

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4669993号  
(P4669993)

(45) 発行日 平成23年4月13日(2011.4.13)

(24) 登録日 平成23年1月28日(2011.1.28)

(51) Int.Cl. F 1  
G 0 6 T 7 / 6 0 (2006.01) G 0 6 T 7 / 6 0 2 5 0 C

請求項の数 20 (全 44 頁)

<p>(21) 出願番号 特願2005-319646 (P2005-319646)                  (22) 出願日 平成17年11月2日 (2005.11.2)                  (65) 公開番号 特開2007-128237 (P2007-128237A)                  (43) 公開日 平成19年5月24日 (2007.5.24)                  審査請求日 平成20年10月30日 (2008.10.30)</p> <p>特許法第30条第1項適用 2005年5月19日 システム制御情報学会主催の「第49回 システム制御情報学会研究発表講演会」において文書をもって発表</p>	<p>(73) 特許権者 800000068                  学校法人東京電機大学                  東京都千代田区神田錦町2-2                  (74) 代理人 100067541                  弁理士 岸田 正行                  (74) 代理人 100087398                  弁理士 水野 勝文                  (74) 代理人 100126147                  弁理士 川上 成年                  (72) 発明者 狩野 弘之                  東京都千代田区神田錦町2の2 学校法人                  東京電機大学内                  (72) 発明者 藤岡 寛之                  東京都千代田区神田錦町2の2 学校法人                  東京電機大学内</p> <p style="text-align: right;">最終頁に続く</p>
---	--

(54) 【発明の名称】 最適平滑化スプラインによる極値検出方法及び極値検出プログラム

(57) 【特許請求の範囲】

【請求項1】

データが入力される入力手段と、プログラムを格納する記憶手段と、前記プログラムを実行し前記データを処理する演算手段と、前記演算手段の処理結果を出力する出力手段とを備える処理装置で実行される極値検出方法であって、

前記入力手段から入力された平面における離散データ群又は関数から、前記演算手段が複数の節点を有する最適平滑化スプライン曲線を生成するステップと、

前記演算手段が、前記最適平滑化スプライン曲線上又はその導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、前記最適平滑化スプライン曲線又はその導関数の極値のすべてを求め、該極値を前記出力手段へ出力するステップと、

を有することを特徴とする最適平滑化スプライン曲線による極値検出方法。

【請求項2】

前記最適平滑化スプライン曲線又はその導関数の極値のすべてを求め、該極値を前記出力手段へ出力するステップは、

前記演算手段が、前記最適平滑化スプライン曲線又はその導関数の各節点区間について各々正規化して、単位区間における正規化された多項式関数又はその導関数を生成するステップと、

前記演算手段が、前記単位区間における正規化された多項式関数又はその導関数について極値を求めることにより、前記最適平滑化スプライン曲線又はその導関数の極値のすべ

てを求めるステップと、  
を有することを特徴とする請求項 1 に記載の最適平滑化スプライン曲線による極値検出方法。

【請求項 3】

前記演算手段が前記最適平滑化スプライン曲線を生成するステップは、  
前記演算手段が前記平面における離散データ群又は関数から節点ごとの最適重み係数を求めるステップと、  
前記演算手段が前記節点ごとの正規化された一様な B スプライン関数を、前記節点ごとの最適重み係数により重みづけして加え合わせて構成することにより、最適平滑化スプライン曲線を生成するステップと、  
を有することを特徴とする請求項 1 または 2 に記載の最適平滑化スプライン曲線による極値検出方法。

10

【請求項 4】

前記スプライン関数は、3 次のスプライン関数であることを特徴とする請求項 3 に記載の最適平滑化スプライン曲線による極値検出方法。

【請求項 5】

データが入力される入力手段と、プログラムを格納する記憶手段と、前記プログラムを実行し前記データを処理する演算手段と、前記演算手段の処理結果を出力する出力手段とを備える処理装置で実行される極値検出方法であって、

前記入力手段から入力された空間における離散データ群又は関数から前記演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、

20

前記演算手段が前記最適平滑化スプライン曲面上又はその偏導関数上の前記節点により格子上に区画された一領域である各節点領域の各々について各々極値を求めることにより、前記最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求め、該極値を前記出力手段へ出力するステップと、

を有することを特徴とする最適平滑化スプライン曲面による極値検出方法。

【請求項 6】

前記最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求め、該極値を前記出力手段へ出力するステップは、

前記演算手段が前記最適平滑化スプライン曲面又はその偏導関数の各節点領域について各々正規化して、単位領域における正規化された多項式関数又はその偏導関数を生成するステップと、

30

前記演算手段が前記単位領域における正規化された多項式関数又はその偏導関数について極値を求めることにより、前記最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求めるステップと、

を有することを特徴とする請求項 5 に記載の最適平滑化スプライン曲面による極値検出方法。

【請求項 7】

前記演算手段が前記最適平滑化スプライン曲面を生成するステップは、

前記演算手段が前記空間における離散データ群又は関数から節点ごとの最適重み係数を求めるステップと、

40

前記演算手段が前記節点ごとの正規化された一様な B スプライン関数を、前記節点ごとの最適重み係数により重みづけして加え合わせて構成することにより、最適平滑化スプライン曲面を生成するステップと、

を有することを特徴とする請求項 5 または 6 に記載の最適平滑化スプライン曲面による極値検出方法。

【請求項 8】

ピクセルデータが入力される入力手段と、プログラムを格納する記憶手段と、前記プログラムを実行し前記ピクセルデータを処理する演算手段と、前記演算手段の処理結果を出力する出力手段とを備える画像処理装置で実行されるデジタル画像のエッジ検出方法であ

50

って、

前記入力手段から入力されたピクセルの座標及び輝度値からなるピクセルデータから、前記演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、

前記演算手段が前記最適平滑化スプライン曲面を第1の座標軸上の複数の座標で固定した複数の曲線からなる第1の曲線群を生成し、該第1の曲線群を構成する複数の曲線の1次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、前記第1の曲線群の1次導関数の極値のすべてを求めるステップと、

前記演算手段が前記最適平滑化スプライン曲面を第2の座標軸上の複数の座標で固定した複数の曲線からなる第2の曲線群を生成し、該第2の曲線群を構成する複数の曲線の1次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、前記第2の曲線群の1次導関数の極値のすべてを求めるステップと、

前記演算手段が前記第1の曲線群の1次導関数の極値又は前記第2の曲線群の1次導関数の極値の絶対値が所定のしきい値より大きい箇所を、デジタル画像のエッジ部分と判定し、該判定結果を前記出力手段へ出力するステップと、

を有することを特徴とするデジタル画像のエッジ検出方法。

【請求項9】

前記第1の曲線群の1次導関数の極値のすべてを求めるステップは、

前記演算手段が前記第1の曲線群を構成する複数の曲線の1次導関数の各節点区間について各々正規化して、単位区間における正規化された第1の曲線群を構成する複数の曲線の1次導関数の極値を求めることにより、前記第1の曲線群の1次導関数の極値のすべてを求めるステップと、を有し、

前記第2の曲線群の1次導関数の極値のすべてを求めるステップは、

前記演算手段が前記第2の曲線群を構成する複数の曲線の1次導関数の各節点区間について各々正規化して、単位区間における正規化された第2の曲線群を構成する複数の曲線の1次導関数の極値を求めることにより、前記第2の曲線群の1次導関数の極値のすべてを求めるステップと、

を有することを特徴とする請求項8に記載のデジタル画像のエッジ検出方法。

【請求項10】

前記演算手段が前記最適平滑化スプライン曲面を生成するステップは、

前記演算手段が前記ピクセルの座標及び輝度値からなるピクセルデータから節点ごとの最適重み係数を求めるステップと、

前記演算手段が前記節点ごとの正規化された一様なBスプライン関数を、前記節点ごとの最適重み係数により重みづけして加え合わせて構成することにより、最適平滑化スプライン曲面を生成するステップと、

を有することを特徴とする請求項8または9に記載のデジタル画像のエッジ検出方法。

【請求項11】

データが入力される入力手段と、プログラムを格納する記憶手段と、前記プログラムを実行し前記データを処理する演算手段と、前記演算手段の処理結果を出力する出力手段とを備える処理装置に、

前記入力手段から入力された平面における離散データ群又は関数から、前記演算手段が複数の節点を有する最適平滑化スプライン曲線を生成するステップと、

前記演算手段が前記最適平滑化スプライン曲線上又はその導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、前記最適平滑化スプライン曲線又はその導関数の極値のすべてを求め、該極値を前記出力手段へ出力するステップと、

を実行させるための最適平滑化スプライン曲線による極値検出プログラム。

【請求項12】

前記最適平滑化スプライン曲線又はその導関数の極値のすべてを求め、該極値を前記出

10

20

30

40

50

力手段に出力するステップは、

前記演算手段が前記最適平滑化スプライン曲線又はその導関数の各節点区間について各々正規化して、単位区間における正規化された多項式関数又はその導関数を生成するステップと、

前記演算手段が前記単位区間における正規化された多項式関数又はその導関数について極値を求めることにより、前記最適平滑化スプライン曲線又はその導関数の極値のすべてを求めるステップと、

を有することを特徴とする請求項 1 1 に記載の最適平滑化スプライン曲線による極値検出プログラム。

【請求項 1 3】

前記演算手段が前記最適平滑化スプライン曲線を生成するステップは、

前記演算手段が前記平面における離散データ群又は関数から節点ごとの最適重み係数を求めるステップと、

前記演算手段が前記節点ごとの正規化された一様な B スプライン関数を、前記節点ごとの最適重み係数により重みづけして加え合わせて構成することにより、最適平滑化スプライン曲線を生成するステップと、

を有することを特徴とする請求項 1 1 または 1 2 に記載の最適平滑化スプライン曲線による極値検出プログラム。

【請求項 1 4】

前記スプライン関数は、3 次のスプラインであることを特徴とする請求項 1 3 に記載の最適平滑化スプライン曲線による極値検出プログラム。

【請求項 1 5】

データが入力される入力手段と、プログラムを格納する記憶手段と、前記プログラムを実行し前記データを処理する演算手段と、前記演算手段の処理結果を出力する出力手段とを備える処理装置に、

前記入力手段から入力された空間における離散データ群又は関数から、前記演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、

前記演算手段が前記最適平滑化スプライン曲面上又はその偏導関数上の前記節点により格子上に区画された一領域である各節点領域の各々について各々極値を求めることにより、前記最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求め、該極値を前記出力手段に出力するステップと、

を実行させるための最適平滑化スプライン曲面による極値検出プログラム。

【請求項 1 6】

前記最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求め、該極値を前記出力手段に出力するステップは、

前記演算装置が前記最適平滑化スプライン曲面又はその偏導関数の各節点領域について各々正規化して、単位領域における正規化された多項式関数又はその導関数を生成するステップと、

