m,$$

$$q_k^{(1)} e_k^{(1)} = q_k^{(0)} e_k^{(0)}, \quad k = 1, 2, \dots, m - 1,$$

$$e_0^{(0)} = 0, \quad e_0^{(1)} = 0$$

30

および reverse-time progressive qd with shift (rpqds) 変換

$$q_k^{(-1)} + e_k^{(-1)} = q_k^{(0)} + e_{k-1}^{(0)} - \dots_j, \quad k = 1, 2, \dots, m,$$

$$q_{k+1}^{(-1)} e_k^{(-1)} = q_k^{(0)} e_k^{(0)}, \quad k = 1, 2, \dots, m - 1,$$

$$e_0^{(0)} = 0, \quad e_m^{(-1)} = 0$$

が採用されている。

【 0 0 4 8 】

40

固有値 λ_j が既知ならば反復的な計算は不要なため計算量が QR アルゴリズムに比べて圧倒的に少なく抑えられるが、常に数値安定かつ精度のよい方法とはいえない。stqds 変換と rpqds 変換ともに減算による桁落ちが発生する可能性がある。例えば、stqds 変換において $q_k^{(0)} + e_{k-1}^{(0)} - \dots_j \sim e_{k-1}^{(1)}$ ならば、 $q_k^{(1)} (= q_k^{(0)} + e_{k-1}^{(0)} - \dots_j - e_{k-1}^{(1)})$ を求める際、倍精度計算であっても $q_k^{(1)}$ の有効桁はわずか 1 桁になることもある。その場合、 $q_k^{(0)} e_k^{(0)} / q_k^{(1)}$ を計算すると誤差が生じる、つまり、 $e_k^{(1)}$ が精度よく計算できないことになる。また、 $q_{k+1}^{(-1)}$ を求めるのに $e_k^{(1)}$ が必要、 $e_k^{(1)}$ を求めるのに $q_k^{(1)}$ が必要と逐次的な計算が要求されるので、1 個所で発生した桁落ちによる誤差が波及し、さらなる誤差増大の可能性も秘めている。その結果、理論上は q_k

50

(1) 0であるが誤差蓄積により $q_k^{(1)} = 0$ となり、 $q_k^{(0)} e_k^{(0)} / q_k^{(1)}$ の計算においてオーバーフローが起こるといった数値不安定な状況も想定される。 $B^{(0)} = B$ の成分 $\{b_{2k-1}^{(0)}, b_{2k}^{(0)}\}$ が与えられる、すなわち $\{q_k^{(0)}, e_k^{(0)}\}$ が与えられると、 j および $e_{k-1}^{(1)}$ が一意的に決まるので、この状況は避けることはできない。 $r p q d s$ 変換も同様の性質を持つため、実用的なレベルにまで達したとはいいがたい。LAPACKにおいてFORTRANルーチンDSTEGRとして改良版が公開されているものの欠点は完全に解決されてはいない。

【0049】

次に、本発明による対角化のルーチンを説明する。本発明によるルーチンでは、 $s t q d s$ 変換および $r p q d s$ 変換にかわる新たな変換を採用している。本発明では、行列 $B^T B - j^2 I$ に対するツイスティッド分解を行う。正確には、上記(1)を以下の(1a)、(1b)、(1c)の3つの工程にわけること数値安定なコレスキー分解を実現している。

10

【0050】

図3~図8は、本発明によるミウラ型逆変換、 $s d L V v s$ 変換、 $r d L V v s$ 変換およびミウラ型変換を実行する処理手順を示す。

(1a) ミウラ型逆変換

【0051】

【数13】

20

$$\{q_k^{(0)}, e_k^{(0)}\} \mapsto \{u_{2k-1}^{(0)}, u_{2k}^{(0)}\}$$

を行う。

【0052】

$$\begin{aligned} u_{2k-1}^{(0)} &= t_k^{(0)}, k = 1, 2, \dots, m, \\ u_{2k}^{(0)} &= e_k^{(0)} / t_k^{(0)}, k = 1, 2, \dots, m-1, \\ t_k^{(0)} &= q_k^{(0)} / (q_k^{(0)} + u_{2k-2}^{(0)}) - 1, \quad (0) \quad 1 / (0) \end{aligned}$$

ただし、 (0) は任意に選ぶことができる。

(1b) stationary 不等間隔差分 Lotka - Volterra ($s d L V v s$) 変換

30

$$\begin{aligned} u_k^{(1)} &= u_k^{(0)} * (1 + (0) u_{k-1}^{(0)}) / (1 + (1) u_{k-1}^{(1)}), \\ k &= 1, 2, \dots, 2m-1, \end{aligned}$$

によって

【0053】

【数14】

$$u_k^{(0)} \mapsto u_k^{(1)}$$

reverse - time 不等間隔差分 Lotka - Volterra ($r d L V v s$) 変換

40

$$\begin{aligned} u_k^{(-1)} &= u_k^{(0)} * (1 + (0) u_{k-1}^{(0)}) / (1 + (-1) u_{k+1}^{(-1)}), \\ k &= 1, 2, \dots, 2m-1, \end{aligned}$$

によって

【0054】

【数15】

$$u_k^{(0)} \mapsto u_k^{(-1)}$$

を実行する。ただし、 $u_k^{(n)} = 0, u_{2m}^{(0)} = 0$ とし、 $j = 1 / (0) - 1$

50

$\gamma^{(n)}$, $n = \pm 1$ を満たすように $\beta^{(n)}$, $n = \pm 1$ を設定する。

(1c) ミウラ型変換

【0055】

【数16】

$$\{u_{2k-1}^{(n)}, u_{2k}^{(n)}\} \mapsto \{q_k^{(n)}, e_k^{(n)}\}, n = \pm 1$$

$$q_k^{(n)} = \beta^{(n)} * (1 + \gamma^{(n)} u_{2k-2}^{(n)}) * (1 + \gamma^{(n)} u_{2k-1}^{(n)}),$$

$$k = 1, 2, \dots, m,$$

$$e_k^{(n)} = \beta^{(n)} * u_{2k-1}^{(n)} * u_{2k}^{(n)}$$

$$k = 1, 2, \dots, m-1,$$

10

を行う。

【0056】

qd型変換では見られない離散Lotka-Volterra型変換の大きな特徴は、任意パラメータを持つことである。すなわち、 $\beta_j = 1/\gamma^{(0)} - 1/\gamma^{(\pm 1)}$ を満たす範囲で $\beta^{(n)}$ の値を任意に設定できる。 $\beta^{(n)}$ を変動させると補助変数 $u_k^{(n)}$ の値も変化するが、桁落ちによる誤差や数値不安定が発生するかどうかは事前に判定できる。この判定は、if文によって実装されてもよい。この場合は、 $\beta^{(n)}$ を再設定後に再度計算すればよい。また、 $u_k^{(\pm 1)}$ が求めればミウラ変換によって $q_k^{(\pm 1)}$ および $e_k^{(\pm 1)}$ が独立に計算されるので、誤差が伝播しないという性質を持つ。なお、ミウラ型逆変換をミウラ変換、ミウラ型変換を逆ミウラ変換、sdLVvs変換をstLVv変換、rdLVv変換をrLVv変換と呼び変えても問題はない。

20

【0057】

入力および出力の変数の対応は、Parlettの手法と同じである。メモリ消費を抑えるために、補助変数 $u_k^{(n)}$ のための配列は必ずしも用意する必要はない。一方、 $1 + \gamma^{(0)} u^{(0)}$ のためのメモリ領域を確保し、(1a)~(1c)のステップにまたがってこの値を利用することで、メモリ消費を抑え、計算量を低減することができる。これにより、誤差も低減される。このように、1.3で説明する方法を利用して誤差を低減すること等、さらなる工夫を行い得る。

