

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6615062号
(P6615062)

(45) 発行日 令和1年12月4日 (2019. 12. 4)

(24) 登録日 令和1年11月15日 (2019. 11. 15)

(51) Int. Cl.	F I
GO 6 T 7/00 (2017. 01)	GO 6 T 7/00 6 4 0
GO 6 T 1/00 (2006. 01)	GO 6 T 1/00 2 8 5
GO 6 T 5/00 (2006. 01)	GO 6 T 5/00 7 0 0

請求項の数 20 外国語出願 (全 17 頁)

(21) 出願番号	特願2016-147213 (P2016-147213)	(73) 特許権者	000006013
(22) 出願日	平成28年7月27日 (2016. 7. 27)		三菱電機株式会社
(65) 公開番号	特開2017-59218 (P2017-59218A)		東京都千代田区丸の内二丁目7番3号
(43) 公開日	平成29年3月23日 (2017. 3. 23)	(74) 代理人	100110423
審査請求日	令和1年5月17日 (2019. 5. 17)		弁理士 曾我 道治
(31) 優先権主張番号	14/854, 634	(74) 代理人	100111648
(32) 優先日	平成27年9月15日 (2015. 9. 15)		弁理士 梶並 順
(33) 優先権主張国・地域又は機関	米国 (US)	(74) 代理人	100122437
			弁理士 大宅 一宏
		(74) 代理人	100147566
			弁理士 上田 俊一
早期審査対象出願		(74) 代理人	100161171
			弁理士 吉田 潤一郎
		(74) 代理人	100161115
			弁理士 飯野 智史

最終頁に続く

(54) 【発明の名称】 画像を処理する方法及びシステム

(57) 【特許請求の範囲】

【請求項 1】

シーンの一連の画像をオンライン処理する方法であって、前記一連の画像は画像テンソルとして表され、前記方法は、

少なくとも1つのセンサーからの前記シーンの前記一連の画像を入力インタフェース装置から取得するステップであって、取得された前記シーンの前記一連の画像は、前記シーン内の構造物によって生じる遮蔽を含み、前記構造物は、前記シーンと1以上のセンサーとの間のビル、雲、あるいは前記ビルと前記雲の両方を含む、取得するステップと、

前記一連の画像の第1画像を用いて、低管状階数テンソル及びスパーステンソルを初期化し、前記低管状階数テンソル及び前記スパーステンソルをメモリ内に記憶するステップであって、前記低管状階数テンソルは、低階数スパニングテンソル基底及び対応するテンソル係数のテンソル積である、記憶するステップと、

連続して取得された画像のそれぞれの連続画像を用いて、メモリに記憶された前回の反復からの連続して記憶された低管状階数テンソル及び連続して記憶されたスパーステンソルを反復的に更新し、更新された低管状階数テンソル及び更新されたスパーステンソルを取得する、更新するステップであって、前記更新された低管状階数テンソルは、低階数スパニングテンソル基底及び対応する更新されたテンソル係数のテンソル積である、更新するステップと、

連続して取得された画像のすべての繰り返しが完了したときに、前記更新された低管状階数テンソル、前記更新されたスパーステンソル、あるいは前記更新された低管状階数

10

20

テンソルおよび前記更新されたスパーステンソルの両方のうちのいずれかを出力するステップと、を備え、それぞれのステップは前記メモリと通信するプロセッサにおいて実行される

方法。

【請求項 2】

前記低管状階数テンソルは、前記連続して取得された画像のすべてに共通のシーン内の背景を表し、前記スパーステンソルは、前記連続して取得された画像の 1 以上の遮蔽からの前記シーン内の物体を表す、請求項 1 に記載の方法。

【請求項 3】

前記更新された低管状階数テンソルは、1 組の出力画像を表し、前記更新されたスパーステンソルは、1 組のスパース画像を表す、請求項 1 に記載の方法。

【請求項 4】

前記第 1 画像は、前記低管状階数テンソル及び前記スパーステンソルを初期化するために用いられる前記一連の画像内の 1 組の画像である、請求項 1 に記載の方法。

【請求項 5】

前記画像は多時点画像又はマルチアングルビュー画像である、請求項 1 に記載の方法。

【請求項 6】

前記画像テンソルは、テンソルロバスト主成分分析（PCA）を用いて、前記低管状階数テンソル及び前記スパーステンソルに分解される、請求項 1 に記載の方法。

【請求項 7】

前記低管状階数テンソルは、前記シーン内の背景を表し、前記スパーステンソルは、前記シーン内の前景を表す、請求項 1 に記載の方法。

【請求項 8】

前記低管状階数テンソルは、ノイズが低減された画像を表し、前記スパーステンソルはノイズを表す、請求項 1 に記載の方法。

【請求項 9】

少なくとも 1 つのセンサーは前記シーンの前記一連の画像が取得されるときに移動している、請求項 1 に記載の方法。

【請求項 10】

前記少なくとも 1 つのセンサーは、人工衛星又は航空機内に配置される、請求項 9 に記載の方法。

【請求項 11】

前記少なくとも 1 つのセンサーは、2 以上のセンサーであり、前記 2 以上のセンサーにより取得された前記シーンの前記一連の画像は、前記 2 以上のセンサーとシーンとの間の雲及び前記シーン内の構造物によって生じる遮蔽を含み、1 組の出力画像は、前記遮蔽のない前記シーンを表し、前記出力される 1 組のスパース画像は前記遮蔽のみを表す、請求項 1 に記載の方法。

【請求項 12】

画像を処理するシステムであって、各画像は画像テンソルとして表現され、前記システムは、

少なくとも 1 つのセンサーによって取得された 1 組の入力画像を連続して受信する入力インタフェースであって、取得されたシーンの一連の画像は、前記シーン内の構造物によって生じる遮蔽を含み、前記構造物は、前記シーンと 1 以上のセンサーとの間のビル、雲、あるいは前記ビルと前記雲の両方を含む、入力インタフェースと、

メモリと通信するプロセッサとを備え、前記プロセッサは、

一連画像の第 1 画像を用いて、低管状階数テンソル及びスパーステンソルを初期化し、前記低管状階数テンソル及び前記スパーステンソルをメモリ内に記憶し、

連続して取得された画像のそれぞれの連続画像を用いて、メモリに記憶された前回の反復からの連続して記憶された低管状階数テンソル及び連続して記憶されたスパーステンソルを反復的に更新し、更新された低管状階数テンソル及び更新されたスパーステンソル