前記演算装置が前記単位領域における正規化された多項式関数又はその偏導関数について極値を求めることにより、前記最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求めるステップと、

を有することを特徴とする請求項 1 5 に記載の最適平滑化スプライン曲面による極値検出プログラム。

【請求項 1 7】

前記演算手段が前記最適平滑化スプライン曲面を生成するステップは、

前記演算手段が空間における離散データ群又は関数から節点ごとの最適重み係数を求めるステップと、

前記演算手段が前記節点ごとの正規化された一様な B スプライン関数を、前記節点ごとの最適重み係数により重みづけして加え合わせて構成することにより、最適平滑化スプライン曲面を生成するステップと、

10

20

30

40

50

を有することを特徴とする請求項 15 または 16 に記載の最適平滑化スプライン曲面による極値検出プログラム。

【請求項 18】

ピクセルデータが入力される入力手段と、プログラムを格納する記憶手段と、前記プログラムを実行し前記ピクセルデータを処理する演算手段と、前記演算手段の処理結果を出力する出力手段とを備える画像処理装置に、

前記入力手段から入力されたピクセルの座標及び輝度値からなるピクセルデータから前記演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、

前記演算手段が前記最適平滑化スプライン曲面を第 1 の座標軸上の複数の座標で固定した複数の曲線からなる第 1 の曲線群を生成し、該第 1 の曲線群を構成する複数の曲線の 1 次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、前記第 1 の曲線群の 1 次導関数の極値のすべてを求めるステップと、

前記演算手段が前記最適平滑化スプライン曲面を第 2 の座標軸上の複数の座標で固定した複数の曲線からなる第 2 の曲線群を生成し、該第 2 の曲線群を構成する複数の曲線の 1 次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、前記第 2 の曲線群の 1 次導関数の極値のすべてを求めるステップと、

前記演算手段が前記第 1 の曲線群の 1 次導関数の極値又は前記第 2 の曲線群の 1 次導関数の極値の絶対値が所定のしきい値より大きい箇所を、デジタル画像のエッジ部分と判定し、該判定結果を前記出力手段へ出力するステップと、  
を実行させるためのデジタル画像のエッジ検出プログラム。

【請求項 19】

前記第 1 の曲線群の 1 次導関数の極値のすべてを求めるステップは、

前記演算手段が前記第 1 の曲線群を構成する複数の曲線の 1 次導関数の各節点区間について各々正規化して、単位区間における正規化された前記第 1 の曲線群を構成する複数の曲線の 1 次導関数の極値を求めることにより、前記第 1 の曲線群の 1 次導関数の極値のすべてを求めるステップと、を有し、

前記第 2 の曲線群の 1 次導関数の極値のすべてを求めるステップは、

前記演算手段が前記第 2 の曲線群を構成する複数の曲線の 1 次導関数の各節点区間について各々正規化して、単位区間における正規化された前記第 2 の曲線群を構成する複数の曲線の 1 次導関数の極値を求めることにより、前記第 2 の曲線群の 1 次導関数の極値のすべてを求めるステップと、

を有することを特徴とする請求項 18 に記載のデジタル画像のエッジ検出プログラム。

【請求項 20】

前記演算手段が前記最適平滑化スプライン曲面を生成するステップは、

前記演算手段が前記ピクセルの座標及び輝度値からなるピクセルデータから節点ごとの最適重み係数を求めるステップと、

前記演算手段が前記節点ごとの正規化された一様な B スプライン関数を、前記節点ごとの最適重み係数により重みづけして加え合わせて構成することにより、最適平滑化スプライン曲面を生成するステップと、

を有することを特徴とする請求項 18 または 19 に記載のデジタル画像のエッジ検出プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、離散データ点や関数に関する極値検出方法及び極値検出プログラムに関し、特に、離散データ点や関数に関する最適平滑化スプラインを設計することにより、すべての極値を検出する極値検出方法及び極値検出プログラム、およびその極値検出方法及び極値検出プログラムを利用したデジタル画像のエッジ検出方法及びエッジ検出プログラムに

10

20

30

40

50

関する。

【背景技術】

【0002】

離散データ点や関数に関する極値検出の問題は、工学を始めとする様々な分野で現れ、いわゆる「数理計画法」あるいは「最適化法」という分野の中で重要な問題として知られている。離散データ点や関数に関する極値検出の方法は、山登り法（山下り法）に代表されるような数値的な極値探索法が一般的であり、従って局所的な方法である。

【0003】

しかしながら、このような局所的な方法による極値探索は、離散データ点の局所のノイズの影響を受けやすく、極値を精度良く安定的に検出することが困難である。したがって、離散データ点に関して大域的に評価することが可能な極値検出方法が求められている。

10

【0004】

離散データ点に関して大域的に評価することが可能な極値検出方法に関しては、例えば、特許文献1に記載の方法がある。この方法によれば、1組のデータ点について大域的にサンプリングを行い、この1組のデータ点に基づいて単一のポリラインを生成し、このポリラインの極値を求める方法が開示されている。

【特許文献1】特開2005-222554号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

20

特許文献1に記載の方法においては、曲線を作成する際に使用する「特定の点」や「分割点」の選択に、曲率（2次導関数）が用いられている。特に「特定の点」は、曲率が極値（極小値、極大値）をとる点あるいは0になる点として選ばれている。一方、「分割点」は曲率の積分値に基づいて選択されている。

【0006】

しかしながら、曲率の極値や、曲率が0になる点の計算法は何ら具体的に示されていない。曲率の値を幾つかの点で計算し、得られた値から極大や極小、さらに0になる点を選んでいないのかと推測されるが、定かではない。

【0007】

また、特許文献1に記載の方法は、曲線設計に関する高い技術を有していないユーザを対象として、ユーザの希望する曲線を描き、それを修正することを支援するものに過ぎず、離散データ点に関して大域的に評価することが可能ではない。

30

【0008】

本発明は、このような極値に関する従来の問題も解決するもので、離散データ点や関数に関する最適平滑化スプラインを設計することにより、最適平滑化スプラインに対する極値を精度良く安定的に検出する方法を提供しようとするものである。また、すべての極値が求まることから、最大値や最小値も得られる方法を提供しようとするものである。

【課題を解決するための手段】

【0009】

本発明は、データが入力される入力手段と、プログラムを格納する記憶手段と、プログラムを実行しデータを処理する演算手段と、演算手段の処理結果を出力する出力手段とを備える処理装置で実行される極値検出方法であって、入力手段から入力された平面における離散データ群又は関数から、演算手段が複数の節点を有する最適平滑化スプライン曲線を生成するステップと、演算手段が、前記最適平滑化スプライン曲線上又はその導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、最適平滑化スプライン曲線又はその導関数の極値のすべてを求め、極値を出力手段へ出力するステップとを有することを特徴とする最適平滑化スプライン曲線による極値検出方法である。

40

【0010】

50

また、本発明は、データが入力される入力手段と、プログラムを格納する記憶手段と、プログラムを実行しデータを処理する演算手段と、演算手段の処理結果を出力する出力手段とを備える処理装置で実行される極値検出方法であって、入力手段から入力された空間における離散データ群又は関数から演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、演算手段が最適平滑化スプライン曲面上又はその偏導関数上の前記節点により格子上に区画された一領域である各節点領域の各々について各々極値を求めることにより、最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求め、極値を前記出力手段へ出力するステップとを有することを特徴とする最適平滑化スプライン曲面による極値検出方法である。

10

## 【0011】

また、本発明は、ピクセルデータが入力される入力手段と、プログラムを格納する記憶手段と、プログラムを実行しピクセルデータを処理する演算手段と、演算手段の処理結果を出力する出力手段とを備える画像処理装置で実行されるデジタル画像のエッジ検出方法であって、入力手段から入力されたピクセルの座標及び輝度値からなるピクセルデータから、演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、演算手段が最適平滑化スプライン曲面を第1の座標軸上の複数の座標で固定した複数の曲線からなる第1の曲線群を生成し、該第1の曲線群を構成する複数の曲線の1次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、第1の曲線群の1次導関数の極値のすべてを求めるステップと、演算手段が最適平滑化スプライン曲面を第2の座標軸上の複数の座標で固定した複数の曲線からなる第2の曲線群を生成し、第2の曲線群を構成する複数の曲線の1次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、第2の曲線群の1次導関数の極値のすべてを求めるステップと、演算手段が第1の曲線群の1次導関数の極値又は第2の曲線群の1次導関数の極値の絶対値が所定のしきい値より大きい箇所を、デジタル画像のエッジ部分と判定し、判定結果を出力手段へ出力するステップとを有することを特徴とするデジタル画像のエッジ検出方法である。

20

## 【0012】

また、本発明は、データが入力される入力手段と、プログラムを格納する記憶手段と、プログラムを実行しデータを処理する演算手段と、演算手段の処理結果を出力する出力手段とを備える処理装置に、入力手段から入力された平面における離散データ群又は関数から、演算手段が複数の節点を有する最適平滑化スプライン曲線を生成するステップと、演算手段が最適平滑化スプライン曲線上又はその導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、最適平滑化スプライン曲線又はその導関数の極値のすべてを求め、極値を出力手段に出力するステップとを実行させるための最適平滑化スプライン曲線による極値検出プログラムである。

30

## 【0013】

また、本発明は、データが入力される入力手段と、プログラムを格納する記憶手段と、プログラムを実行しデータを処理する演算手段と、演算手段の処理結果を出力する出力手段とを備える処理装置に、入力手段から入力された空間における離散データ群又は関数から、演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、演算手段が最適平滑化スプライン曲面上又はその偏導関数上の前記節点により格子上に区画された一領域である各節点領域の各々について各々極値を求めることにより、最適平滑化スプライン曲面又はその偏導関数の極値のすべてを求め、極値を出力手段に出力するステップとを実行させるための最適平滑化スプライン曲面による極値検出プログラムである。

40

## 【0014】

また、本発明は、ピクセルデータが入力される入力手段と、プログラムを格納する記憶

50

手段と、プログラムを実行しピクセルデータを処理する演算手段と、演算手段の処理結果を出力する出力手段とを備える画像処理装置に、入力手段から入力されたピクセルの座標及び輝度値からなるピクセルデータから演算手段が複数の節点を有する最適平滑化スプライン曲面を生成するステップと、演算手段が最適平滑化スプライン曲面を第1の座標軸上の複数の座標で固定した複数の曲線からなる第1の曲線群を生成し、第1の曲線群を構成する複数の曲線の1次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、第1の曲線群の1次導関数の極値のすべてを求めるステップと、演算手段が最適平滑化スプライン曲面を第2の座標軸上の複数の座標で固定した複数の曲線からなる第2の曲線群を生成し、第2の曲線群を構成する複数の曲線の1次導関数上の隣接する前記節点により区分された一区間である各節点区間の各々について各々極値を求めることにより、第2の曲線群の1次導関数の極値のすべてを求めるステップと、演算手段が第1の曲線群の1次導関数の極値又は第2の曲線群の1次導関数の極値の絶対値が所定のしきい値より大きい箇所を、デジタル画像のエッジ部分と判定し、判定結果を出力手段へ出力するステップとを実行させるためのデジタル画像のエッジ検出プログラムである。