【0058】

上記の方法で $B^T B - \beta_j I$ のコレスキー分解を求めた後は、Parlettらと同じ方法を用いて特異ベクトル v_j を計算する。そうすると、 $B^T B$ の固有値分解 $B^T B = V_B^T B^T B V_B$ が得られる。ただし V_B の列ベクトルが v_j である。 $U_B = B V_B^{-1}$ より U_B を求めると、 B の特異値分解 $B = U_B V_B^T$ も得られる。

30

【0059】

1.2.高精度特異値算出dLVsルーチン

本発明によるdLVsルーチンは、基本的な枠組みではDLASQルーチンと同じであり、 $B^T B$ の固有値(あるいはBの特異値)のみが得られ $B^T B$ の固有ベクトル(あるいはBの特異ベクトル)は求まらない。ただ、ルーチンの核となる部分に使用される計算法はDLASQと異なる。

40

【0060】

まず、DLASQルーチンによる固有値の計算方法を説明する。qd法は固有ベクトルが計算できないが、1ステップは上2重対角行列の要素を更新するのみなので高速に特異値を求めることができる。かつ、計算量が少ないため丸め誤差の蓄積が抑えられ高精度計算が期待できる。ここで、 $Y^{(n)}$ を対称な3重対角行列

【0061】

【数17】

$$Y^{(n)} = \begin{pmatrix} q_1^{(n)2} & q_1^{(n)}e_1^{(n)} & & & 0 \\ q_1^{(n)}e_1^{(n)} & q_2^{(n)2} + e_1^{(n)2} & \dots & & \\ & \dots & \dots & \dots & \\ 0 & & & q_{m-1}^{(n)}e_{m-1}^{(n)} & \\ & & & q_{m-1}^{(n)}e_{m-1}^{(n)} & q_m^{(n)2} + e_{m-1}^{(n)2} \end{pmatrix}$$

とする。また、変換 $Y^{(n)} \rightarrow Y^{(n+1)}$ が qd 法の漸化式

$$\begin{aligned} q_k^{(n+1)} &= q_k^{(n)} + e_k^{(n)} - e_{k-1}^{(n+1)}, \quad k = 1, 2, \dots, m, \\ e_k^{(n+1)} &= e_k^{(n)} * q_{k+1}^{(n)} / q_k^{(n+1)}, \quad k = 1, 2, \dots, m-1, \\ e_0^{(n)} &= 0, \quad e_m^{(n)} = 0, \quad n = 0, 1, \dots \end{aligned} \quad 10$$

によって実現されるとする。そのとき、適切な $R^{(n)}$ が存在して $Y^{(n+1)} = R^{(n)} Y^{(n)} (R^{(n)})^{-1}$ となることを Rutishauser は発見している。これは、 $Y^{(n+1)}$ と $Y^{(n)}$ の固有値が同じであることを意味し、すなわち、上記の漸化式によって固有値保存変形がなされる。この変形を繰り返せば非対角成分が 0 に近づき、 $Y^{(n)}$ が対角化されることも証明されている。よって $Y^{(0)} = B^T B$ とすると $\lim_n Y^{(n)} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_m)$ となる。ただし、 λ_k は $B^T B$ の固有値である。さらに、 λ_k の正の平方根をとると B の特異値 σ_k が得られる。 20

【0062】

LAPCAK の特異値計算ルーチン DLASQ では、qd 法の differential 型が採用されている。これは dqd 法と呼ばれる。1 ステップは

【0063】

【数18】

```

d := q1(n)
for k := 1, m-1
  qk(n+1) := d + ek(n)
  ek(n+1) := ek(n) * (qk+1(n) / qk(n+1))
  d := d * (qk+1(n) / qk(n+1))
end for
qm(n+1) := d

```

30

で与えられ、変換 $Y^{(n)} \rightarrow Y^{(n+1)}$ が実現される。differential 型は減算を含まないため、qd 法では起こりうる桁落ち誤差の心配もない。さらに高速版として原点シフト付き dqd (dqds) 法も組み込まれている。高速化に伴う計算量の減少で丸め誤差の発生も抑えることができる。ただし、収束するかどうかの保証はない。原点シフト版の 1 ステップは 40

【0064】

【数 1 9】

```

d := q1(n) - s
for k := 1, m - 1
  qk(n+1) := d + ek(n)
  ek(n+1) := ek(n) * (qk+1(n) / qk(n+1))
  d := d * (qk+1(n) / qk(n+1)) - s
end for
qm(n+1) := d

```

10

与えられる。s は原点シフト量であり、G e r s g o r i n 型境界条件により見積もられた $Y^{(n)}$ の最小固有値とする。よって、D L A S Q ルーチンのメインループの核となる部分では最小固有値の見積もりを行い、 $s > \epsilon$ (ϵ : 非常に小さいとみなせる正の数) ならば d q d s 法が、そうでなければ d q d 法が用いられる。メインループの他の部分 (行列の分割・減次) はほとんど D B D S Q R ルーチンと変わらない。すなわち、D B D S Q R ルーチンに含まれる Q R 法を d q d 法に、原点シフト Q R 法を d q d s 法に置き換えたものが D L A S Q ルーチンとなる。(Q R 法と d q d 法、原点シフト付き Q R 法と d q d s 法で扱う変数が異なるので少しだけ判定式が異なる部分もあるが本質的に同じである。)

次に本発明による d L V s ルーチンを説明する。d L V s ルーチンでは d q d 法かわりに d L V 法を、d q d s 法かわりに d L V s 法を採用する。ここで、d L V s ルーチンは D L A S Q ルーチンとは違って収束することが保証されている。その 1 ステップは以下で与えられる。

20

【 0 0 6 5 】

【数 2 0】

```

u1(n) := w1(n);
for k = 2, 2m - 1
  uk(n) := wk(n) / (1 + δ(n) * uk-1(n));   vk-1(n) := uk-1(n) * (1 + δ(n) * uk(n));
end for
v2m-1(n) := u2m-1(n)
(原点シフト s の決定)
if (s > ε) w1(n+1) := v1(n) - s;
  for k = 2, m
    w2k-2(n+1) := v2k-2(n) * v2k-3(n) / w2k-3(n+1);   w2k-1(n+1) := v2k-1(n) + v2k-2(n) - w2k-2(n+1) - s;
  end for
end if
else
  for k = 1, 2m - 1
    wk(n+1) := vk(n);
  end for
end else

```

30

40

D L A S Q ルーチン同様、メインループの核の部分では最小特異値の見積もりによって原点シフト量 s を決定し、d L V 法を使うか d L V s 法を使うかを決定する。ただし、補助変数 $v_k^{(n)}$ を求めたあとで s を決定する点は D L A S Q ルーチンと異なる。

【 0 0 6 6 】

1. 3. 本発明の方法による誤差の低減

人間が理想的な状況、すなわち、無限桁の計算をいくらでもできるとすると、P a r l

50

e t tらの手法、本発明による特異ベクトル計算ルーチン(1.1参照)のどちらを使っても問題ない。しかしながら、コンピュータで計算を行う場合は注意が必要である。有限桁の計算しか行えないコンピュータ上では、数学的に正しい計算法を使ったとしても必ずしも正しい結果が得られる訳ではない。そればかりかいつまでたっても計算が終了しないといった思わぬ数値的な問題が発生する場合もある。