10

20

30

40

50

を取得し、

連続して取得された画像のすべての繰り返しが完了したときに、前記更新された低管状階数テンソル、前記更新されたスパーステンソル、あるいは前記更新された低管状階数テンソルおよび前記更新されたスパーステンソルの両方のうちのいずれかを出力するシステム。

【請求項 1 3】

前記低管状階数テンソルは、前記連続して取得された画像のすべてに共通のシーン内の背景を表し、前記スパーステンソルは、前記連続して取得された画像の 1 以上の遮蔽からの前記シーン内の物体を表す、請求項 1 2 に記載のシステム。

【請求項 1 4】

前記更新された低管状階数テンソルは、1 組の出力画像を表し、前記更新されたスパーステンソルは、1 組のスパース画像を表す、請求項 1 2 に記載のシステム。

【請求項 1 5】

前記第 1 画像は、前記低管状階数テンソル及び前記スパーステンソルを初期化するために用いられる前記一連の画像内の 1 組の画像である、請求項 1 2 に記載のシステム。

【請求項 1 6】

前記低管状階数テンソルは、低階数スパニングテンソル基底及び対応するテンソル係数のテンソル積であり、前記更新された低管状階数テンソルは、低階数スパニングテンソル基底及び対応する更新されたテンソル係数のテンソル積である、請求項 1 2 に記載のシステム。

【請求項 1 7】

前記画像テンソルは、テンソルロバスト主成分分析 (PCA) を用いて、前記低管状階数テンソル及び前記スパーステンソルに分解される、請求項 1 2 に記載のシステム。

【請求項 1 8】

前記低管状階数テンソルは、前記シーン内の背景を表し、前記スパーステンソルは、前記シーン内の前景を表す、請求項 1 2 に記載のシステム。

【請求項 1 9】

前記低管状階数テンソルは、ノイズが低減された画像を表し、前記スパーステンソルはノイズを表す、請求項 1 2 に記載のシステム。

【請求項 2 0】

シーンの一連の画像をオンライン処理する方法であって、前記一連の画像は画像テンソルとして表され、前記方法は、

少なくとも 2 つのセンサーからの前記シーンの前記一連の画像を入力インタフェース装置から取得するステップであって、取得された前記シーンの前記一連の画像は、2 以上のセンサーとシーンとの間の雲及び前記シーン内の構造物によって生じる遮蔽を含む、取得するステップと、

取得された前記一連の画像の第 1 画像を用いて、低管状階数テンソル及びスパーステンソルを初期化し、前記低管状階数テンソル及び前記スパーステンソルをメモリ内に記憶するステップであって、前記低管状階数テンソルは、低階数スパニングテンソル基底及び対応するテンソル係数のテンソル積である、記憶するステップと、

連続して取得された画像のそれぞれの連続画像を用いて、メモリに記憶された前回の反復からの連続して記憶された低管状階数テンソル及び連続して記憶されたスパーステンソルを反復的に更新し、更新された低管状階数テンソル及び更新されたスパーステンソルを取得する、更新するステップであって、前記更新された低管状階数テンソルは、低階数スパニングテンソル基底及び対応する更新されたテンソル係数のテンソル積である、更新するステップと

連続して取得された画像のすべての繰り返しが完了したときに、前記更新された低管状階数テンソル、前記更新されたスパーステンソル、あるいは前記更新された低管状階数テンソルおよび前記更新されたスパーステンソルの両方のうちのいずれかを出力するステップと、を備え、それぞれのステップは前記メモリと通信するプロセッサにおいて実行さ

10

20

30

40

50

れる

方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、包括的にはコンピュータービジョンに関し、より詳細には、一連の画像をオンラインで処理することに関する。

【背景技術】

【0002】

多くのコンピュータービジョン用途において、画像を処理して物体を検出するか、又は、例えば背景差分、望ましくないアーチファクト、ノイズ及び遮蔽の除去若しくは低減によって入力画像の品質を改善することができる。画像処理において、次元を低減するために、主成分分析（PCA：principal component analysis）が一般的に適用される。しかしながら、画像データが、全体的な破損、遮蔽又は異常値等の意図しないアーチファクトを含む場合、従来のPCAは失敗する可能性がある。この問題を解決するために、ロバストPCA（RPCA：robust PCA）モデルを用いることができる。

10

【0003】

オンライン再帰RPCAは、オンラインモードで、すなわち、前回の推定値及び新たに取得したデータのみを用いて、データサンプルを分離することができる。最初に全てのデータサンプルを保存し、その後これらを処理する従来のRPCA方法と異なり、オンラインRPCAは、必要とされるメモリ要件を大幅に低減し、計算効率及び収束を改善する。

20

【0004】

2よりも大きい次数の多次元データ（テンソル）の場合、従来の行列ベースの手法を依然として用いることができるように、データをベクトル化することによってベクトル空間にデータを埋め込むことが一般的である。このベクトル化プロセスはほとんどの場合に良好に機能するが、多次元の視点から情報を抽出する際のテンソル表現の効率性が制限される。

【0005】

代替的に、テンソル代数手法は、高次のデータを処理する際の多次元情報の保持において大きな利点を呈する。しかしながら、全ての高次元データが記憶され処理される必要があるので、テンソルRPCAがバッチモードで動作することは非常に時間がかかる。

30

【発明の概要】

【0006】

テンソルロバスト主成分分析（PCA）が、背景差分、ノイズ除去、並びに異常値及び物体検出等の多くの画像処理用途において用いられている。本発明の実施形態は、テンソルの形態で1組の画像を表す多次元データが連続して処理されるオンラインテンソルロバストPCAを提供する。テンソルPCAは、前回の推定及び新たに取得したデータに基づいてテンソルを更新する。

【0007】

バッチモードで動作する従来のPCAと比較して、本発明は、要求されるメモリ量を大幅に低減し、計算効率を改善する。加えて、本方法は、従来のバッチモード手法と比較して、収束速度及び性能が優れている。例えば、性能は、相対二乗誤差による行列ベースのオンラインロバストPCA方法よりも少なくとも10%良好であり、収束速度は、行列ベースのオンラインロバストPCA方法よりも少なくとも3倍速い。