10

【発明の効果】

【0015】

本発明の最適平滑化スプラインによる極値検出方法によれば、設計されたスプラインを対象とするため、各節点ごとに定義された多項式の極値（極小値、極大値）を求めれば良い。これによりすべての極値が求まり、最小値や最大値を求めることが可能となる。

20

【発明を実施するための最良の形態】

【0016】

（第1実施形態）

以下、本発明の第1実施形態である最適平滑化スプライン曲線による極値検出方法及び極値検出プログラムについて、図を参照して詳細に説明をする。

【0017】

よく知られるように、スプライン関数は様々な工学分野で用いられており、これまで幅広い研究がなされてきている。特に、最適制御理論に基づいた「動的スプライン」の理論が提案されており、そのようなアプローチは、Bスプライン関数の解析や、最適平滑化曲線の設計に利用されている。さらには、最適内挿および平滑化スプラインの枠組みが日本の書道で見られるような文字や文字列の設計および再設計を行うために利用されている。

30

【0018】

本実施形態では、まず、与えられた離散データの集合に対して最適平滑化スプライン曲線の設計を行う。そのような曲線の設計には、正規化された一様なBスプライン関数を基底関数として用いる。まず、最適平滑化曲線の設計について、簡潔な計算手順が得られる基本的な結果を示す。次に、最適曲線のすべての極値、すなわち極小値、極大値を計算するためのアルゴリズムを示す。その際、スプラインが区分的多項式であり、連続かつ微分可能であるということを利用するものである。スプライン曲線は、区分的多項式であり制御点によって表すことができるので、その極値を検出し計算するアルゴリズムは簡潔かつ数値計算に対しても利用しやすいものとなっている。

40

【0019】

$B_k(\cdot)$ を、整数値の節点をもつk次の正規化された一様なBスプライン関数とし、一定間隔でシフトし重みづけしたBスプライン関数の和としてスプライン曲線 $x(t)$ を生成する。すなわち、k次スプライン曲線 $x(t)$ は、

【0020】

【数1】

$$x(t) = \sum_{i=-k}^{m-1} \tau_i B_k(\alpha(t - t_i)).$$

50



## 【 0 0 2 1 】

のように表現される。

## 【 0 0 2 2 】

ここで、 $m$ は、整数であって、予め定めうる設計パラメータである。 $\alpha_i$   $R$ は制御点と呼ばれる重み係数である。 $\Delta$  ( $>0$ )は、予め定めうる設計パラメータであり、等間隔に配置された節点 $t_i$ の間隔(節点区間)を、 $t_{i+1} - t_i = \Delta$  に従って調整するための定数である。

## 【 0 0 2 3 】

数式 1 によってスプライン曲線 $x(t)$ が構成される様子を図 1 に示す。各つり鐘状の波形は適宜シフトされた $k$ 次Bスプライン $B_k(\cdot)$ に対応する制御点 $\alpha_i$  (「x」印で示す)によって重み付けされた関数を表し、スプライン曲線 $x(t)$ はこれらをすべて加え合わせて構成されている。制御点 $\alpha_i$   $R$ を適切に選ぶことによって、任意の $k$ 次スプライン曲線 $x(t)$ を区間 $[t_0, t_m]$ で設計することができる。

10

## 【 0 0 2 4 】

$k$ 次B - スプライン関数 $B_k(t)$ は、次式

## 【 0 0 2 5 】

## 【数 2】

$$B_k(t) = \begin{cases} N_{k-j, k}(t-j) & j \leq t \leq j+1 \quad j = 0, \dots, k \\ 0 & t \leq 0, k+1 \leq t \end{cases} \quad 20$$

..... (2)

## 【 0 0 2 6 】

で定義されている。 $B_k(t)$ は整数値の節点をもち、 $j=0, \dots, k$ に対する各節点区間 $[j, j+1]$ で基底関数 $N_{j, k}(t-j)$ によって指定され、 $[0, k+1]$ 以外の区間では恒等的に0である。

## 【 0 0 2 7 】

ここで基底関数 $N_{j, k}(t)$ は、

## 【 0 0 2 8 】

30

【数3】

$$N_{0, k}(t) = \frac{1-t}{k} N_{0, k-1}(t)$$

$$N_{k, k}(t) = \frac{t}{k} N_{k-1, k-1}(t)$$

$$N_{j, k}(t) =$$

$$\frac{k-j+t}{k} N_{j-1, k-1}(t) + \frac{1+j-t}{k} N_{j, k-1}(t)$$

$$j = 1, \dots, k-1 \quad \dots\dots (3)$$

10

【0029】

によって、初期関数 $N_{0,0}(t)=1$ から順次生成することができる。

【0030】

實際上、 $k=3$ 、すなわち3次のスプライン関数を選定した場合、基底関数 $N_{j,3}(t)$ は次式

【0031】

【数4】

$$N_{0,3}(t) = (1-t)^3 / 6$$

$$N_{1,3}(t) = (4-6t^2+3t^3) / 6$$

$$N_{2,3}(t) = (1+3t+3t^2-3t^3) / 6$$

$$N_{3,3}(t) = t^3 / 6 \quad \dots\dots (4)$$

20

30

【0032】

のように、4つ(すなわち $k+1$ 個)の関数を含むことになる。

【0033】

このことは、各 $t$ における $k$ 次スプライン関数の値は、 $t$ が含まれる区間における $k+1$ 個の基底関数(従って対応する $k+1$ 個の制御点)の影響を受けるだけで、その範囲外の基底関数(従って対応する制御点)の影響を受けないことを意味している。

【0034】

(最適化過程)

次に、離散データ $D$ の集合が

【0035】

40

【数5】

$$\mathcal{D} = \{(u_i; d_i) : t_0 \leq u_i \leq t_m, d_i \in \mathbf{R}, i = 1, \dots, N\}$$

【0036】

によって与えられているとし、また、 $\mathbf{R}^M (M=m+k)$  を以下のように定義される制御点ベクトルとする。

【0037】

【数6】

10

$$\tau = [\tau_{-k} \quad \tau_{-k+1} \quad \cdots \quad \tau_{m-1}]^T$$

【0038】

そのとき、最適平滑化スプライン曲線の設計の基本問題は、評価関数

【0039】

【数7】

20

$$J(\tau) = \lambda \int_{t_0}^{t_m} (x^{(2)}(t))^2 dt + \sum_{i=1}^N w_i (x(u_i) - d_i)^2,$$

【0040】

を最小にするスプライン曲線 $x(t)$ もしくはは等価的に制御点ベクトル $\mathbf{R}^M$ を見つける問題となる。ただし、 $w_i > 0$ 、 $w_i \in (0, 1)$   $i$ である。

【0041】

この問題に対する最適解は以下の線形方程式の解として得られる。

30

【0042】

【数8】

$$(\lambda Q + BWB^T)\tau = BWd$$

【0043】

ここで、 $Q \in \mathbf{R}^M \times \mathbf{R}^M$  は

40

【0044】

【数9】

$$Q = \int_{t_0}^{t_m} \frac{d^2 b(t) d^2 b^T(t)}{dt^2 dt^2} dt$$

【0045】

によって定義されるグラミアンで、与えられる離散データ $\mathcal{D}$ とは無関係に予め計算してお

50

くことができる。

【 0 0 4 6 】

また  $W \in \mathbb{R}^N \times \mathbb{N}$ 、 $d \in \mathbb{R}^N$  および  $B \in \mathbb{R}^M \times \mathbb{N}$  は

【 0 0 4 7 】

【 数 1 0 】

$$W = \text{diag}\{w_1, w_2, \dots, w_N\}$$

10

【 0 0 4 8 】

【 数 1 1 】

$$d = [d_1 \ d_2 \ \dots \ d_N]^T$$

【 0 0 4 9 】

【 数 1 2 】

$$B = [b(u_1) \ b(u_2) \ \dots \ b(u_N)]$$

20

【 0 0 5 0 】

である。

【 0 0 5 1 】

ただし数式 9 , 1 2 における  $b(t) \in \mathbb{R}^M$  は、

【 0 0 5 2 】

【 数 1 3 】

$$b(t) = [B_k(\alpha(t - t_{-k})) \ \dots \ B_k(\alpha(t - t_{m-1}))]^T$$

30

【 0 0 5 3 】

なるベクトルである。

【 0 0 5 4 】

数式 8 は可解であり、従って解 は常に存在する。もちろん  $Q+BWB^T > 0$  のときに限り唯一となる。この解 を数式 1 に用いることによって、区間  $[t_0, t_m]$  における  $k$  次の最適平滑化スプライン  $x(t)$  が設計される。

【 0 0 5 5 】

このように、制御点ベクトル に関する最適化解を求め、数式 1 を用いることによって、 $k$  次最適平滑化スプライン曲線  $x(t)$  が生成される。

40

【 0 0 5 6 】

(スプライン曲線の極値)

次に、最適平滑化スプライン曲線  $x(t)$  の極値を検出する方法を説明する。具体的には、 $x^{(1)}(t) = 0$  と  $x^{(2)}(t) > 0$  もしくは  $x^{(2)}(t) < 0$  を満たしている点  $t \in (t_0, t_m)$  を見つければ良い。

【 0 0 5 7 】

区間  $[t_0, t_m]$  で設計された  $k$  次最適平滑化スプライン  $x(t)$  は、各区間  $[t_j, t_{j+1})$  ( $k=0, 1, 2, \dots, m-1$ ) で各々  $k$  次多項式である。従って、区間毎に極値を検出することによって、 $x(t)$  の最大値、最小値を含むすべての極値が求まる。

50

【 0 0 5 8 】

$x(t)$ は区間 $[t_j, t_{j+1})$ において、

【 0 0 5 9 】

【 数 1 4 】

$$x(t) = \sum_{i=0}^k \tau_{j-k+i} N_{i,k}(\alpha(t - t_j))$$

10

【 0 0 6 0 】

となり、 $k + 1$ 個の制御点  $\tau_{j-k}$ 、 $\tau_{j-k+1}$ 、 $\dots$ 、 $\tau_j$ によって決まる $k$ 次多項式である。

【 0 0 6 1 】

$t$ から  $\delta$  への変数変換を次のように行う。

【 0 0 6 2 】

【 数 1 5 】

$$\delta = \alpha(t - t_j)$$

20

【 0 0 6 3 】

このとき変数  $t$  の区間 $[t_j, t_{j+1})$ は変数  $\delta$  に対する単位区間 $[0, 1)$ に正規化され、関数  $x(t)$ は  $\delta$  の関数