【0067】

コンピュータ計算による誤差には、丸め誤差および桁落ちによる誤差などが知られている。丸め誤差単独では、高々有効桁の最後の桁が真値と比べて異なる程度で大きな誤差にはならない。また、指数部が異なる2つの実数の加算、乗算、除算の少なくとも1つの演算を行えばやはり丸め誤差が生じるが、それ以上の誤差は発生しない。さらに、このような丸め誤差を発生するような操作が繰り返されても、丸めモードがnear(四捨五入)ならば、一方的に切り上げられたりあるいは切り捨てられたりして誤差が極端に蓄積することは少ない。よって、多くの数値計算法は加算、乗算、除算の少なくとも1つの演算によって発生する丸め誤差を特別注意することは少なく、d L V sルーチンによる特異値計算でも結果的に丸め誤差は一様に増大しない。

10

【0068】

問題となるのは、同符号の実数の減算および異符号の実数の加算により生じる、桁落ちである。桁落ちによる誤差で値が0となった後、その値による除算を行うと、0が分母にくるような不定形となり計算不可能となる。こうなるといつまでたっても計算が終了しない。減算 除算と計算する部分がParlettの手法、本発明による方法とも存在するので、桁落ち誤差には十分に注意する必要がある。

20

【0069】

本発明による計算方法では、上述の桁落ちによる誤差を含んでいるかどうかは減算によって得られた値が小さいかどうかで判断できる。Parlettの手法の場合、図2のD O文中の $q_1[k]$ 、 $q_2[k]$ をチェックすればよい。ところが、Parlettの手法は桁落ち誤差を含むことが分かったとしても、それを回避することはできない。なぜならば、初期値として $\{q_0[k], e_0[k]\}$ が与えられると、 α は一意的に決定され、 $\{q_1[k], e_1[k]\}$ および $\{q_2[k], e_2[k]\}$ も一意的に導出されるためである。すなわち、任意パラメータを持たない自由度のない計算法であるためである。

【0070】

それに対して本発明による方法は自由に設定できるパラメータ $\alpha_1 (= 1 / \alpha^{(0)})$ をもつため、補助変数 $u_k^{(n)}$ の値を様々に変化させることができる(図9参照)。すなわち、様々な経路で $\{q_1[k], e_1[k]\}$ および $\{q_2[k], e_2[k]\}$ を計算することができる。よって、桁落ちが発生する場合も回避できる。図6~図8の条件判定によって桁落ちの影響をチェックし、減算によって得られた値の絶対値が小さな数より大きいという条件が満たされなければ、パラメータ α_1 の設定に戻るというものである。この処理は、条件が満たされるまで繰り返される。なお、精度よりも高速性を重視する場合は、数回条件が満たさなければ($q_1[k] = 0$ あるいは $q_2[k] = 0$ ならば)、例外処理(文献PDF参照)を行ってもよい。

30

【0071】

2. 本発明による特異値分解装置

特異値分解を実行する装置を説明する。特異値分解を実行する装置は、例えば、図1に示される各ステップの機能を有する装置である。特異値分解を実行する装置の1つの実施形態は、特異値分解プログラムを実行するコンピュータである。そのプログラムは、例えば、図1に示される各ステップをCPUに実行させるプログラムである。

40

【0072】

図10は、図1の処理をコンピュータプログラムとして実行するコンピュータ10を示す。コンピュータ10は、CPU1001と、メモリ1002と、入力インターフェース部1003と、出力インターフェース部1004と、バス1005とを含む。

【0073】

50

CPU1001は、プログラムを実行する。例えば、そのプログラムは、図1の処理を実行するプログラムである。そのプログラムおよびそのプログラムの実行に必要なデータは、例えば、メモリ1002に格納される。そのプログラムは、任意の様態でメモリ1002に含まれ得る。例えば、メモリ1002が書き換え可能なメモリである場合には、コンピュータ1002の外部からプログラムをローディングすることにより、そのプログラムをメモリ1002に格納してもよい。あるいは、メモリ1002が読み出し専用メモリである場合には、メモリ1002に焼き付ける形式でそのプログラムをメモリ1002に格納してもよい。

【0074】

さらに、入力インターフェース部1003は、特異値分解を実行する対象である行列Aを外部から受け取るためのインターフェースとして機能する。出力インターフェース部1004は、計算結果を出力するためのインターフェースとして機能する。バス1005は、コンピュータ10内の構成要素1001～1004を相互に接続するために使用される。

【0075】

3. 本発明による特異値分解方法の応用

本発明による特異値分解方法は、様々な分野に応用可能である。以下に、2次元画像から3次元画像へ復元する画像処理への応用例、ならびに、文書検索への応用例を示す。しかし、これら2つの応用例は例示に過ぎず、本発明による特異値分解方法の応用は、これら2つの応用例に限定されない。本発明による特異値分解方法の応用は、特異値分解を利用するあらゆる分野に適用可能である。

【0076】

3.1. 2次元から3次元へ復元する画像処理への応用

図11を参照する。図11は、本発明による特異値分解方法を利用した物体の複数の2次元画像を3次元画像へ復元する画像処理の1つの実施形態を説明する。

【0077】

複数の2次元画像から3次元復元を行うために必要となるステップは、2次元画像から特徴点を抽出するステップと、特徴点データより形状（元の物体の特徴点の3次元座標データ）および回転（3次元データから特徴点データへの変換）に関するデータを計算するステップと、形状および回転のデータより可視化を行うステップとを包含する。

【0078】

ステップ1102において、2次元画像 j （ $j = 1, \dots, m$ 、 m は3以上の整数）から特徴点 i （ $i = 1, \dots, n$ 、 n は2以上の整数）の座標（ x_i^j, y_i^j ）を抽出する。取り扱う2次元画像は、弱中心射影画像であることが好ましい。このとき

【0079】

【数21】

$$\begin{bmatrix} x_i^j \\ y_i^j \end{bmatrix} = s_j \begin{bmatrix} \mathbf{r}_{1,j}^T \\ \mathbf{r}_{2,j}^T \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix}$$

が成り立つ。ここで、 s_j は物体のスケールに相対する j 番目の画像のスケール、 $\mathbf{r}_{1,j}$ 、 $\mathbf{r}_{2,j}$ はそれぞれ物体座標系に相対する j 番目のカメラ座標系の回転行列の1番目と2番目の行ベクトル、 $[X_i, Y_i, Z_i]^T$ は i 番目の点の3次元座標である。物体のスケールは1番目の画像のスケールと同じにし（ $s_1 = 1$ ）、物体の座標系の姿勢は1番目の画像のカメラ座標系と同じにする（ $\mathbf{r}_{1,1} = [1, 0, 0]^T$ 、 $\mathbf{r}_{2,1} = [0, 1, 0]^T$ ）。 m 枚全ての画像における n 個全ての点の座標を行列Dの要素として並べ

10

20

30

40

50

ると、

$$D = MS$$

が得られる。

ただし、

【 0 0 8 0 】

【 数 2 2 】

$$D = \begin{bmatrix} x_1^1 & \cdots & x_i^1 & \cdots & x_n^1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^j & \cdots & x_i^j & \cdots & x_n^j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^m & \cdots & x_i^m & \cdots & x_n^m \\ y_1^1 & \cdots & y_i^1 & \cdots & y_n^1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y_1^j & \cdots & y_i^j & \cdots & y_n^j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y_1^m & \cdots & y_i^m & \cdots & y_n^m \end{bmatrix}, \quad M = \begin{bmatrix} s_1 \mathbf{r}_{1,1}^T \\ \cdots \\ s_j \mathbf{r}_{1,j}^T \\ \cdots \\ s_m \mathbf{r}_{1,m}^T \\ s_1 \mathbf{r}_{2,1}^T \\ \cdots \\ s_j \mathbf{r}_{2,j}^T \\ \cdots \\ s_m \mathbf{r}_{2,m}^T \end{bmatrix} \quad \begin{matrix} 10 \\ 20 \\ 30 \end{matrix}$$

$$S = \begin{bmatrix} X_1 & \cdots & X_i & \cdots & X_n \\ Y_1 & \cdots & Y_i & \cdots & Y_n \\ Z_1 & \cdots & Z_1 & \cdots & Z_n \end{bmatrix} \quad 30$$