40

【0008】

メモリを低減し、計算効率を増大させるために、本発明者らはオンラインテンソルRPCAアルゴリズムを提供する。このアルゴリズムは、オンライン行列PCA方法を、高次元データ（テンソル）に拡張する。オンラインテンソルRPCAは、部分的に、テンソル特異値分解（t-SVD：tensor singular value decomp

50

osition) 構造に基づく。

【0009】

このテンソル代数の背後にある主要な着想は、続いて、テンソル管に沿ってグループ環を構築することである。例えば、2Dアレイを管のベクトルとみなし、3Dテンソルを管の行列とみなす。そのようなテンソルの枠組みは、高次元データの圧縮及び完成において用いられている。実施形態は、バッチテンソルRPCA問題を拡張し、連続データ集合の利点を提供する。これによって、必要とされるメモリが低減し、効率が增大する。

【図面の簡単な説明】

【0010】

【図1】本発明の実施形態による、画像を向上させるシステムの概略図である。

10

【図2】本発明の実施形態による、画像を向上させる方法の流れ図である。

【図3】本発明の実施形態によって用いられる乗算演算(t積)の概略図である。

【図4】スパニング基底及び係数のt-線形結合によって生成される自由モジュールの要素の概略図である。

【図5】本発明の実施形態によって用いられるt-SVD $X = U * S * V^T$ の概略図である。

【図6】本発明の実施形態による、オンラインテンソルロバストPCAのための擬似コードのブロック図である。

【図7】本発明の実施形態による、データサンプルを投影するための擬似コードのブロック図である。

20

【図8】本発明の実施形態による、スパニング基底を更新するための擬似コードのブロック図である。

【図9】本発明の実施形態による、向上した画像と雲のみのスパース画像とに分離された入力画像の概略図である。

【発明を実施するための形態】

【0011】

図1は、本発明の実施形態による、画像を処理するシステムの概略図である。センサー101、例えば人工衛星内のカメラが、シーン103の1組のT個の入力画像102を連続して捕捉する。画像は、時間ステップtにおいて単一の移動センサーから得ることができる。連続取得により、画像を記憶するためのメモリ要件が低減する。なぜなら、画像は、センサーによって取得され、プロセッサによって受信されるときにオンライン処理されるためである。画像は、互いの位置合わせを容易にするために重複することができる。画像は、グレイスケール画像又はカラー画像とすることができる。さらに、画像は、連続して取得された多時点画像又はマルチアングルビュー画像とすることができる。

30

【0012】

図1に示す例示的な用途では、センサーは移動空間又は空中プラットフォーム(人工衛星、航空機又はドローン)内に配置され、シーン103は地上地形である。シーンは、建物等のシーン内の構造物、及びシーンとセンサーとの間の雲に起因した遮蔽を含む可能性がある。目標は、遮蔽なしの1組の向上した出力画像104を生成することである。副生成物として、システムは、遮蔽、例えば雲のみを含む1組のスパース画像105も生成する。

40

【0013】

図2に示すように、本方法は、センサー101に電氣的又は無線で接続することができるプロセッサ100において動作する。画像を処理する前に、スパニングテンソルが L_0 として初期化され(222)、テンソル係数 R_0 及びスパーステンソル E_0 が初期化される(223)。これについては、以下で説明される図6を参照されたい。

【0014】

1組の入力画像101は、プロセッサによって直接又は間接的に取得され(210)、例えば、画像は、カメラ、ビデオカメラによって取得することもできるし、他の手段によって、又は他のソースから得ることもできる。これは例えば、メモリ転送又は無線通信若

50

しくは無線通信である。本明細書に記載する処理の目的で、各画像は画像テンソル Z_t によって表される。

【0015】

ステップ t における画像ごとに、以下のステップが実行される。画像内のデータサンプルが、前回のスパニングテンソル基底 L_{t-1} を用いてテンソル係数 R_t 及びスパーステンソル E_t に投影される (220)。ここで、

【数1】

$$R_t(t, :, :) = \vec{R}$$

は、スパニング基底 L_{t-1} に対応する係数を表し、

【数2】

$$E_t(:, t, :) = \vec{E}, t = 1, 2, \dots, T$$

である。

【0016】

スパニングテンソル基底は、前回の基底 L_{t-1} を開始点として用いることによって更新される (225)。更新されたスパニングテンソル基底 L_t は、次の処理される画像のために保存される。低階数管状テンソル $X = L_t * R_t^T$ 及びスパーステンソル E_t が更新され (230)、

【数3】

$$X_t + E_t = Z_t$$

であるような1組の出力画像 X_t 104及び1組のスパース画像 E_t 105が生成される。

【0017】

テンソルフレームワークの概観

三次テンソルを例として用いて、本発明の実施形態によって用いられるテンソル構造体を説明する。 $n_1 \times n_3$ の大きさの画像を、従来の画像処理のように次元 $n_1 n_3$ のベクトルにベクトル化する代わりに、各画像を、画像平面に対し垂直な管状スカラーのベクトルとみなす。全てのベクトルが、単位元 (identity) を有する可換環にわたって自由モジュールを形成し、自由モジュールは、明確に定義された (well-defined) 基底及び次元を有するベクトル空間に対し同様にふるまう。

【0018】

表記及び定義

$n_1 \times n_2 \times n_3$ の大きさの三次テンソル A について、 $A(i, j, k)$ は、 A の (i, j, k) 番目の要素を表し、 $A(i, j, :)$ は、 (i, j) 番目の管状スカラーを表す。それぞれ、 $A(i, :, :)$ は、 i 番目の水平方向スライス (horizontal slice) であり、 $A(:, j, :)$ は j 番目の横方向スライス (lateral slice) であり、 $A(:, :, k)$ 又は $A^{(k)}$ は A の k 番目の正面方向スライス (frontal slice) を表す。

【0019】

t 積

【数4】

$$\vec{v} \in \mathbb{R}^{1 \times 1 \times n_3}$$

を、画像平面に対し垂直な n_3 タプルとする。図3に示すように、巡回畳み込みによる2つの n_3 タプル

【数5】

$\vec{u}$
及び

10

20

30

40

【数 6】

$\vec{v}$

間の乗算演算（ t 積）の結果、

【数 7】

$$\vec{w}(i) = \vec{u} * \vec{v} = \sum_{k=0}^{n_3-1} \vec{u}(k) \vec{v}((i-k) \bmod(n_3)) \quad (1)$$