【 0 0 6 4 】

【 数 1 6 】

$$\hat{x}(\delta) = \sum_{i=0}^k \tau_{j-k+i} N_{i,k}(\delta)$$

30

【 0 0 6 5 】

に変換される。 $\hat{x}(\delta)$ は  $\delta$  に関する $k$ 次多項式であり、これを

【 0 0 6 6 】

【 数 1 7 】

$$\hat{x}(\delta) = a_0 \delta^k + a_1 \delta^{k-1} + \dots + a_k$$

40

【 0 0 6 7 】

とする。

【 0 0 6 8 】

なお、以下に使用される数式に関して、 $\delta$  を変数とする関数 $x$ には、すべて‘ハット (^)’がつくものであるが、明細書本文中では省略するものとする。

50

【0069】

明らかに $x(t) = x(\dots)$ 、また $x^{(1)}(t) = x^{(1)}(\dots)$ 、 $x^{(2)}(t) = x^{(2)}(\dots)$ かつ  $\dots > 0$ より、 $x(t)$ の区間 $[t_j, t_{j+1})$ における極値は $x(\dots)$ の単位区間 $[0, 1)$ における極値として得ることができる。 $x(\dots)$ が  $\dots = \dots_i (0 \dots_i < 1)$ で極値 $x(\dots_i)$ を持てば $x(t)$ は  $t = (1 / \dots)_i + t_j$ で極値 $x(t) = x(\dots_i)$ を持つ。

【0070】

$x(\dots)$ の単位区間 $[0, 1)$ における極値の求め方は、後述する図3のステップ103～ステップ105の通りである。その際、1次導関数 $x^{(1)}(\dots)$ の根の計算は、既存の多項式の根の計算プログラムを用いることができる。

【0071】

10

(3次スプライン)

実際に良く用いられている3次スプライン(すなわち $k = 3$ )の場合には、以下の仮定1を置くことによって、上のステップはより詳細なアルゴリズムとして書くことができる。これは $x^{(1)}(\dots)$ が2次多項式となり、その根が陽的に表現できるからである。

【0072】

ここでは、実用的な目的上、次の仮定を導入する。

【0073】

(仮定1) 各節点区間 $[t_j, t_{j+1})$ の $x(t)$ の極値の個数は、多くても1つである。ただし、 $j=0, 1, 2, \dots, m-1$

【0074】

20

$x(t)$ が、最適平滑化スプライン曲線であることに注意すると、節点区間 $[t_j, t_{j+1})$ が小さければ仮定1は妥当なものである。

【0075】

3次スプライン(すなわち $k = 3$ )の場合には、多項式 $x(\dots)$ は4つの重み係数  $\dots_{j-3}, \dots_{j-2}, \dots_{j-1}, \dots_j$ だけに依存し、 $\dots$ の変域 $[0, 1)$ に対して次式のように表現することができる。

【0076】

【数18】

$$\hat{x}(\delta) = \frac{1}{6}(p_j\delta^3 + 3q_j\delta^2 + 3r_j\delta + s_j), \quad \delta \in [0, 1),$$

30

ただし、

$$p_j = \tau_j - 3\tau_{j-1} + 3\tau_{j-2} - \tau_{j-3},$$

$$q_j = \tau_{j-1} - 2\tau_{j-2} + \tau_{j-3},$$

$$r_j = \tau_{j-1} - \tau_{j-3},$$

$$s_j = \tau_{j-1} + 4\tau_{j-2} + \tau_{j-3}.$$

40

【0077】

すなわち、係数 $p_j, q_j, r_j, s_j$ は、上記最適化過程ですでに得られている重み係数(ベクトル)  $\dots$ のうち、 $\dots_{j-3}, \dots_{j-2}, \dots_{j-1}, \dots_j$ の4つから唯一に決定される。

【0078】

前述のように、 $x(\dots)$ が  $\dots = \dots_e$ 。  $[0, 1)$ で極値をもつことがわかれば、そのとき $x(t)$ は、 $t_e = t_j + \dots_e(1 / \dots)$   $[t_j, t_{j+1})$ で極値をもち、その値は、 $x(t_e) = x(\dots_e)$ により得られることとなる。

【0079】

$\dots_e$ を得るためには、数式18の $x(\dots)$ の1次導関数、すなわち

50

【数 1 9】

$$\hat{x}^{(1)}(\delta) = \frac{1}{2}(p_j\delta^2 + 2q_j\delta + r_j)$$

が重要な役割をはたす。  $p_j \neq 0$  である場合には、  $x^{(1)}(\delta)$  の2つの根は、

【 0 0 8 0】

【数 2 0】

10

$$\delta_+ = \frac{-q_j + \sqrt{q_j^2 - p_j r_j}}{p_j}, \quad \delta_- = \frac{-q_j - \sqrt{q_j^2 - p_j r_j}}{p_j}.$$

のように表すことができる。そのとき、  $x^{(1)}(0) = r_j/2$  および  $x^{(1)}(1) = r_{j+1}/2$  となるので、極値が存在するかどうかは  $r_j$  と  $r_{j+1}$  の符号により調べることができる。具体的には、仮定 1 のもとでの幾何学的な観察により、以下の3つの場合が得られる。

20

【 0 0 8 1】

(P1)  $r_j \cdot r_{j+1} < 0$  であれば、極値は  $[0, 1)$  に存在する。

【 0 0 8 2】

(P2)  $r_j \cdot r_{j+1} = 0$  であれば、極値はいくつかの付加的条件のもとで  $[0, 1)$  に存在する。

【 0 0 8 3】

(P3)  $r_j \cdot r_{j+1} > 0$  であれば、極値は  $[0, 1)$  には存在しない。

【 0 0 8 4】

(P1) と (P2) の場合について、以下の結果が得られる。

【 0 0 8 5】

(命題 1)  $r_j \cdot r_{j+1} < 0$  と仮定する。そのとき、  $r_j > 0$  かつ  $r_{j+1} < 0$  ( $r_j < 0$  かつ  $r_{j+1} > 0$ ) ならば、関数  $x(\delta)$  は  $\delta = \delta_-$  ( $\delta = \delta_+$ ) で極大(極小)をもつ。ただし、

30

【 0 0 8 6】

【数 2 1】

$$\delta_e = \begin{cases} \delta_- \text{ (resp., } \delta_+) & \text{if } p_j \neq 0 \\ -\frac{r_j}{2q_j} & \text{if } p_j = 0 \end{cases}$$

40

【 0 0 8 7】

(命題 2)  $r_j \cdot r_{j+1} = 0$  と仮定するとき、以下の 2 つの場合が生じる。

【 0 0 8 8】

(i)  $r_j = 0$  の場合:  $q_j < 0$  ( $q_j > 0$ ) であれば、そのとき  $x(\delta)$  は  $\delta = 0$  で極大(極小)をもつ。  $q_j = 0$  である場合には、極値は存在しない。

【 0 0 8 9】

(ii)  $r_j \neq 0$  かつ  $r_{j+1} = 0$  の場合:  $r_j > 0$  かつ  $q_{j+1} > 0$  ( $r_j < 0$  かつ  $q_{j+1} < 0$ ) であれば、そのとき  $x(\delta)$  は  $\delta = r_j/p_j$  ( $\delta = 0$ ) で極大(極小)をもつ。また、  $r_j \cdot q_{j+1} < 0$  であれば極値は存在しない。

50

い。

【0090】

したがって、3次最適平滑化スプライン関数x(t)の極値を検出し計算するアルゴリズムは、以下のようにまとめることができる。

【0091】

重み係数  $w_{-3}, w_{-2}, w_{-1}, \dots, w_{m-1}$  が与えられ、 $r_j$  と  $r_{j+1}$  を、 $j=0, 1, 2, \dots, m-1$  に対して計算し、そのとき  $r_j \cdot r_{j+1} < 0$  および  $r_j \cdot r_{j+1} = 0$  の場合には、それぞれ命題1および2を使う。  $t_e$  で極値  $x(t_e)$  が見つければ、 $t_e = t_j + (1/r_j)$  で極値  $x(t_e)$  を計算する。

【0092】

図2は、本実施形態のk次最適平滑化スプライン曲線による極値検出方法及び極値検出プログラムに用いられる処理装置を示すブロック図である。本実施形態の最適平滑化スプラインによる極値検出方法に用いられる処理装置1は、CPU5(演算手段)とメモリ装置4(RAMやROM等:記憶手段)を有し、このメモリ装置4に格納されたプログラムを実行し、図3、4のフローチャートに記載した動作を実行することにより、図2のブロック図に示すように、CPU5と、メモリ装置4とマウス、キーボード、スキャナ、データロガー等の入力装置6(入力手段)とバス3と、ディスプレイ装置7(出力手段)と、プリンタ装置8(出力手段)と、外部ネットワーク10と接続するネットワーク装置9(入力手段・出力手段)を備えた処理装置として機能する。

10

【0093】

図3は、本実施形態の一般のk次最適平滑化スプラインによる極値検出方法の処理の流れを示すフローチャートである。

20

【0094】

ステップ101(図ではステップをSと略す)は、離散データDについて重み係数  $w_i$  を求め入力する工程である。すなわち、CPU5は、入力した離散データ  $D(u_i, d_i)$  ,  $i=1, \dots, N$  について、最適な重み係数  $w_i$  ,  $i = -k, -k+1, \dots, m-1$  を決定し、最適平滑化スプライン曲線  $x(t)$  を設計する。

【0095】

ステップ102では、CPU5は、節点区間  $[t_j, t_{j+1})$  での最適平滑化スプライン曲線  $x(t)$  を正規化して得られる、単位区間  $[0, 1)$  でのk次多項式  $x(\delta)$  を生成する。すなわち、 $x(\delta)$  の各係数  $a_0, \dots, a_k$  の計算を行う。

30

【0096】

$x(\delta)$  は、数式16で定義されている。ここで、基底関数(多項式)  $N_{0,k}(\delta)$  ,  $N_{1,k}(\delta)$  ,  $\dots$  ,  $N_{h,k}(\delta)$  は、数式3の逐次式から計算機を用いて予め求めておくことができ、いずれもk次多項式である。いま、

【0097】

【数22】

$$N_{i,k}(\delta) = a_{i,0}\delta^k + a_{i,1}\delta^{k-1} + \dots + a_{i,k}, \quad i = 0, 1, \dots, k$$

40

【0098】

とし、求める  $x(\delta)$  の多項式表現を数式17とすると、その係数  $a_0, \dots, a_k$  は数式17、数式22によって次式から求められる。

【0099】



【数 2 3】

$$\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} a_{0,0} & a_{1,0} & \cdots & a_{k,0} \\ a_{0,1} & a_{1,1} & \cdots & a_{k,1} \\ \vdots & \vdots & & \vdots \\ a_{0,k} & a_{1,k} & \cdots & a_{k,k} \end{bmatrix} \begin{bmatrix} \tau_{j-k} \\ \tau_{j-k+1} \\ \vdots \\ \tau_j \end{bmatrix}$$