である。MとSの形から分かるように、Dのランクは3である。いま、ステップ1102よりDが与えられている。以下、回転に関するデータMおよび形状Sを求める。

【 0 0 8 1 】

そこで、行列Dの特異値分解

$$D = U \quad V^T$$

を考える。ここで、 λ は特異値を大小順に対角線上に並べたもので、UおよびVはそれぞれ左および右直交行列である。ステップ1104では、行列Dの特異値分解を計算するために、行列Dを上2重対角化して上2重対角化行列Bを得る。ステップ1106において、行列B、すなわちDの特異値を計算する。ステップ1106では、本発明によるdLVSルーチン、DLASQルーチン、その他の特異値計算ルーチン、あるいはその組み合わせを利用し得る。このとき、画像のデジタル誤差のため、ゼロでない特異値は3つ以上出てくる。しかし、4番目以降の特異値はノイズによるもので、最初の3つの特異値と比べて格段に小さい。

【 0 0 8 2 】

そこで、ステップ1108において、最初の3つの特異値に対して特異ベクトルを計算する。ステップ1108は、図1のステップ110～120を利用し得る。採用する3個

40

50

のベクトルをまとめると、次式となる。

【0083】

$$D' = L' \quad R'^T = M' S'$$

ただし、 $M' = L' (\quad)^{1/2}$ 、 $S' = (\quad)^{1/2} R'^T$ 、 D' は $D - D'$ を最小にするランク3の行列である。

【0084】

次に、 D' からMおよびSを求めたいが、その組合せは唯一ではない。なぜなら、任意の正則行列Cが

$$D' = (M' C) (C^{-1} S')$$

を満たすからである。そこで、上式におけるCを $M = M' C$ を満たすように決める。Cは下記の式を満たす。

【0085】

【数23】

$$MM^T = M'CC^TM'^T = \begin{bmatrix} 1 & \cdots & \cdots & 0 & \cdots & \cdots \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \cdots & \cdots & s_m^2 & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & 1 & \cdots & \cdots \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \cdots & \cdots & 0 & \cdots & \cdots & s_m^2 \end{bmatrix}$$

10

20

$E = CC^T$ とすると、上式からEの6つの要素に関する $2m + 1$ 個の線形方程式が得られる。 $m = 3$ であるので、Eの要素を一意に決めることができる。ステップ1110において、行列Eを求める。

【0086】

次に、ステップ1112において、EからCを求める。Cの自由度(9)はEの自由度(6)より多い。そこで、条件 $r_{1j} = [1, 0, 0]^T$ 、 $r_{2j} = [0, 1, 0]^T$ を加えれば、Cを決めることができる。このとき2つの解(Necker Reversal)が出る。

30

【0087】

次に、ステップ1114において、 $M = M' C$ および $S = C^{-1} S'$ より、回転に関するデータMおよび形状Sが決まる。

【0088】

DBDSQRルーチンでは上記のようにU、V(n個のベクトル)を求めた後、3個のベクトル(n-3個は不要)を採用する。ここでは、原則的にはn個のベクトルを求めるが、例外的にn個より少ない数のベクトルを求めてもよい。この実施形態では3個のベクトルだけ求めればよい。すなわち、計算コストを削減することができる。

40

【0089】

3.2. 本発明による特異値分解方法を利用した文書検索方法

文書中からその文書の内容に関連する索引語を抽出し、索引語の重みを計算する処理の後、ベクトル空間モデルでは、この索引語の重みを要素とするベクトルで文書を表現する。いま、検索対象となる文書を D_1, D_2, \dots, D_n とし、これら文書集合全体を通して全部でm個の索引語 w_1, w_2, \dots, w_m があるとする。このとき、文書 D_j は、次のようなベクトルで表現されることになる。これを文書ベクトルと呼ぶ。

【0090】

【数 2 4】

$$\mathbf{d}_j = \begin{bmatrix} d_{1j} \\ d_{2j} \\ \vdots \\ d_{mj} \end{bmatrix}$$

ここで、 d_{ij} は索引語 w_i の文書 D_j における重みである。また、文書集合全体は、次のような $m \times n$ 行列 D によって表現することができる。 10

【0 0 9 1】

【数 2 5】

$$D = [\mathbf{d}_1 \ \mathbf{d}_2 \ \cdots \ \mathbf{d}_n] = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{bmatrix} \quad 20$$

行列 D を索引語文書行列と呼ぶ。索引語文書行列の各列は文書に関する情報を表している文書ベクトルであるが、同様に、索引語文書行列の各行は索引語に関する情報を表しているベクトルであり、これを索引語ベクトルと呼ぶ。検索質問も、文書と同様に、索引語の重みを要素とするベクトルで表現することができる。検索質問文に含まれる索引語 w_i の重みを q_i とすると、検索質問ベクトル \mathbf{q} は次のように表されることになる。

【0 0 9 2】

【数 2 6】

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_m \end{bmatrix} \quad 30$$

実際の文書検索においては、与えられた検索質問文と類似した文書を見つけ出す必要があるが、検索質問ベクトル \mathbf{q} と各文書ベクトル \mathbf{d}_j の間の類似度を計算することにより行う。ベクトル間の類似度の定義としてはさまざまなものが考えられるが、文書検索においてよく用いられているものはコサイン尺度（2つのベクトルのなす角度）または内積である。 40

【0 0 9 3】

【数 2 7】

- コサイン尺度

$$\cos(\mathbf{d}_j, \mathbf{q}) = \frac{\mathbf{d}_j \cdot \mathbf{q}}{\|\mathbf{d}_j\| \|\mathbf{q}\|} = \frac{\sum_{i=1}^m d_{ij} q_i}{\sqrt{\sum_{i=1}^m d_{ij}^2} \sqrt{\sum_{i=1}^m q_i^2}}$$

- 内積

$$\mathbf{d}_j \cdot \mathbf{q} = \sum_{i=1}^m d_{ij} q_i$$

なお、ベクトルの長さが1に正規化（コサイン正規化）されている場合には、コサイン尺度と内積とは一致する。

【0094】

図12を参照する。図12は、本発明による特異値分解装置を利用した、文書検索方法の1つの実施形態を説明する。

【0095】

ステップ1202において、質問ベクトル q を受け取る。次に、ステップ1204では、行列 D の特異値分解を計算するために、行列 D を上2重対角化して上2重対角化行列 B を得る。ステップ1206では、行列 B 、すなわち行列 D の特異値を求める。ステップ1206では、本発明による dLV ルーチン、 $DLASQ$ ルーチン、その他の特異値計算ルーチン、あるいはその組み合わせを利用し得る。