によって表される別の n_3 タプル

【数 8】

$\vec{w}$

が得られる。ここで、 $i = 0, 1, \dots, n_3 - 1$ である。

10

【0 0 2 0】

2つの三次テンソル

【数 9】

$$A \in \mathbb{R}^{n_1 \times n_2 \times n_3}$$

及び

【数 1 0】

$$B \in \mathbb{R}^{n_2 \times n_4 \times n_3}$$

を所与として、 A 及び B の t 積の結果は、以下のように定義される $n_1 \times n_4 \times n_3$ の大きさの三次テンソル C である。

【数 1 1】

20

$$C(i, l, :) = A * B = \sum_{j=1}^{n_2} A(i, j, :) * B(j, l, :) \quad (2)$$

ここで、 $i = 1, 2, \dots, n_1$ 及び $l = 1, 2, \dots, n_4$ である。これは、乗算演算に対応する t 積「 $*$ 」を用いた行列間の乗算と一致する。

【0 0 2 1】

可換環

規定の乗算（ t 積）及び加算の下で、 n_3 タプルの組は、単位元

【数 1 2】

$$\mathbb{R}(\mathbb{G}_{n_3})$$

30

を有する可換環

【数 1 3】

$$\vec{e} = [1, 0, 0, \dots, 0]$$

を形成する。

【0 0 2 2】

可換環にわたる自由モジュール

【数 1 4】

$$\mathbb{M}_{n_3}^{n_1}$$

40

を、 $n_1 \times 1 \times n_3$ の大きさの、1組の全て 2 D の横方向スライスとする。

【数 1 5】

$$\mathbb{M}_{n_3}^{n_1}$$

におけるすべての要素は、管状スカラーのベクトルとして特徴づけることができる。任意の要素

【数 1 6】

$$X \in \mathbb{M}_{n_3}^{n_1}$$

及び係数

50

【数 1 7】

$$v \in \mathbb{R}(\mathbb{G}_{n_3})$$

について、 $Y = X * v$ も

【数 1 8】

$$\mathbb{M}_{n_3}^{n_1}$$

の要素であるため、

【数 1 9】

$$\mathbb{M}_{n_3}^{n_1}$$

10

は管状スカラー乗算の下で閉じている。

【0 0 2 3】

さらに、

【数 2 0】

$$\mathbb{M}_{n_3}^{n_1}$$

は、可換環

【数 2 1】

$$\mathbb{R}(\mathbb{G}_{n_3})$$

20

にわたる次元 n_3 の自由モジュールである。フーリエ変換と巡回畳み込みとの間の関係を用いて、このモジュールのためのスパニング基底

【数 2 2】

$$\{B_1, B_2, \dots, B_{n_3}\}$$

を構築することができる。

【0 0 2 4】

図 4 は、スパニング基底を所与として、任意の要素

【数 2 3】

$$X \in \mathbb{M}_{n_3}^{n_1}$$

30

を管状スカラー係数

【数 2 4】

$$\vec{c}$$

との t -線形結合として一意に表現することができることを示す。

【数 2 5】

$$\vec{X} = \sum_{i=1}^{n_3} \vec{B} * \vec{c}_i \quad (3)$$

【0 0 2 5】

40

テンソル PCA 及びテンソル特異値分解 (t -SVD)

データを概ね含む、より低次元の部分空間を特定する行列 PCA と同様に、高次テンソルデータののためのテンソル PCA を検討する。三次テンソルに焦点を当てる。2D データサンプルが、自由モジュール

【数 2 6】

$$\mathbb{M}_{n_3}^{n_1}$$

のより低次元の自由サブモジュールから到来すると仮定する。ここで、自由サブモジュールは、次元 $d < n$ のスパニング基底を有する

【数 2 7】

$$\mathbb{M}_{n_3}^{n_1}$$

のサブセットである。本発明者らの目標は、2 D データサンプルを含む自由サブモジュールを特定することである。

【0 0 2 6】

t - S V D

 $n_1 \times n_3$ の大きさの n_2 個の 2 D サンプル

【数 2 8】

$$X_1, \dots, X_{n_2}$$

10

を所与として、サンプルを横方向スライスとして配置して、 $n_1 \times n_2 \times n_3$ の大きさの 3 D テンソル X を作成する。次に、t - S V D 方法を用いて、サブモジュールのスパニング基底（主要成分）を決定する。

【0 0 2 7】

図 5 に示すように、t - S V D は以下のように定義される。

【数 2 9】

$$X = U * S * V^T \quad (4)$$

ここで、

【数 3 0】

20

$$U \in \mathbb{R}^{n_1 \times d \times n_3}$$

及び

【数 3 1】

$$V \in \mathbb{R}^{n_2 \times d \times n_3}$$

は、 $U * U^T = I$ かつ $V * V^T = I$ を満たす直交テンソルと呼ばれ、ここで、 I は第 1 の正面方向スライスが単位行列であることを除いて、正面方向スライスが全てゼロである単位テンソルであり、上付き文字 T は転置演算子である。

【0 0 2 8】

【数 3 2】

$$S \in \mathbb{R}^{d \times d \times n_3}$$

30

は、正面方向スライスが対角行列であるテンソルである。画像平面における管状スカラー $S(i, i, :)$, $i = 1, 2, \dots, d$ は特異管と呼ばれ、ここで、 d は管状スカラーの階数である。

【0 0 2 9】

巡回畳み込みと離散フーリエ変換 (DFT : discrete Fourier transform) との間の関係に基づいて、フーリエ領域において S V D により t - S V D を求めることができる。

【数 3 3】

$$\hat{X}$$

を、

40

【数 3 4】

$$\hat{X} = \text{fft}(X, [], 3)$$

によって表されるテンソル X の第 3 の次元に沿った D F T とする。フーリエ領域における S V D

【数 3 5】

$$[U(:, :, k), S(:, :, k), V(:, :, k)] = \text{SVD}(\hat{X}(:, :, k)) \quad \text{ただし } k = 1, \dots, n_3$$

を所与として、式 (4) における t - S V D を以下によって求めることができる。

【数 3 6】

$$U = \text{ifft}(U, [], 3), S = \text{ifft}(S, [], 3), V = \text{ifft}(V, [], 3) \quad (5)$$