10

【0 1 0 0】

ステップ 1 0 3 では、CPU 5 は、ステップ 1 0 2 で求めた多項式  $x(\ )$  の 1 次導関数  $x^{(1)}(\ )$  の根を求める。

【0 1 0 1】

ステップ 1 0 4 では、CPU 5 は、多項式  $x(\ )$  が単位区間  $[0, 1)$  で実根を有するか否か判定する。実根を有する場合にはステップ 1 0 5 へ進み、実根を有しない場合にはステップ 1 0 6 へ進む。

【0 1 0 2】

ステップ 1 0 5 では、CPU 5 は、 $x(\ )$  の 2 次導関数  $x^{(2)}(\ )$  へ、ステップ 1 0 3 で求めた多項式  $x(\ )$  の実根  $e$  を代入し、 $x^{(2)}(e) > 0$  の場合には、 $t_e = t_j + (1/ \ )_e$  で極小値  $x(\ e)$  をもつものとし、 $x^{(2)}(e) < 0$  の場合には、 $t_e = t_j + (1/ \ )_e$  で極大値  $x(\ e)$  をもつものとする。これにより、最適平滑化スプライン関数  $x(t)$  の所定の節点区間  $[t_j, t_{j+1})$  における、極大値または極小値が求まることとなる。

20

【0 1 0 3】

ステップ 1 0 6 では、 $j$  が、 $m - 1$  より小さい場合にはステップ 1 0 2 へ戻り、 $j$  が、 $m - 1$  の場合にはステップ 1 0 7 へ進み、区間  $[t_0, t_m]$  のすべての節点区間  $[t_j, t_{j+1})$  についての極値検出手続きが終了するまで、ステップ 1 0 2 からステップ 1 0 5 を繰り返す。

【0 1 0 4】

ステップ 1 0 7 では、CPU 5 は、最適平滑化スプライン関数  $x(t)$  の全区間  $[t_0, t_m]$  に存在するすべての極値から、すべての極値の大小を判別することにより、最適平滑化スプライン関数  $x(t)$  の最大値及び最小値を求める。そして、最適平滑化スプライン関数  $x(t)$  の極大値、極小値、最大値及び最小値を、ディスプレイ装置 7 やプリンタ装置 8 に出力する。

30

【0 1 0 5】

図 4 は、本実施形態の最適平滑化スプラインによる極値検出方法の処理の流れにおいて、3 次スプライン ( $k = 3$ ) を用いた場合を示すフローチャートである。なお、図 3 のフローチャートとはステップ 1 0 2 ~ 1 0 4 のみ相違するため、対応する部分を、ステップ 1 0 2 a 及び 1 0 3 a として記載している。

40

【0 1 0 6】

ステップ 1 0 2 a は、図 3 のフローチャートのステップ 1 0 2 に対応する部分である。ステップ 1 0 2 a では、CPU 5 は、3 次多項式である  $x(\ )$  について、各係数  $p_j$ 、 $q_j$ 、 $r_j$ 、 $s_j$  を算出する。上述したように、各係数  $p_j$ 、 $q_j$ 、 $r_j$ 、 $s_j$  は、 $j - 3$ 、 $j - 2$ 、 $j - 1$ 、 $j$  の 4 つから唯一に決定される。

【0 1 0 7】

ステップ 1 0 3 a は、図 3 のフローチャートのステップ 1 0 3 及び 1 0 4 に対応する部分である。ステップ 1 0 3 a では、CPU 5 は、ステップ 1 0 2 a で求めた多項式  $x(\ )$  の 1 次導関数  $x^{(1)}(\ )$  の根を求める。 $q_j^2 - p_j r_j = 0$  の場合には図 3 のステップ 1 0 5 へ進み、 $q_j^2 - p_j r_j < 0$  の場合には図 3 のステップ 1 0 6 へ進む。

50

## 【 0 1 0 8 】

なお、本実施形態においては、離散データDが与えられた場合について極値をもとめた  
が、関数 $f(t)$ 、 $t_0$ 、 $t_m$ が入力データとして与えられた場合についても、同様な方法で  
極値を求めることが可能である。この場合にも、まず $f(t)$ を近似する最適スプラインを以  
下のように設計する。

## 【 0 1 0 9 】

関数 $f(t)$ が与えられたとき、評価関数は、数式7の代わりに

## 【 0 1 1 0 】

## 【数24】

$$J_c(\tau) = \lambda \int_{t_0}^{t_m} \left( x^{(2)}(t) \right)^2 dt + \int_{t_0}^{t_m} (x(t) - f(t))^2 dt$$

10

## 【 0 1 1 1 】

を用いる。この最適解 は線形方程式

## 【 0 1 1 2 】

## 【数25】

$$(\lambda Q + Q_0)\tau = \int_{t_0}^{t_m} b(t)f(t)dt.$$

20

## 【 0 1 1 3 】

の解として得られる。ここで、 $Q$ は、数式9で定義される行列、 $Q_0$ は

## 【 0 1 1 4 】

## 【数26】

$$Q_0 = \int_{t_0}^{t_m} b(t)b^T(t)dt$$

30

## 【 0 1 1 5 】

で定義される行列で、いずれも $f(t)$ とは無関係に予め計算しておくことができる。右辺は  
、 $f(t)$ が与えられると適当な数値積分法を用いて計算できる。

## 【 0 1 1 6 】

一旦、最適なベクトル  $\tau$  が決まると、極値の検出等、後の扱いは離散データの場合と同  
様である。ただし、既存の最適化アルゴリズムとの併用によって以下のような有効性があ  
る。一般に関数 $f(t)$ が与えられたとき、ある区間（一般性を失わずに $[t_0, t_m]$ と考  
えてよい）での最大値や最小値を求める方法は繰り返しによる数値的探索法によるのが通常  
である。

40

## 【 0 1 1 7 】

その際、繰り返し式の初期値の取り方が問題となり、それによって多峰性関数の場合は  
到達する極値が異なる。最大値や最小値が得られる保証は全くない。本方法によれば、 $f(t)$   
を近似する最適スプラインが得られ、その最大値や最小値が得られる。さらに対応する  
 $t$ の値を初期値として数値的探索法を適用すれば $f(t)$ の最大値や最小値をより正確に求め  
ることができる。

## 【 0 1 1 8 】

## (数値実験例)

図5は、本発明の第1実施形態の最適平滑化スプライン曲線とその極値を示す図である  
。図5は、最適平滑化スプライン曲線 $x(t)$ （図中の実線で示された関数）から極値を抽出

50

する性能を示す図である。図5は、 $k=3$ 、 $\alpha=1$ 、 $t_0=1$ および $t_m=m=50$ の場合での結果を示している。ただし、三角形と逆三角形はそれぞれ検出された極大値および極小値を示す。ここで、データ $d_i$  (図中の\*印)は、関数 $f(t) = e^{a(t-1)} \cos(b(t-1)) + 1$  (図中の点線で示された関数)をサンプリングすることにより生成されている。ただし、 $a=1/(m-1) \log 1/4$ 、 $b=2 \times 3/(m-1)$ である。データの個数は $N=30$ とし、 $u_i$ は区間 $[t_0, t_m]=(1, 50)$ においてランダムに配置されたデータ点であり、またデータに加えるノイズの大きさは $\sigma=0.02$ とする。さらに、平滑化パラメータ $\delta$ を推定するためにいわゆるcross-validation手法を採用し、その最適値は $\delta=0.1585$ のように得られた。

【0119】

図からわかるように、データノイズがかなり大きいにもかかわらず、元の関数である $f(t)$ のすべての極値がかなり精度よく検出されていることがわかる。特に極値を与える $t$ の値はほぼ正確に得られていることがわかる。

10

【0120】

なお、上記実施形態では、最適平滑化スプライン関数 $x(t)$ の極値を求めたが、同様な方法で、 $x(t)$ の第 $i$ 導関数 $x^{(i)}(t)$ の極値を求めることが可能である。

【0121】

区間 $[t_1, t_m]$ で設計された最適スプライン $x(t)$ の第 $i$ 導関数 $x^{(i)}(t)$  ( $i=1, 2, \dots, k$ )も各節点区間 $[t_j, t_{j+1})$  ( $k=0, 1, \dots, m-1$ )毎に考えれば良い。すなわち、節点区間 $[t_j, t_{j+1})$ においては数式14と数式15の変数変換によって、 $x^{(i)}(t) = \delta^i x^{(i)}(\delta)$ の関係が成立する。また $\delta > 0$ より $x^{(i)}(t)$ の節点区間 $[t_j, t_{j+1})$ における極値は、 $x^{(i)}(\delta)$ の単位区間 $[0, 1)$ における極値となる。この第 $i$ 導関数 $x^{(i)}(\delta)$ は、数式17と数式23の結果を用いて、

20

【0122】

【数27】

$$\hat{x}^{(i)}(\delta) = \frac{d^i}{d\delta^i} (a_0 \delta^k + a_1 \delta^{k-1} + \dots + a_k)$$

【0123】

によって計算できる。

30

【0124】

$x^{(i)}(\delta)$  (ただし $i \leq k-2$ )の極値の検出は、図3のステップ102における多項式 $x(\delta)$ を $x^{(i)}(\delta)$ に置き換え、これに伴い $x^{(1)}(\delta)$ 、 $x^{(2)}(\delta)$ をそれぞれ $x^{(i+1)}(\delta)$ 、 $x^{(i+2)}(\delta)$ に置き換えて、ステップ103以降に同様な方法を適用すれば良い。ここで、 $i \leq k-2$ に限定されるのは、 $k-(i+2)$ 次の多項式である $i+2$ 次導関数が必要であるからで、 $k-(i+2) \geq 0$ が必要なためである。

【0125】

以上説明したように、本実施形態の最適平滑化スプラインによる極値検出方法によれば、設計されたスプラインを対象とするため、各節点区間ごとに定義された多項式の極値(極小値、極大値)を求めれば良い。これによりすべての極値が求まり、最小値や最大値を求めることが可能となる。

40

【0126】

また、基底関数としてBスプライン関数を用いているため、設計されたスプラインの表現が簡潔であり、その導関数も容易に求めることができる。

【0127】

そして、極値計算の対象は、離散データでも関数でも良い。いずれの場合にも最適平滑化スプラインが設計でき、その際、節点の間隔やBスプラインの次数を調整することによって十分な精度で近似でき、ノイズの影響を受けにくい数値的に非常に安定した方法といえる。もちろん、この極値検出法は最適平滑化スプライン曲線に限らず、数式1で表現さ

50

れるスプライン曲線であれば適用できるものである。

【 0 1 2 8 】

特に、3次Bスプライン関数を用いた場合には、各区間毎に3次多項式となり、極値は解析的に得ることができる。

【 0 1 2 9 】

(第2実施形態)