10

【0096】

次に、 D の近似行列 D_k を使った検索を考える。ベクトル空間モデルでは、検索質問ベクトル q と索引語文書行列 D 中の各文書ベクトル d_j の間の類似度を計算することにより検索を行うが、ここでは D の代わりに D_k を使う。ベクトル空間モデルでは、文書ベクトルの次元数は索引語の総数と等しい。したがって、検索対象となる文書の数が増えるに従い、文書ベクトルの次元数も増加する傾向にある。しかし、次元数が増加してくると、コンピュータのメモリによる制限や検索時間の増大などの問題が生じてくるばかりでなく、文書中に含まれる不必要な索引語がノイズ的な影響を及ぼし、検索精度を低下させてしまうという現象も起こってくる。潜在的意味インデキシング（latent semantic indexing; LSI）は、高次元の空間にある文書ベクトルを低次元の空間へと射影することにより、検索精度の改善を図る技術である。高次元の空間では別々に扱われていた索引語が、低次元の空間では相互に関連を持ったものとして扱われる可能性もあるため、索引語の持つ意味や概念に基づく検索を行うことができる。たとえば、通常のベクトル空間モデルでは“car”という索引語と“automobile”という索引語はまったく別物であり、一方の索引語による質問ではもう片方の索引語を含んだ文書を検索することができない。しかし、低次元の空間ではこれらの意味的に関連した索引語は1つの次元に縮退することが期待できるため、“car”という検索質問によって“car”を含む文書ばかりでなく“automobile”を含む文書をも検索することが可能となる。潜在的意味インデキシングでは、特異値分解により高次元ベクトルの次元削減を行うが、これは基本的に多変量解析における主成分分析と等価である。

20

30

【0097】

行列ステップ1208では、 $k < r$ なる k の値を選択する。 k の値は、予め与えられていてもよいし、計算ごとに選択可能であってもよい。

【0098】

ステップ1210において、ステップ1206で計算した特異値のうち、大きい順に1番目から k 番目の k 個の特異値に対して、 D の特異ベクトルを計算する。すなわち、

$$D_k = U_k V_k^T$$

なる U_k および V_k を計算する。ここで、 U_k は、最初の k 個の左特異ベクトルのみから構成される $m \times k$ 行列であり、 V_k は、最初の k 個の右特異ベクトルのみから構成される $n \times k$ 行列であり、 Λ_k は、最初の k 個の特異値のみから構成される $k \times k$ 対角行列である。ステップ1210は、図1のステップ110～120を利用し得る。

40

【0099】

次に、ステップ1212において、行列 D_k と質問ベクトル q との類似度を計算する。いま、ベクトル e_j を n 次元の単位ベクトルとすると、 D_k の j 番目の文書ベクトルは $D_k e_j$ で表すことができる。文書ベクトル $D_k e_j$ と検索質問ベクトル q との間の類似度計算は、

【0100】

【数 2 8】

$$\begin{aligned}
 \cos(D_k e_j, q) &= \frac{(D_k e_j) \cdot q}{\|D_k e_j\| \|q\|} = \frac{(D_k e_j)^T q}{\|D_k e_j\| \|q\|} \\
 &= \frac{(U_k \Sigma_k V_k^T e_j)^T q}{\|U_k \Sigma_k V_k^T e_j\| \|q\|} = \frac{e_j^T V_k \Sigma_k U_k^T q}{\|\Sigma_k V_k^T e_j\| \|q\|} \\
 &= \frac{(\Sigma_k V_k^T e_j)^T (U_k^T q)}{\|\Sigma_k V_k^T e_j\| \|q\|}
 \end{aligned}$$

10

としてもよいが、別の定義を用いてもよい。上式では、 D_k を U_k , Σ_k , V_k から再構成する必要はなく特異値分解の結果から、直接、類似度を計算できることを示している。数 2 9 の中に現われる $\Sigma_k V_k^T e_j$ は、

$$\Sigma_k V_k^T e_j = U_k^T D_k e_j$$

と書き直すことができる。この式の右辺は、近似行列 D_k における j 番目の文書ベクトルの基底 U_k のもとでの座標（文書の k 次元表現）を表している。同様に、数 2 9 中の $U_k^T q$ は、検索質問ベクトル q の基底 U_k のもとでの座標（検索質問の k 次元表現）である。

20

【0 1 0 1】

ステップ 1 2 1 4 において、ステップ 1 2 1 2 において計算された類似度を基準に、検索結果を出力する。

【0 1 0 2】

DBDSQRルーチンでは r 個のベクトルを求めた後、 k 個のベクトル（ $r - k$ 個は不要）を採用するのに対して、この実施形態では、 k 個のベクトルだけ求めればよい。すなわち、計算コストを削減することができる。

【0 1 0 3】

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ことが理解される。本明細書において引用した特許、特許出願および文献は、その内容自体が具体的に本明細書に記載されているのと同様にその内容が本明細書に対する参考として援用されるべきであることが理解される。