ここで、 fft 及び ifft はそれぞれ、高速フーリエ変換及びその逆数を表す。

【0 0 3 0】

行列 SVD の多くの特性が $t-SVD$ に保持されることに留意されたい。これらの中で、重要なものは、立証可能に最適な次元低減のためにトランケートされた $t-SVD$ の最適性である。

【0 0 3 1】

オンラインテンソルロバスト PCA

10

ここで、スパースに破損したデータから低次元のサブモジュールのテンソルを回復する問題を検討する。以下のように分解することができる三次テンソル Z を有すると仮定する。

【数 3 7】

$$Z = X + E \quad (6)$$

ここで、 X は低いテンソル管状階数を有するテンソルであり、 E はスパーステンソルである。テンソル $RPCA$ と呼ばれる、 X 及び E を別個に回復する問題は、最適化問題として定式化することができる。

【数 3 8】

$$\min_{X,E} \|X\|_{\text{TNN}} + \lambda \|E\|_1, \quad \text{s.t.} \quad Z = X + E \quad (7) \quad 20$$

ここで、 λ は所定の重み係数であり、 $\|X\|_{\text{TNN}}$ は、 $t-SVD$ の意味でテンソル X の全ての特異値の合算として定義されるテンソル核ノルムを表し、

【数 3 9】

$$\|E\|_1 = \sum_{i,j,k} |E(i,j,k)|$$

であり、 $\lambda > 0$ である。

【0 0 3 2】

式 (7) は以下の問題と等価であることに留意されたい。

30

【数 4 0】

$$\min_{X,E} \frac{1}{2} \|Z - X - E\|_F^2 + \lambda_1 \|X\|_{\text{TNN}} + \lambda_2 \|E\|_1 \quad (8)$$

ここで、 $\lambda_1, \lambda_2 > 0$ である。

【0 0 3 3】

ここで、オンラインで動作するテンソル PCA の実施態様を説明する。1 組の画像 102 を表す 2D データサンプル $Z(:, i, :)$, $i = 1, 2, \dots, T$ が連続して取得される (210) と仮定する。本発明者らの目標は、画像がプロセッサ 100 において受信されるときに X のスパニング基底 (主成分) をオンラインで推定し、スパーステンソルを同時に分離することである。次に進むために、以下の補題に依拠する。

40

【0 0 3 4】

三次テンソル

【数 4 1】

$$X \in \mathbb{R}^{n_1 \times n_2 \times n_3}$$

について、その階数が r によって上界を設けられると仮定すると、以下が得られる。

【数 4 2】

$$\|X\|_{\text{TNN}} = \inf_{\substack{L \in \mathbb{R}^{n_1 \times r \times n_3} \\ R \in \mathbb{R}^{n_2 \times r \times n_3}}} \left\{ \frac{n_3}{2} (\|L\|_F^2 + \|R\|_F^2) : X = L * R^T \right\} \quad (9)$$

【0 0 3 5】

50

上記の補題を用いて、式（８）を以下のように書き換える。

【数４３】

$$\min_{L,R,E} \frac{1}{2} \|Z - L * R^T - E\|_F^2 + \frac{n_3 \lambda_1}{2} (\|L\|_F^2 + \|R\|_F^2) + \lambda_2 \|E\|_1 \quad s.t. X = L * R^T \quad (10)$$

ここで、

【数４４】

$L \in \mathbb{R}^{n_1 \times r \times n_3}$ 、 $R \in \mathbb{R}^{n_2 \times r \times n_3}$
である。

10

【００３６】

連続して取得されるデータ

【数４５】

$\{\vec{Z}_1, \vec{Z}_2, \dots, \vec{Z}_T\} \in \mathbb{R}^{n_1 \times 1 \times n_3}$

について、式（１０）に基づいて、サンプルごとに以下のように損失関数を定義する。

【数４６】

$$\ell(\vec{Z}_i, L) = \min_{\vec{R}, \vec{E}} \frac{1}{2} \|\vec{Z}_i - L * \vec{R}^T - \vec{E}\|_F^2 + \frac{n_3 \lambda}{2} \|\vec{R}\|_F^2 + \lambda_2 \|\vec{E}\|_1 \quad (11)$$

20

【００３７】

図６は、上記で説明するオンラインテンソルRPCA問題を解くためのアルゴリズム１の擬似コードを示す。図７は、データサンプル、すなわち、アルゴリズム１において用いられる画像テンソル Z_t を投影する（２２０）ためのアルゴリズム２の擬似コードを示す。図８は、アルゴリズム１において用いられるスパニングテンソル基底を更新する（２２５）のためのアルゴリズム３の擬似コードを示す。これらのアルゴリズムによって用いられる全ての変数は、本明細書において定義される。

【００３８】

アルゴリズム１への入力は、取得データ A_T 及びタイムラウンド数 T を含む。簡単にするために、

30

【数４７】

$\hat{A}$
を用いて $\text{fft}(A, [], 3)$ を表し、

【数４８】

$A \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$
を用いて、

【数４９】

$$A = \text{blkdiag}(\hat{A}) = \text{diag}(\hat{A}^{(1)}, \hat{A}^{(2)}, \dots, \hat{A}^{(n_3)}) \quad (12)$$

によって定義される

40

【数５０】

$\hat{A}$
のブロック対角行列を表す。

【００３９】

本発明のオンラインテンソルRPCAアルゴリズムの１つの主要な着想は、新たな画像 Z_t において、スパニングテンソル基底 L_{t-1} の前の推定値を所与として、 Z_t にわたる損失関数を最小にし、最適な R_t 及び E_t を生成するというものである。このとき、代替的には、最後に推定された成分を用いて、累積損失を最小にすることによって、スパニング基底 L_t を更新する（２２５）。

【００４０】

50

特に、 R_t 及び E_t は、ステップ 3 において最適化され、詳細はアルゴリズム 3 に与えられる。アルゴリズム 2 におけるデータ投影 220 ステップにおいて、 $S_\lambda[\cdot]$ は以下によって定義されるソフト閾値処理演算子である。

【数 5 1】

$$S_\lambda[x] = \begin{cases} x - \lambda, & \text{if } x > +\lambda \\ x + \lambda, & \text{if } x < -\lambda \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