以下、本発明の第2実施形態である最適平滑化スプライン曲面による極値検出方法及び極値検出プログラムについて、図を参照して詳細に説明をする。

【 0 1 3 0 】

(最適平滑化スプライン曲面)

本発明の第1実施形態では、平面における離散データについて最適平滑化スプライン曲線を生成し極値の検出を行ったが、本実施形態は、空間における離散データを近似する最適平滑化スプライン曲面を生成し極値の検出を行うものである。また、本実施形態は、本発明の第1実施形態の最適平滑化スプライン曲線の極値検出方法を最適平滑化スプライン曲面に素直に拡張した方法を提供するものである。

【 0 1 3 1 】

図6は、本実施形態の最適平滑化スプライン曲面を概略的に示す図である。図6Aは、空間における離散データから求められた最適な制御点(重み係数)の分布を示す図であり、図6Bは、空間における離散データに基づいて生成された最適平滑化スプライン曲面 $x(s, t)$ を示す図である。

【 0 1 3 2 】

(曲面の生成)

本実施形態の最適平滑化スプライン曲面 $x(s, t)$ は、 $k$ 次Bスプライン関数 $B_k(\cdot)$ を用いて、数式1の自然な拡張である以下の数式28で表すことが可能である。

【 0 1 3 3 】

【数28】

$$x(s, t) = \sum_{i=-k}^{m_1-1} \sum_{j=-k}^{m_2-1} \tau_{i,j} B_k(\alpha(s - s_i)) B_k(\beta(t - t_j))$$

30

【 0 1 3 4 】

ここで、 $\alpha, \beta (>0)$  は定数、 $m_1, m_2 (>2)$  は整数、 $s_i, t_j$  は以下のように等間隔に配置された節点を表す。

【 0 1 3 5 】

【数29】

$$s_{i+1} - s_i = \frac{1}{\alpha}, \quad t_{j+1} - t_j = \frac{1}{\beta}$$

40

【 0 1 3 6 】

このとき、制御点  $s_i, t_j$  を適切に選ぶことによって、任意の  $k$  次スプライン曲面を  $(s, t)$  の領域

【 0 1 3 7 】

【数30】

$$S = [s_0, s_{m_1}] \times [t_0, t_{m_2}] \subset \mathbf{R}^2$$

50

【 0 1 3 8 】

上で設計することができる。この制御点をまとめて行列  $R^{M_1 \times M_2}$  (ただし  $M_1 = m_1 + k$ 、 $M_2 = m_2 + k$ ) で表すことにする。

【 0 1 3 9 】

【数 3 1】

$$\tau = \begin{bmatrix} \tau_{-k,-k} & \tau_{-k,-k+1} & \cdots & \tau_{-k,m_2-1} \\ \tau_{-k+1,-k} & \tau_{-k+1,-k+1} & \cdots & \tau_{-k+1,m_2-1} \\ \vdots & \vdots & & \vdots \\ \tau_{m_1-1,-k} & \tau_{m_1-1,-k+1} & \cdots & \tau_{m_1-1,m_2-1} \end{bmatrix} \quad 10$$

【 0 1 4 0 】

いま、空間内に離散データが

【 0 1 4 1 】

【数 3 2】

$$\mathcal{D} = \{(u_i, v_i; d_i) : (u_i, v_i) \in \mathcal{S}, d_i \in \mathbf{R}, i = 1, 2, \dots, N\}$$

20

【 0 1 4 2 】

のように与えられたとする。このとき最適平滑化スプライン曲面を、下記の評価関数  $J(\tau)$  が最小となるように設計する。

【 0 1 4 3 】

【数 3 3】

$$J(\tau) = \lambda \int_{I_1} \int_{I_2} (\nabla^2 x(s, t))^2 ds dt + \sum_{i=1}^N w_i (x(u_i, v_i) - d_i)^2,$$

30

ただし、積分区間は  $I_1 = [s_0, s_{m_1}]$ 、 $I_2 = [t_0, t_{m_2}]$ 、また、 $\nabla^2$  はラプラシアン、す

$\nabla^2 = \frac{\partial^2}{\partial s^2} + \frac{\partial^2}{\partial t^2}$  である。一方、 $\lambda (> 0)$  は平滑化パラメータ、 $w_i (0 < w_i \leq 1)$  は誤する重みを表すパラメータで、いずれも設計パラメータである。

【 0 1 4 4 】

この問題に対する解、すなわち最適な制御点は次の線形方程式の解として得られる。

【 0 1 4 5 】

【数 3 4】

$$(\lambda Q + B W B^T) \hat{\tau} = B W d$$

40

ただし、 $\hat{\tau} = \text{vec } \tau$  (行列  $\tau$  の各列を順に繋げてできる  $M_1 M_2$  次元のベクトル) であり、 $\otimes$  はクロネッカ積を表す。

【 0 1 4 6 】

以下で数式 3 4 において現れる各行列、ベクトルを示す。まず、 $Q$  は次式で与えられる

50

$M_1 M_2 \times M_1 M_2$  の行列である。

【 0 1 4 7 】

【数 3 5】

$$Q = Q_2^{(00)} \otimes Q_1^{(22)} + Q_2^{(02)} \otimes (Q_1^{(02)})^T + (Q_2^{(02)})^T \otimes Q_1^{(02)} + Q_2^{(22)} \otimes Q_1^{(00)},$$

【 0 1 4 8 】

ここで、 $Q_l^{(ij)} \in \mathbb{R}^{M^1 \times M^2}$  は

10

【 0 1 4 9 】

【数 3 6】

$$Q_l^{(ij)} = \int_{I_l} \frac{d^i b_l(t) d^j b_l^T(t)}{dt^i dt^j} dt, \quad l = 1, 2, \quad ; \quad i, j = 0, 1, 2$$

【 0 1 5 0 】

によって定義される。ただし、ベクトル  $b_1(t) \in \mathbb{R}^{M^1}$ 、 $b_2(t) \in \mathbb{R}^{M^2}$  は、Bスプライン関数

20

から構成されている。

【 0 1 5 1 】

【数 3 7】

$$b_1(s) = [B_k(\alpha(s - s_{-k})) \ B_k(\alpha(s - s_{-k+1})) \ \cdots \ B_k(\alpha(s - s_{m_1-1}))]^T$$

$$b_2(t) = [B_k(\beta(t - t_{-k})) \ B_k(\beta(t - t_{-k+1})) \ \cdots \ B_k(\beta(t - t_{m_2-1}))]^T.$$

【 0 1 5 2 】

数式 3 6 に示す各行列、従って  $Q$  は与えられるデータとは独立に Bスプラインから予め計

30

算しておくことができる。

【 0 1 5 3 】

行列  $W \in \mathbb{R}^{N \times N}$  は、数式 3 3 に示す評価関数における重みから構成される対角行列である

【 0 1 5 4 】

【数 3 8】

$$W = \text{diag}\{w_1, w_2, \cdots, w_N\}$$

40

【 0 1 5 5 】

行列  $B \in \mathbb{R}^{M^1 M^2 \times N}$ 、およびベクトル  $d \in \mathbb{R}^N$  は与えられるデータによって以下のように定められる。

【 0 1 5 6 】

【数 3 9】

$$B = [b_2(v_1) \otimes b_1(u_1) \quad b_2(v_2) \otimes b_1(u_2) \quad \cdots \quad b_2(v_N) \otimes b_1(u_N)],$$

$$d = [d_1, d_2, \cdots, d_N]^T.$$

【0 1 5 7】

以上によって、数式 3 4 に現れる各行列、ベクトルが定まる。この方程式を解くことによって最適な、従って最適な制御点行列が求められ、最適平滑化曲面  $x(s, t)$  が数式 2 8 によって構成できる。

10

【0 1 5 8】

(曲面の極値)

次に、上記の方法により求めた最適平滑化曲面  $x(s, t)$  の極値を求める方法について説明する。

【0 1 5 9】

(対象とする曲面)

数式 2 8 の  $k$  次スプライン曲面の極値を求める。ここでパラメータ  $k$ 、 $m_1$ 、 $m_2$  は設定済みで、制御点  $i, j$  は前述の方法等によってすでに与えられているとする。なお、 $(s, t)$  の定義域は  $S = [s_0, s_{m_1}] \times [t_0, t_{m_2}] \subset R^2$  である。

20

【0 1 6 0】

(極値)

曲面  $x(s, t)$  の極値は

【0 1 6 1】

【数 4 0】

$$\nabla x(s, t) = 0, \text{ かつ } \nabla^2 x(s, t) > 0 \text{ または } \nabla^2 x(s, t) < 0$$

ここで  $\nabla x(s, t)$ 、 $\nabla^2 x(s, t)$  はそれぞれ  $x(s, t)$  の勾配ベクトル、ヘッセ行列を表す。

30

【0 1 6 2】

をみたす点  $(s, t) \in S$  として得られる。

【0 1 6 3】

曲面  $x(s, t)$  のヘッセ行列が正定値となる場合が極小値、曲面  $x(s, t)$  のヘッセ行列が負定値となる場合が極大値を与える。 $x(s, t)$  の領域  $S$  での極値を求める問題は、節点間の部分領域  $S_{i, \mu} = [s_i, s_{i+1}] \times [t_\mu, t_{\mu+1}]$  における極値を求める問題を全領域  $i=0, 1, \cdots, m_1-1; \mu=0, 1, \cdots, m_2-1$  に渡って繰り返せば良い。

【0 1 6 4】

(部分領域の正規化)

領域  $S_{i, \mu}$  に注目し、 $u = (s - s_i)$ 、 $v = (t - t_\mu)$  なる変数変換を行うと、 $u, v$  の変域はいずれも  $[0, 1]$  と正規化される。すなわち、 $(s, t)$  の領域  $S_{i, \mu}$  は、 $u, v$  にかかわらず、 $(u, v)$  の単位領域  $[0, 1] \times [0, 1]$  になる。このとき、 $S_{i, \mu}$  における関数  $x(s, t)$  は、

40

【0 1 6 5】

【数 4 1】

$$\hat{x}(u, v) = \sum_{i=0}^k \sum_{j=0}^k \tau_{\kappa-k+i, \mu-k+j} N_{i, k}(u) N_{j, k}(v)$$

50

【 0 1 6 6 】

と表すことができる。なお、以下に使用される数式に関して、 $u$ 、 $v$ を変数とする関数 $x$ には、すべて‘ハット( ^ )’がつくものであるが、明細書本文中では省略するものとする。

【 0 1 6 7 】

さらに $x(s, t)$ の偏導関数と $x(u, v)$ の偏導関数の間の関係は

【 0 1 6 8 】

【数 4 2 】

$$\frac{\partial^{i+j}}{\partial s^i \partial t^j} x(s, t) = \alpha^i \beta^j \frac{\partial^{i+j}}{\partial u^i \partial v^j} \hat{x}(u, v), \quad i, j = 0, 1, \dots$$