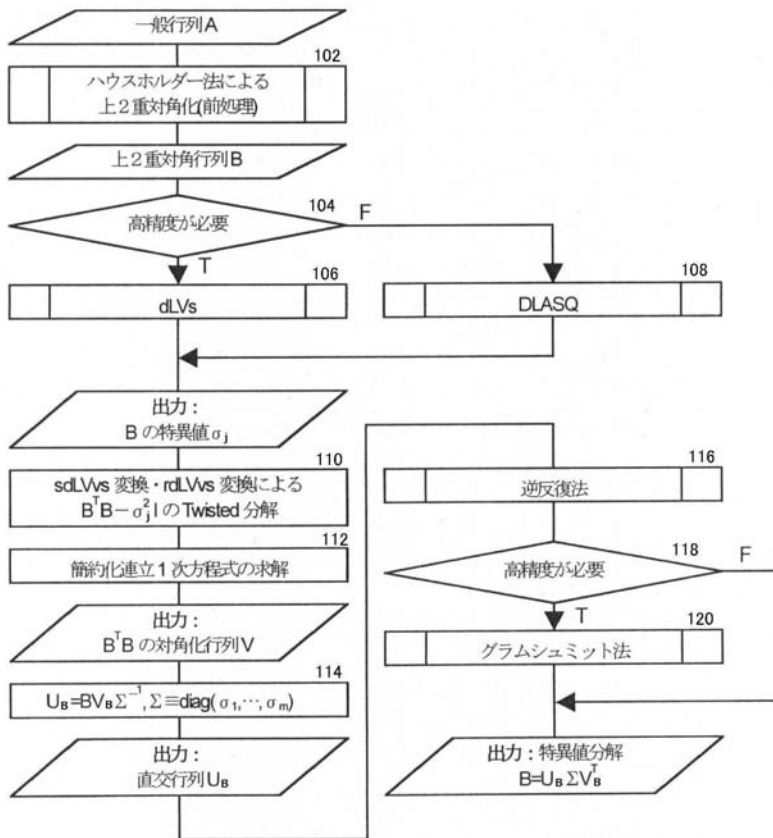
30

【産業上の利用可能性】

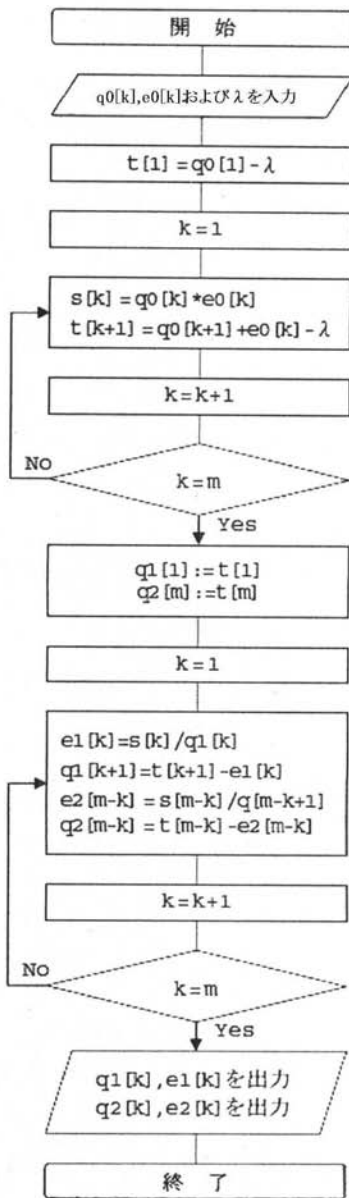
【0 1 0 4】

本発明は、任意の行列 A の特異値分解を高速かつ高精度に実行することを可能にする方法、プログラム、および装置等として有用である。

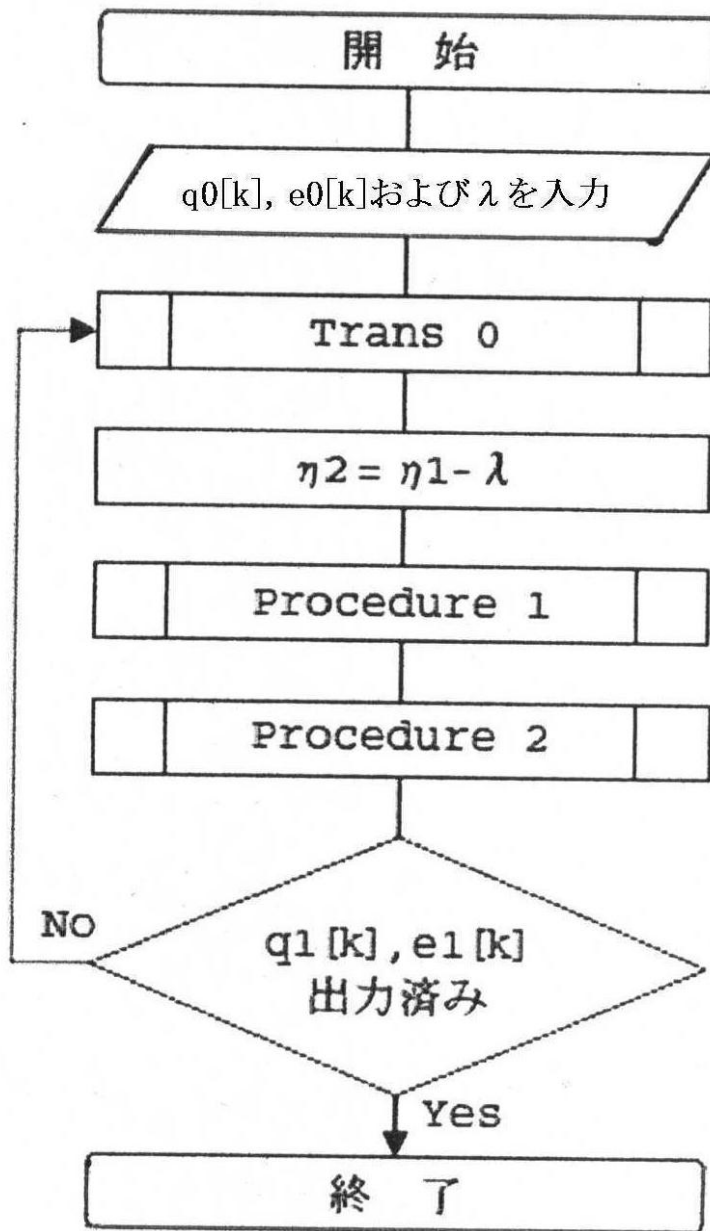
【図1】



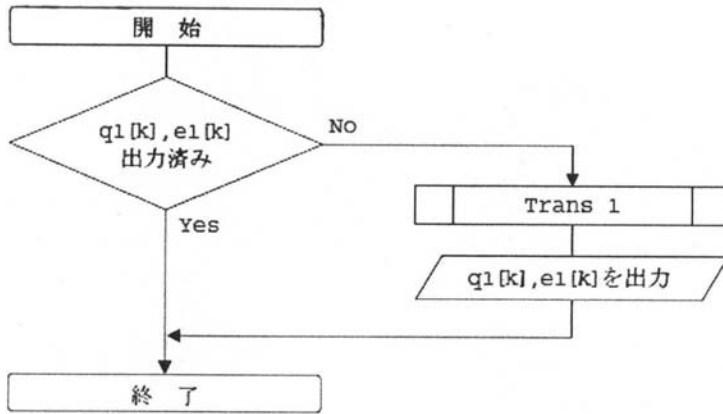
【 図 2 】



【図3】

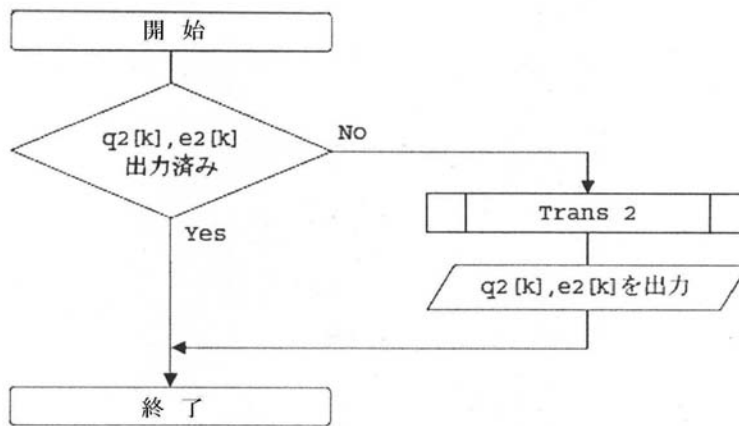


【 図 4 】



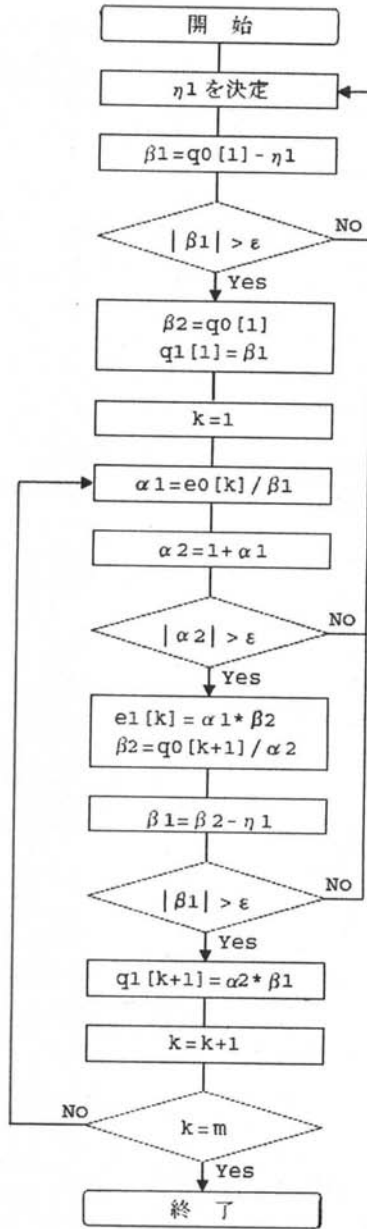
Procedure 1

【 図 5 】



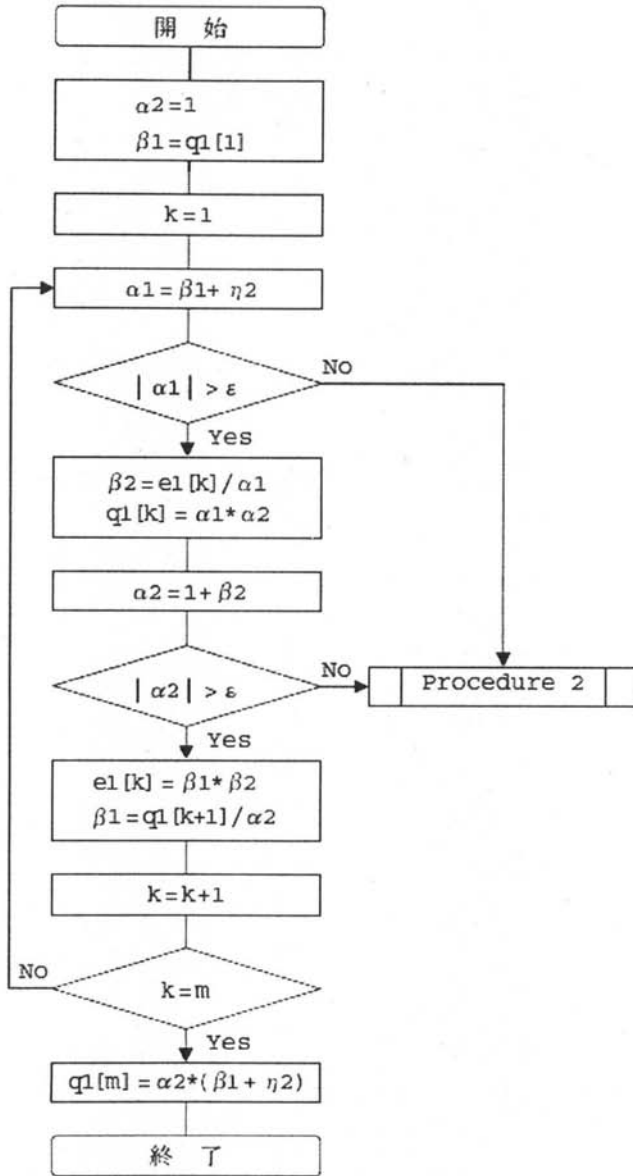
Procedure 2

【図6】



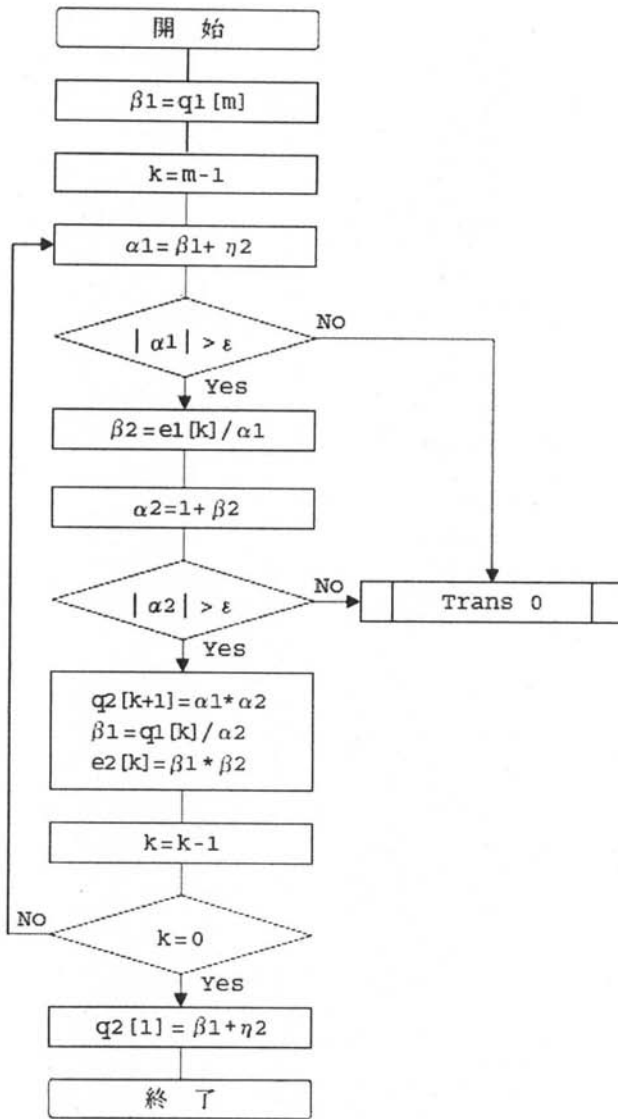
Trans 0

【 図 7 】



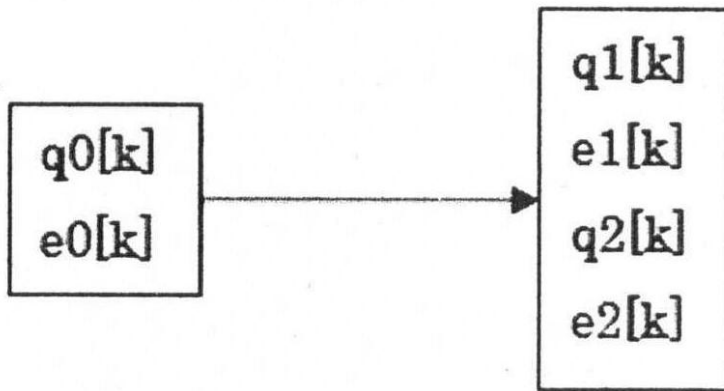
Trans 1

【 図 8 】

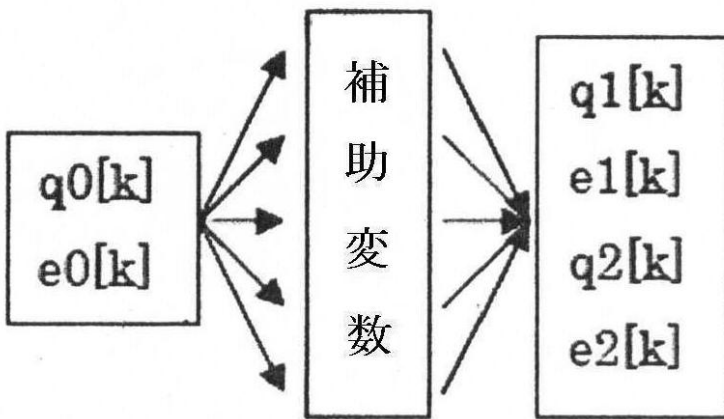


Trans 2

【図9】



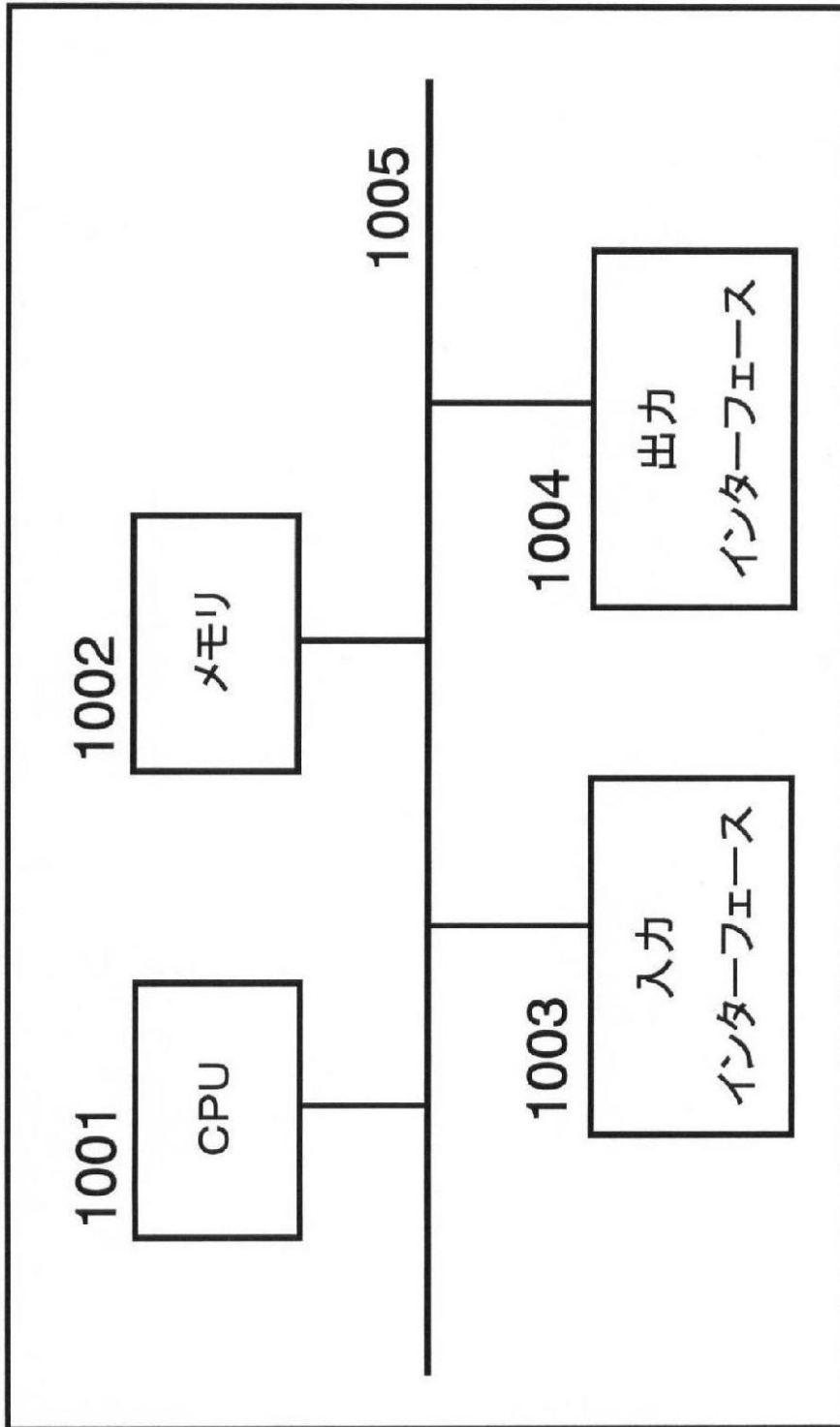
Parlettの手法



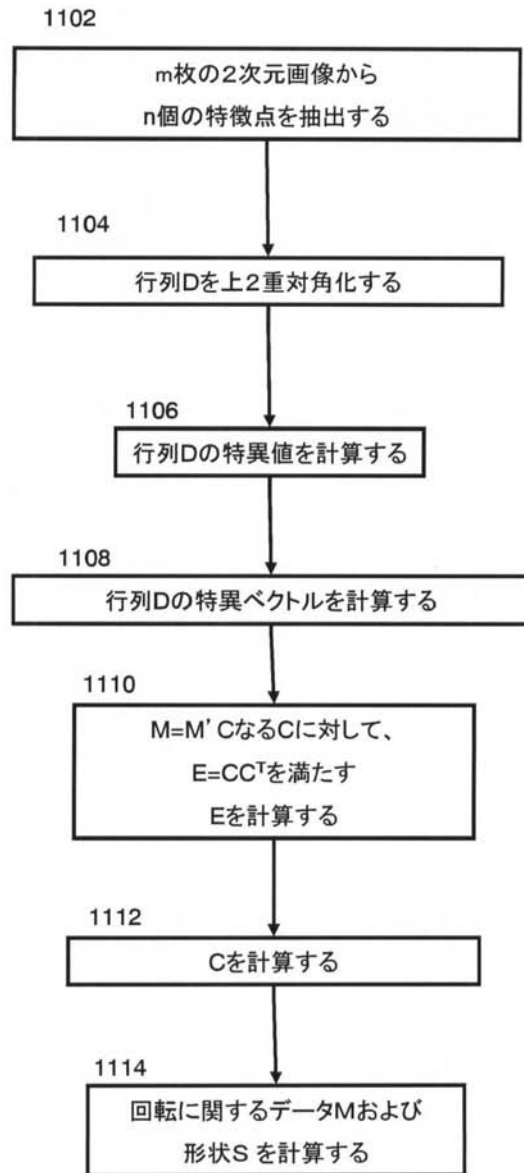
本発明による方法