【0041】

スパニング基底 L_t を更新する (225) ために、以下を得る。

10

【数 5 2】

$$\begin{aligned} \mathcal{L}_t &= \underset{\mathcal{L}}{\operatorname{argmin}} \sum_{i=1} \left(\frac{1}{2} \|\vec{Z}_i - \mathcal{L} * \vec{R}_i^T - \vec{E}_i\|_F^2 \right. \\ &\quad \left. + \frac{n_3 \lambda_1}{2} \|\vec{R}_i\|_F^2 + \lambda_2 \|\vec{E}_i\|_1 \right) + \frac{n_3 \lambda_1}{2} \|\mathcal{L}\|_F^2 \\ &= \underset{\mathcal{L}}{\operatorname{argmin}} \frac{1}{2} \|\vec{Z}_t - \vec{E}_t - \mathcal{L} * \vec{R}_t^T\|_F^2 + \frac{n_3 \lambda_1}{2} \|\mathcal{L}\|_F^2 \\ &= \underset{\mathcal{L}}{\operatorname{argmin}} \frac{1}{2} \|\vec{Z}_t - \vec{E}_t - \mathcal{L} \vec{R}_t^T\|_F^2 + \frac{n_3 \lambda_1}{2} \|\mathcal{L}\|_F^2 \\ &= \underset{\mathcal{L}}{\operatorname{argmin}} \frac{1}{2} \operatorname{tr}((\vec{Z}_t - \vec{E}_t - \mathcal{L} \vec{R}_t^T)^T (\vec{Z}_t - \vec{E}_t - \mathcal{L} \vec{R}_t^T)) \\ &\quad + \frac{n_3 \lambda_1}{2} \operatorname{tr}(\mathcal{L}^T \mathcal{L}) \\ &= \underset{\mathcal{L}}{\operatorname{argmin}} \frac{1}{2} \operatorname{tr}(\mathcal{L} (\vec{R}_t^T \vec{R}_t + n_3 \lambda_1 I) \mathcal{L}^T) - \operatorname{tr}(\mathcal{L}^T (\vec{Z}_t - \vec{E}_t) \vec{R}_t) \end{aligned} \quad (14)$$

20

ここで、 tr はトレース演算子である。

【0042】

【数 5 3】

30

$$A_t = A_{t-1} + \vec{R}_t * \vec{R}$$

及び

【数 5 4】

$$B_t = B_{t-1} + (\vec{Z} - \vec{E}_t) * \vec{R}_t$$

とする。ここで、アルゴリズム 1 のステップ 4 に示すように、

【数 5 5】

$$\vec{R} \in \mathbb{R}^{r \times 1 \times n_3}, \vec{E}_t \in \mathbb{R}^{n_1 \times 1 \times n_3},$$

40

である。 A_t 、 B_t を新たなデータが到来するたびに更新し、図 1 のステップ 5 に示し、詳細をアルゴリズム 3 に示すように、更新された A_t 、 B_t を、ブロック座標降下法を用いてフーリエ領域におけるスパニング基底 L_t を更新することができるよう保存する。本発明のアルゴリズムは、データサンプル全体の階数の推定される上界に関する事前情報を必要とすることに留意されたい。

【0043】

バッチテンソルロバスト PCA の場合、画像 T までの全てのデータサンプル、すなわち、

【数 5 6】

$$\{Z_i\}_{i=1}^T$$

内の全ての数のエントリが必要とされる。したがって、バッチテンソルロバスト PCA のためのメモリ要件は、 $n_1 n_3 T$ である。

【0 0 4 4】

オンラインテンソルロバスト PCA の場合、

【数 5 7】

$$L_{t-1} \in \mathbb{R}^{n_1 \times r \times n_3}, R_T \in \mathbb{R}^{T \times r \times n_3}$$

(A_T は R_T を通じて求めることができる)、

【数 5 8】

$$B_T \in \mathbb{R}^{n_1 \times r \times n_3}$$

を保存する必要がある。したがって、総ストレージ要件は、 $n_3 r T + n_1 n_3 r$ である。これは、 $r \ll T$ であるとき、バッチテンソルロバスト PCA の場合よりかはるかに小さい。換言すれば、データサンプルをオンラインで処理するためのメモリ要件は、バッチモードテンソルロバスト PCA を用いてデータを処理する場合の約 r / T である。ここで、 r は低階数管状テンソルの階数であり、 T は入力画像の数である。

【0 0 4 5】

他の画像処理用途

本発明は、他の用途のために用いることもできる。前景検出とも呼ばれる背景差分の場合、画像の前景が、例えば、歩道、車両等の物体の検出及び認識等の更なる処理のために抽出される。背景差分は、一連の画像（ビデオ）における移動物体を検出するのにも用いることができる。背景差分は、コンピュータービジョンにおける多数の用途、例えば、監視追跡又は人間の姿勢推定のための重要な手がかりを提供する。