10

【 0 1 6 9 】

となる。 $\alpha, \beta > 0$ より、 $x(s, t)$ の領域 $S_{s, t}$ 上での極値は $x(u, v)$ の領域 $S_{u, v}$ 上での極値と一致する。

【 0 1 7 0 】

(部分領域での極値)

数式 4 0 に示す、曲面 $x(s, t)$ の勾配ベクトルが 0、かつ、曲面 $x(s, t)$ のヘッセ行列が正定値となるか、又は、曲面 $x(s, t)$ のヘッセ行列が負定値となる点 $(u^*, v^*)$ を検出すればよい。このとき $x(s, t)$ は、点 $(s^*, t^*) = ((1/\alpha) u^* + s_0, (1/\beta) v^* + t_0)$ で極値 $x(s^*, t^*) = x(u^*, v^*)$ をもつ。

20

【 0 1 7 1 】

( $x(u, v)$ の極値の求め方)

(関数 $x(u, v)$ の表現)

$x(u, v)$ は $u, v$ についてそれぞれ $k$ 次の2変数多項式であり、以下のように簡潔に表すことができる。

【 0 1 7 2 】

【数 4 3 】

30

$$\hat{x}(u, v) = h_{k+1}^T(u) A h_{k+1}(v)$$

ただし、 $h_{k+1}(\zeta) = [\zeta^k \zeta^{k-1} \dots 1]^T$ である。

【 0 1 7 3 】

$(k+1) \times (k+1)$ 行列 $A$ は

40

【 0 1 7 4 】

【数 4 4 】

$$A = S^T T_{\kappa\mu} S$$

【 0 1 7 5 】

で定義される。また、 $S, T_{\kappa\mu}$ は次のようにして得られる $(k+1) \times (k+1)$ 行列である。

【 0 1 7 6 】

行列 $S$ は、その各行がBスプラインの基底関数 $N_{i, k}(t)$ ,  $i=0, 1, \dots, k$ の各係数が

50



ら構成される行列である。すなわち、

【 0 1 7 7 】

【数 4 5】

$$N_k(t) = [N_{0,k}(t) \quad N_{1,k}(t) \quad \cdots \quad N_{k,k}(t)]^T$$

10

【 0 1 7 8 】

としたとき、 $N_k(t) = Sh_{k+1}(t)$ となる行列Sであり、例えばk=3の場合、数式4よりSはつぎのようになる。

【 0 1 7 9 】

【数 4 6】

$$S = \frac{1}{3!} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 0 & 4 \\ -3 & 3 & 3 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

20

【 0 1 8 0 】

行列 $T_{\kappa, \mu}$ は、曲面 $x(u, v)$ の定義式に現れる $(k+1)^2$ 個の制御点から次のように構成される。

【 0 1 8 1 】

【数 4 7】

$$T_{\kappa\mu} = \begin{bmatrix} \tau_{\kappa-k, \mu-k} & \tau_{\kappa-k, \mu-(k-1)} & \cdots & \tau_{\kappa-k, \mu} \\ \tau_{\kappa-(k-1), \mu-k} & \tau_{\kappa-(k-1), \mu-(k-1)} & \cdots & \tau_{\kappa-(k-1), \mu} \\ \vdots & \vdots & & \vdots \\ \tau_{\kappa, \mu-k} & \tau_{\kappa, \mu-(k-1)} & \cdots & \tau_{\kappa, \mu} \end{bmatrix}$$

30

【 0 1 8 2 】

すなわち、曲面 $x(s, t)$ を構成する全体の制御点行列（数式31）のうちの、要素 $T_{\kappa, \mu}$ から左上の $(k+1) \times (k+1)$ のブロックとして得られる部分行列である。

【 0 1 8 3 】

(勾配ベクトルとヘッセ行列)

$x(u, v)$ の勾配ベクトルおよびヘッセ行列は以下のように表現される。

【 0 1 8 4 】

【数 4 8】

$$\nabla \hat{x}(u, v) = \begin{bmatrix} \hat{x}_u(u, v) \\ \hat{x}_v(u, v) \end{bmatrix} = \begin{bmatrix} h_k^T(u) C_k^T A h_{k+1}(v) \\ h_{k+1}^T(u) A C_k h_k(v) \end{bmatrix}$$

50

【 0 1 8 5 】

【 数 4 9 】

$$\nabla^2 \hat{x}(u, v) = \begin{bmatrix} h_{k-1}^T(u) C_{k-1}^T C_k^T A h_{k+1}(v) & h_k^T(u) C_k^T A C_k h_k(v) \\ h_k^T(u) C_k^T A C_k h_k(v) & h_{k+1}^T(u) A C_k C_{k-1} h_{k-1}(v) \end{bmatrix}$$

【 0 1 8 6 】

10

ただし  $C_k$  は次のように定義される  $(k+1) \times k$  行列である。

【 0 1 8 7 】

【 数 5 0 】

$$C_k = \begin{bmatrix} k & & & \\ & k-1 & & \\ & & \ddots & \\ & & & 1 \\ 0 & \dots & & 0 \end{bmatrix}$$

20

【 0 1 8 8 】

( 極 値 の 候 補 点 )

曲面  $x(u, v)$  の勾配ベクトルが 0 となる点  $(u, v)$  を求める。曲面  $x(u, v)$  の勾配ベクトルの各要素は、次のように与えられる。

【 0 1 8 9 】

【 数 5 1 】

$$\hat{x}_u(u, v) = q_1(v)u^{k-1} + q_2(v)u^{k-2} + \dots + q_k(v)$$

$$\hat{x}_v(u, v) = r_1(v)u^k + r_2(v)u^{k-1} + \dots + r_{k+1}(v)$$

30

【 0 1 9 0 】

ただし、

【 0 1 9 1 】

【 数 5 2 】

$$q(v) = [q_1(v) \ q_2(v) \ \dots \ q_k(v)]^T \text{ および } r(v) = [r_1(v) \ r_2(v) \ \dots \ r_{k+1}(v)]^T$$

40

【 0 1 9 2 】

は次式で定義されている。

【 0 1 9 3 】

【数 5 3】

$$q(v) = C_k^T A h_{k+1}(v), \quad r(v) = A C_k h_k(v),$$

【0 1 9 4】

$x_u(u, v)=0$ 、 $x_v(u, v)=0$ が同時に成り立つためには、これらを $u$ に関する多項式とみなしたときの終結式が0となる必要がある。すなわち、

【0 1 9 5】

10

【数 5 4】

$$|R(v)| = 0$$

【0 1 9 6】

ただし $R(v)$ は以下の $(2k - 1) \times (2k - 1)$ 行列である。

20

【0 1 9 7】

【数 5 5】

$$R(v) = \begin{bmatrix} q_1(v) & q_2(v) & \cdots & q_k(v) & \cdots & \cdots & \cdots \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ r_1(v) & r_2(v) & \cdots & q_1(v) & q_2(v) & \cdots & q_k(v) \\ & \ddots & \ddots & \cdots & r_{k+1}(v) & \ddots & \ddots \\ & & r_1(v) & r_2(v) & \cdots & \cdots & r_{k+1} \end{bmatrix}$$

30

【0 1 9 8】

$|R(v)|$ は $v$ に関する多項式であり、この多項式の根を求め、 $0 < v < 1$ をみたす根を $v=v^*$ とする。なお、このような根がない場合には $x(u, v)$ は、単位領域に極値をもたない。この $v^*$ に対応する $u$ は、 $u$ に関する連立方程式 $x_u(u, v^*)=0$ 、 $x_v(u, v^*)=0$ の解として求められる。具体的には、 $q_i(\text{バー})=q_i(v^*)$ 、 $r_i(\text{バー})=r_i(v^*)$ として得られる2つの多項式

【0 1 9 9】

40

【数 5 6】

$$f_1(u) = \bar{r}_1 u^k + \bar{r}_2 u^{k-1} + \cdots + \bar{r}_{k+1}$$

$$f_2(u) = \bar{q}_1 u^{k-1} + \bar{q}_2 u^{k-2} + \cdots + \bar{q}_k$$

【0 2 0 0】

の共通因子 $f_0(u)$ を求める。その際、例えばユークリッド互除法を適用することができる。なお、 $f_1(u)$ 、 $f_2(u)$ が共通因子をもつことは保証されており、もちろん、 $f_0(u)$ は $u$ に関する次数1以上の多項式である。 $f_0(u)$ の根 $u$ が $0 < u < 1$ をみたすとき、 $u=u^*$ とおく。みた

50

さないときは $x(u, v)$ は領域 に極値をもたない。

【 0 2 0 1 】

( 極値の判定 )

得られた点 $(u^*, v^*)$  が関数 $x(u, v)$ の極値の候補となる点であり、この点に対するヘッセ行列が正定(極小)あるいは負定(極大)の極値条件をみたすかどうか判定すればよい。なお、ヘッセ行列の計算の際、次のように定義される多項式

【 0 2 0 2 】

【数 5 7】

$$\hat{q}(u) = [\hat{q}_1(u) \hat{q}_2(u) \cdots \hat{q}_k(u)]^T,$$

10

【 0 2 0 3 】

【数 5 8】

$$\hat{q}(u) = C_k^T A^T h_{k+1}^T(u)$$

20

【 0 2 0 4 】

を導入し、さらに $q(v)$ 、 $r(v)$ を用いることによって得られる次式を用いるのがよい。

【 0 2 0 5 】

【数 5 9】

$$\nabla^2 \hat{x}(u, v) = \begin{bmatrix} h_{k-1}^T(u) C_{k-1}^T q(v) & h_k^T(u) C_k^T r(v) \\ h_k^T(u) C_k^T r(v) & \hat{q}(u) C_{k-1} h_{k-1}(v) \end{bmatrix}$$

30

【 0 2 0 6 】

図7は、本実施形態の最適平滑化スプラインによる極値検出方法及び極値検出プログラムに用いられる処理装置を示すブロック図である。本実施形態の極値検出処理装置11は、CPU15(演算手段)とメモリ装置14(RAMやROM等:記憶手段)を有し、このメモリ装置14に格納されたプログラムを実行し、図8~11のフローチャートに記載した動作を実行することにより、図7のブロック図に示すように、CPU15(演算手段)と、メモリ装置14(記憶手段)と、マウス、キーボード、スキャナ、データロガー等の入力装置16(入力手段)と、バス13と、ディスプレイ装置17(出力手段)と、プリンタ装置18(出力手段)と、外部ネットワーク20と接続するネットワーク装置19(入力手段・出力手段)を備えた処理装置として機能する。

40

【 0 2 0 7 】

図8は、本実施形態の最適平滑化スプラインによる極値検出方法及び極値検出プログラムの処理の概要を示すフローチャートである。図9は、図8のフローチャートのステップ203の処理の詳細を示すフローチャートであり、図10は、図8のフローチャートのステップ206の処理の詳細を示すフローチャートであり、図11は、図8のフローチャー