【図10】

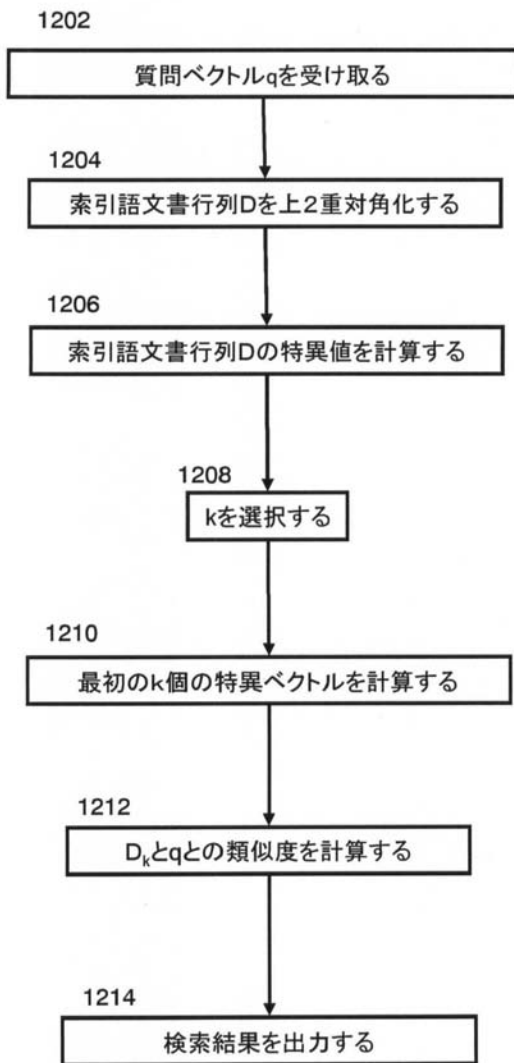
10



【図11】



【図 1 2】



フロントページの続き

(72)発明者 阪野 真也

京都府京都市左京区下鴨北園町69-1北園荘3号

審査官 鳥居 稔

(56)参考文献 特開平10-031747(JP,A)

特開2002-202983(JP,A)

Nakamura, Yoshimasa, A new approach to numerical algorithms in terms of integrable systems, Informatics Research for Development of Knowledge Society Infrastructure, 2004. ICKS 2004. International Conference on , 2004年 3月 1日, 194 - 205

Masashi Iwasaki, Yoshimasa Nakamura, An application of the discrete Lotka-Volterra system with variable step-size to singular value computation, 2004 Inverse Problems 20 553-563, 2004年 2月27日, 553-563

誉田 太郎 Taro Konda, 中村 佳正 Yoshimasa Nakamura, ロトカ・ボルテラ系による特異値計算アルゴリズムの並列化 A Parallelization of Singular Value Computation Algorithm by the Lotka-Volterra System, 情報処理学会研究報告 Vol.2004 No.128 IPSJ SIG Technical Reports, 日本, 社団法人情報処理学会 Information Processing Society of Japan, 2004年 12月17日, 第2004巻, 13-18

中村 佳正 Yoshimasa NAKAMURA, 高機能化を目指した非線形システム制御 Nonlinear Engineering and Control, システム/制御/情報 第43巻 第11号 SYSTEMS, CONTROL AND INFORMATION, 日本, システム制御情報学会 The Institute of Systems, Control and Information Engineers, 1999年 11月15日, 第43巻, 8-15

中村 佳正 Yoshimasa NAKAMURA, 非線形現象の不思議 ソリトン理論と数値計算法, 電子情報通信学会誌 第80巻 第11号 THE JOURNAL OF THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS, 日本, 社団法人電子情報通信学会 THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS, 1997年 11月25日, 第80巻, 1143-1146

(58)調査した分野(Int.Cl., DB名)

G06F 17/16

G06F 17/30

G06T 1/00

IEEE Xplore

JSTPlus(JDreamII)