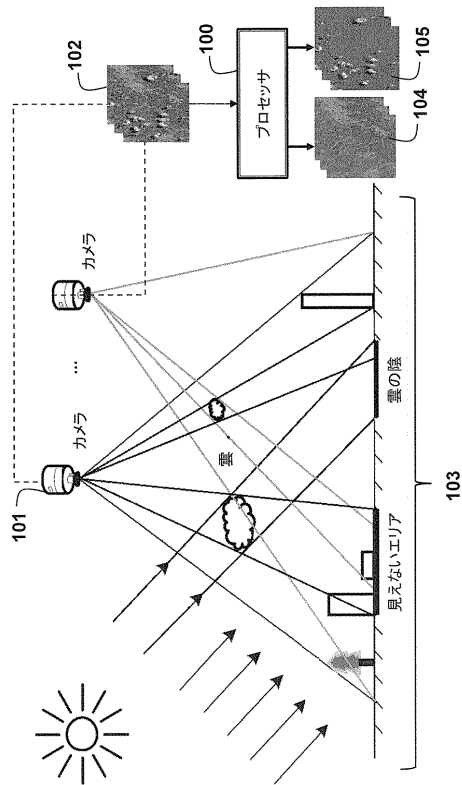
【0 0 4 6】

本発明の実施形態による処理方法において、出力背景画像は、低階数管状テンソル X から構築され、前景画像は、スパーステンソル E から構築される。

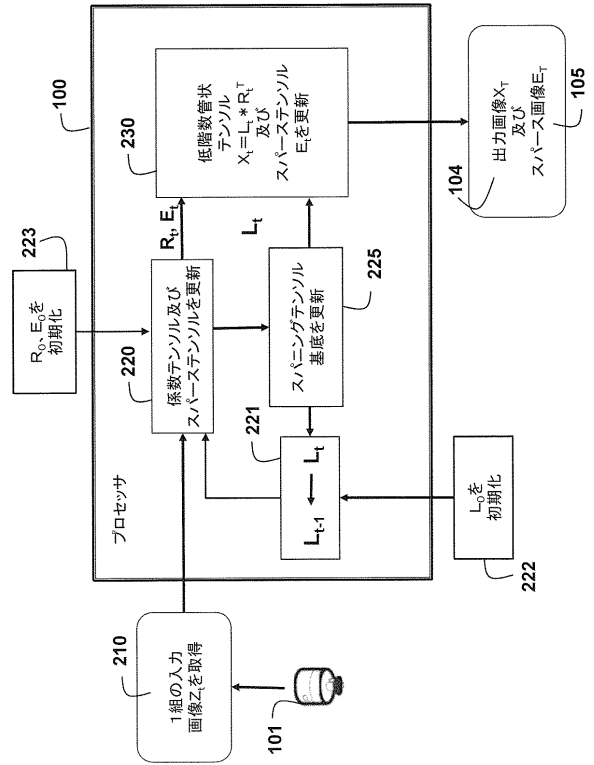
【0 0 4 7】

ノイズ低減の場合、ノイズが低減された画像は、低階数管状テンソル X から導出される。ノイズを表すスパーステンソル E は、本質的に無視することができる。

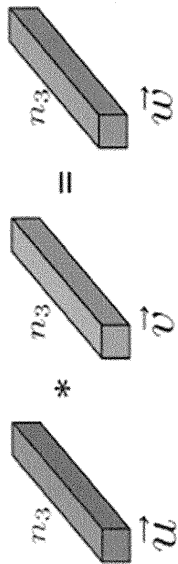
【図 1】



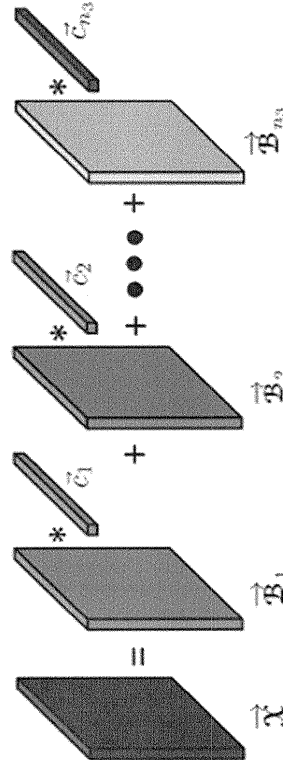
【図 2】



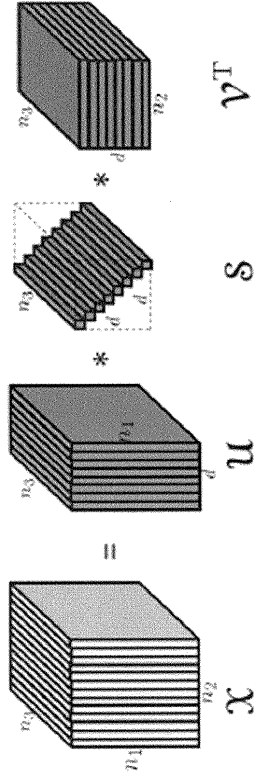
【図 3】



【図 4】



【図 5】



【図 6】

アルゴリズム1 オンラインテンソルロバストPCA

入力: 連続して観測されるデータ $\mathcal{Z}_T = \{\mathcal{Z}_1, \dots, \mathcal{Z}_T\} \in \mathbb{R}^{n_1 \times 1 \times n_3}$, $\lambda_1, \lambda_2 > 0$, ラウンド数 $\mathcal{Z}_0 \in \mathbb{R}^{r \times 1 \times n_3}$, $\mathcal{E}_0 \in \mathbb{R}^{n_1 \times 1 \times n_3}$.

初期: $\mathcal{L}_0 \in \mathbb{R}^{n_1 \times r \times n_3}$, $\mathcal{R}_0 \in \mathbb{R}^{r \times 1 \times n_3}$, $\mathcal{E}_0 \in \mathbb{R}^{n_1 \times 1 \times n_3}$.

1: for $t = 1, 2, \dots, T$ do
 データサンプル $\mathcal{Z}_t$ を公開
 新たなサンプルを投影(アルゴリズム2を参照):
 $\{\mathcal{R}_t, \mathcal{E}_t\} = \underset{\mathcal{R} \in \mathbb{R}^{r \times 1 \times n_3}}{\operatorname{argmin}} \frac{1}{2} \|\mathcal{Z}_t - \mathcal{L}_{t-1} * \mathcal{R} - \mathcal{E}\|_F^2$
 $+ \frac{\lambda_1}{2} \|\mathcal{R}\|_F^2 + \lambda_2 \|\mathcal{E}\|_1$
 4: $\mathcal{A}_t = \mathcal{A}_{t-1} + \mathcal{R}_t * \mathcal{R}_t^T$, $\mathcal{B}_t = \mathcal{B}_{t-1} + (\mathcal{Z}_t - \mathcal{E}_t) * \mathcal{R}_t^T$.
 次に, $\mathcal{A}_t$, $\mathcal{B}_t$ をブロック対角行列形式 $\mathcal{A}_t$, $\mathcal{B}_t$ に書く
 5: $\mathcal{L}_{t-1}$ を開始点として用いて $\mathcal{L}_t$ を計算する(アルゴリズム3を参照)
 $\mathcal{L}_t \triangleq \operatorname{argmin}_{\mathcal{L}} \frac{1}{2} \operatorname{tr}[\mathcal{Z}^T (\mathcal{A}_t + n_3 \lambda_1 I) \mathcal{L}] - \operatorname{tr}(\mathcal{Z}^T \mathcal{B}_t)$.
 6: ブロック対角行列 $\mathcal{L}_t$ をテンソル形式 $\hat{\mathcal{L}}_t$ に編成し,
 $\mathcal{L}_t = \text{ifft}(\hat{\mathcal{L}}_t, [], 3)$ とする
 7: end for
 8: $\mathcal{R}_T(t, :, :) = \mathcal{R}_T^T$, $\mathcal{E}_T(t, t, :) = \mathcal{E}_T^T$, $t = 1, 2, \dots, T$.
 9: $\mathcal{X}_T = \mathcal{L}_T * \mathcal{R}_T^T$, $\mathcal{E}_T$ を返す
 出力: 低rank階数テンソル $\mathcal{X}_T$ 及びスパーステンソル $\mathcal{E}_T$

【図 7】

アルゴリズム2 データサンプルの投影

入力: $\mathcal{L}_{t-1} \in \mathbb{R}^{n_1 \times r \times n_3}$, $\mathcal{Z}_t \in \mathbb{R}^{n_1 \times 1 \times n_3}$, $\lambda_1, \lambda_2 > 0$.

初期: $\hat{\mathcal{E}}_t = 0$, $\hat{\mathcal{R}}_t \in \mathbb{C}^{r \times 1 \times n_3}$.

1: $\hat{\mathcal{Z}}_t = \text{fft}(\mathcal{Z}_t, [], 3)$, $\hat{\mathcal{L}}_{t-1} = \text{fft}(\mathcal{L}_{t-1}, [], 3)$.

2: while 収束しない do
 3: for $i = 1, 2, \dots, n_3$ do
 4: $\hat{\mathcal{r}}_t^{(i)} = ((\hat{\mathcal{L}}_{t-1}^T \hat{\mathcal{L}}_{t-1} + \lambda_1 I)^{-1} \hat{\mathcal{L}}_{t-1}^{(i)})^T (\hat{\mathcal{Z}}_t^{(i)} - \hat{\mathcal{e}}_t^{(i)})$.
 5: end for
 6: $\hat{\mathcal{R}}_t = \text{ifft}(\hat{\mathcal{r}}_t, [], 3)$.
 7: $\hat{\mathcal{E}}_t = S_{\lambda_2}[\mathcal{Z}_t - \mathcal{L}_{t-1}^T * \hat{\mathcal{R}}_t]$.
 8: end while
 出力: $\hat{\mathcal{R}}_t$ 及び $\hat{\mathcal{E}}_t$

【図 8】

アルゴリズム3 スパニングテンソル基底の更新

入力: $\mathcal{L}_{t-1} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, $\mathcal{A}_t \in \mathbb{R}^{r \times r \times n_3}$, $\mathcal{B}_t \in \mathbb{R}^{n_1 \times r \times n_3}$

出力: $\mathcal{L}_t$.

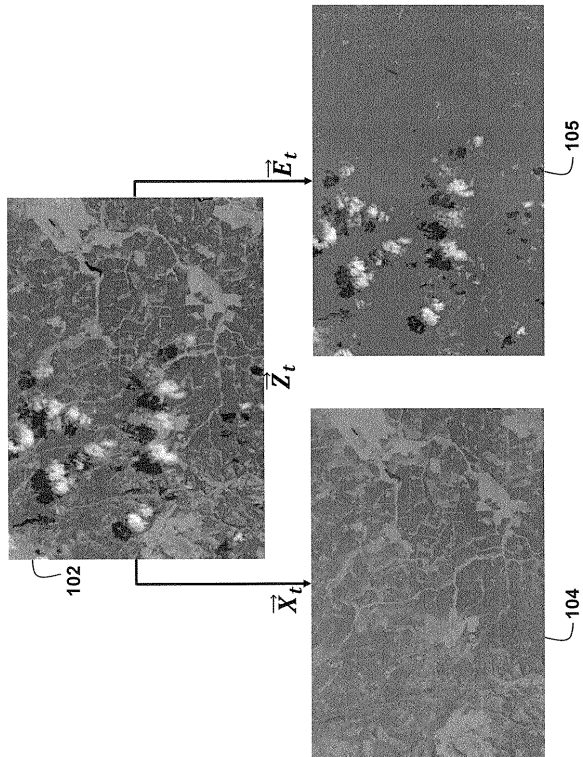
1: $\mathcal{L}_t = \mathcal{L}_{t-1}$, $\hat{\mathcal{L}}_t = \text{fft}(\mathcal{L}_t, [], 3)$.

2: $\hat{\mathcal{B}}_t = \text{fft}(\mathcal{B}_t, [], 3)$.

3: $\mathcal{C}_t = \mathcal{A}_t + \lambda_1 I$, $\hat{\mathcal{C}}_t = \text{fft}(\mathcal{C}_t, [], 3)$.

4: for $j = 1, 2, \dots, r$ do
 5: for $k = 1, 2, \dots, n_3$ do
 6: $\hat{\mathcal{L}}_t(:, j, k) = \frac{\hat{\mathcal{B}}_t(:, j, k) - \hat{\mathcal{L}}_t(:, j, k) * \hat{\mathcal{C}}_t(:, j, k)}{\hat{\mathcal{C}}_t(j, j, k)}$
 7: end for
 8: end for
 9: $\mathcal{L}_t = \text{ifft}(\hat{\mathcal{L}}_t, [], 3)$.

【図 9】



フロントページの続き

(74)代理人 100188329

弁理士 田村 義行

(72)発明者 デホン・リウ

アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウェイ 201、ケアオブ・ミッ
ビシ・エレクトリック・リサーチ・ラボラトリーズ・インコーポレイテッド

(72)発明者 ゼミン・ザン

アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウェイ 201、ケアオブ・ミッ
ビシ・エレクトリック・リサーチ・ラボラトリーズ・インコーポレイテッド

(72)発明者 アンソニー・ヴェトロ

アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウェイ 201、ケアオブ・ミッ
ビシ・エレクトリック・リサーチ・ラボラトリーズ・インコーポレイテッド

審査官 村松 貴士

(56)参考文献 米国特許出願公開第2013/0191425 (US, A1)

特開2013-143102 (JP, A)

Zemin Zhang, 外4名, "Novel methods for multilinear data completion and de-noising bas
ed on tensor-SVD", 2014 IEEE Conference on Computer Vision and Pattern Recognition,
米国, IEEE, 2014年, p.3842-3849

(58)調査した分野(Int.Cl., DB名)

G06T 1/00 — 7/90