50

トのステップ211の処理の詳細を示すフローチャートである。

【0208】

ステップ201では、極値検出処理装置11に曲面の最適設計に用いられたパラメータ、 $m_1$ 、 $m_2$  および得られた最適制御点  $i, j, i = -k, -k+1, \dots, m_1 - 1, j = -k, -k+1, \dots, m_2 - 1$  を入力する。

【0209】

ステップ202では、CPU15は、 $(k+1) \times (k+1)$  行列S、 $(k+1) \times k$  行列 $C_k$  および  $k \times (k-1)$  行列 $C_{k-1}$  を準備する。

【0210】

以下、すべての  $(i, \mu), i = 1, 2, \dots, m_1 - 1, \mu = 1, 2, \dots, m_2 - 1$  の組み合わせに対して、以下の手順に従って領域  $S_{i, \mu}$  での  $x(s, t)$  の極値の検出、計算を行う。

【0211】

CPU15は、 $(k+1) \times (k+1)$  行列A (ステップ203a)、多項式  $q_i(v), i = 1, 2, \dots, k$ 、多項式  $r_i(v), i = 1, 2, \dots, k+1$  (ステップ203b)、および終結式  $|R(v)|$  を求め (ステップ203c)、 $v$  の多項式である終結式の根を求める (ステップ203)。この終結式  $|R(v)|$  の計算には、数式処理を援用することができる。 $0 < v < 1$  をみたく根  $v$  が存在しない場合には、次の  $(i, \mu)$  の組に移る (ステップ204)。

【0212】

CPU15は、得られた  $v (0 < v < 1)$  を  $v^*$  とおき (ステップ205)、さらに、 $q_i(\text{バー}) = q_i(v^*), i = 1, 2, \dots, k$ 、および  $r_i(\text{バー}) = r_i(v^*), i = 1, 2, \dots, k+1$  を計算する。それぞれ  $q_i(\text{バー}), r_i(\text{バー})$  を係数とする  $u$  の多項式  $f_1(u), f_2(u)$  を定め (ステップ206a)、例えば、ユークリッド互助法を適用するなどして、 $f_1(u)$  と  $f_2(u)$  の共通因子  $f_0(u)$  を求める (ステップ206b)。  $f_0(u)$  の根  $u$  (ステップ206c) が、 $u < 1$  をみたくとき (ステップ207)、 $u = u^*$  とおく (ステップ208)。そのような根  $u$  が存在しない場合には、次の  $(i, \mu)$  の組に移る。

【0213】

CPU15は、点  $(u^*, v^*)$  におけるヘッセ行列を計算し (ステップ209)、点  $(u^*, v^*)$  におけるヘッセ行列が正定となる場合には極小、点  $(u^*, v^*)$  におけるヘッセ行列が負定となる場合には極大と判断し (ステップ210)、対応する極値  $x(u^*, v^*)$  を計算する (ステップ211a)。これら両ケース以外の場合は、次の  $(i, \mu)$  の組に移る。

【0214】

CPU15は、 $x(s, t)$  の極値をとる点  $(s^*, t^*)$  を、 $(s^*, t^*) = ((1/\lambda)u^* + s_0, (1/\lambda)v^* + t_0)$  で計算し (ステップ211b)、その値  $x(s^*, t^*) = x(u^*, v^*)$  とともに出力する (ステップ211c)。

【0215】

(2変数関数に対する最適曲面の設計)

なお、本実施形態においては、離散データ  $D$  が与えられた場合について極値をもとめたが、関数  $f(s, t)$  が入力データとして与えられた場合についても、同様な方法で極値を求めることが可能である。関数  $f(s, t)$  が与えられたときの  $k$  次最適スプライン曲面  $x(s, t)$  の設計方法は以下の通りである。評価関数は、数式33の代わりに次式を用いる。

【0216】

【数60】

$$J(\tau) = \lambda \int_{I_1} \int_{I_2} (\nabla^2 x(s, t))^2 ds dt + \int_{s_0}^{s_m} \int_{t_0}^{t_n} (x(s, t) - f(s, t))^2 ds dt.$$

【0217】

この評価関数を最小にする (=vec )は、数式 3 4 の代わりに線形方程式  
 【 0 2 1 8 】  
 【数 6 1 】

$$(\lambda Q + Q_2^{(00)} \otimes Q_1^{(00)}) \hat{\tau} = \int_{I_1} \int_{I_2} f(s, t) (b_2(t) \otimes b_1(s)) ds dt$$

【 0 2 1 9 】 10

を解けばよい。この右辺のベクトルは、数値積分を用いて求めることができる。なお、行列  $Q, Q_1^{(00)}, Q_2^{(00)}$  は数式 3 5、ベクトル  $b_1(s), b_2(t)$  は数式 3 7 で定義されている。

【 0 2 2 0 】

(数値実験例)

極値検出の数値例を示す。極値検出の対象となる最適平滑化スプライン曲面  $x(s, t)$  は、次式をサンプルして得られたデータに対して設計した。

【 0 2 2 1 】

【数 6 2 】

$$d(s, t) = \sum_{i=1}^6 a_i \cdot \exp\left(-\frac{(s - p_i)^2 + (t - q_i)^2}{r_i}\right)$$
20

【 0 2 2 2 】

ここでのパラメータの値を次の表に示す。

【 0 2 2 3 】

【表 1 】

$i$	$a_i$	$p_i$	$q_i$	$r_i$
1	-22	5	3.5	0.7
2	23	4	4	0.5
3	-10	2.5	6	1
4	25	5.5	8.5	0.5
5	30	6	5.5	0.4
6	-7	5	5	30

30

40

50

10

## 【0224】

曲面の設計の際に用いた各パラメータは、 $k=3$ 、 $\alpha=1$ 、 $s_0=t_0=0$ 、 $m_1=m_2=10$  である。従って、3次スプライン曲面を領域 $S=[0, 10] \times [0, 10]$ で設計した。設計に用いたデータは、上記 $d(s, t)$ を $s, t$ 方向とも0.5間隔の格子点でサンプリングして生成した。設計された曲面 $x(s, t)$ に上述の極値検出法を適用して得られた結果を図12に示す。図では、検出された極大値(○印)と極小値(●印)を、曲面 $x(s, t)$ の3次元表示に重ねて示した。また、 $st$ 平面上には曲面の等高線表示と極値点とを示した。いずれの図からもすべての極値が検出されていることが確認できる。

20

## 【0225】

以上説明したように、本実施形態の最適平滑化スプライン曲面による極値検出方法及び極値検出プログラムによれば、与えられた $k$ 次スプライン曲面 $x(s, t)$ のすべての極値(極小値、極大値)を求めることが可能となる。従って、最小値や最大値も求めたことになる。曲面 $x(s, t)$ の設計は、与えられた離散空間データ $D$ (数式32)に基づく既述の方法でも良いし、あるいは与えられた関数 $f(s, t)$ を近似する曲面の設計によっても良い。後者の場合、設計されたスプライン曲面 $x(s, t)$ の最大値や最小値を求め、これらを与える点を初期値として、関数 $f(s, t)$ にさらに山登り法等の数値的探索法を適用することによって、 $f(s, t)$ の最大値や最小値がより正確、かつ系統だった方法で求められることになる。

30

## 【0226】

なお、上記実施形態では、最適平滑化スプライン関数 $x(s, t)$ の極値を求めたが、同様な方法で、 $x(s, t)$ の任意の偏導関数の極値を求めることが可能である。

## 【0227】

(偏導関数の極値)

いま記号の簡単化のために $x(s, t)$ と $x(u, v)$ の偏導関数を以下のようにおく。

40

## 【0228】

【数63】

$$x_{i,j}(s, t) = \frac{\partial^{i+j}}{\partial s^i \partial t^j} x(s, t), \quad \hat{x}_{i,j}(u, v) = \frac{\partial^{i+j}}{\partial u^i \partial v^j} \hat{x}(u, v)$$

## 【0229】

また、問題が意味をもつために $i, j \leq k-2$ とする。このとき、数式42の関係は、 $x_i, \dots$

50

$x_{i,j}(s, t) = x_{i,j}(u, v)$  となり、偏導関数  $x_{i,j}(s, t)$  の極値は、 $x_{i,j}(u, v)$  の極値から求めることができる。すなわち、 $x_{i,j}(u, v)$  が点  $(u^*, v^*)$  で極値をとれば、 $x_{i,j}(s, t)$  は点  $(s^*, t^*) = ((1/\mu)u^* + s_0, (1/\mu)v^* + t_0)$  で、極値  $x_{i,j}(s, t) = x_{i,j}(u^*, v^*)$  をもつことが分かる。

【 0 2 3 0 】

(関数  $x_{i,j}(u, v)$  の表現)

$x_{i,j}(u, v)$  は、数式 4 3 から

【 0 2 3 1 】

【数 6 4】

10

$$\hat{x}_{i,j}(u, v) = h_{k+1-i}^T(u) A_{i,j} h_{k+1-j}(v)$$

【 0 2 3 2 】

と表される。ここで、

【 0 2 3 3 】

【数 6 5】

20

$$A_{i,j} = (C_k C_{k-1} \cdots C_{k+1-i})^T A (C_k C_{k-1} \cdots C_{k+1-j})$$

【 0 2 3 4 】

である。この  $x_{i,j}(u, v)$  は、数式 4 3 の  $x(u, v)$  と同形式であることから、その極値も  $x(u, v)$  の極値の検出、計算の場合と同様な手順に従って求めることができる。もちろん、偏導関数  $x_{i,j}(s, t)$  のすべての極値（極小値、極大値）を求めることができ、従って、その最小値や最大値も求まったことになる。

30

【 0 2 3 5 】

(第 3 実施形態)

以下、本発明の第 3 実施形態である最適平滑化スプラインによる極値検出方法及び極値検出プログラムについて、図を参照して詳細に説明をする。

【 0 2 3 6 】

本発明の第 2 実施形態では、空間における離散データを近似する最適平滑化スプライン曲面を生成し極値の検出を行ったが、本実施形態では、離散データとしてデジタル画像の各ピクセルの輝度値データについて極値の検出を行う。さらに、ここでは、最適平滑化スプライン曲面  $x(s, t)$  の  $s$  方向および  $t$  方向の 1 次偏導関数の極値を検出することにより、デジタル画像のエッジ検出を行う。

40

【 0 2 3 7 】

(曲面の生成)

デジタル画像のエッジ検出に上記の極値検出法を適用した。対象としたのは  $256 \times 256$  (pixel) のモノクロ画像であり、各ピクセルの輝度値をデータとして、まず、最適平滑化スプライン曲面  $x(s, t)$  を設計し、画像のエッジは、輝度値の変化率が急激に変化する部分（つまり、最適平滑化スプライン曲面  $x(s, t)$  の  $s$  方向および  $t$  方向の 1 次偏導関数の極値が存在する部分）に存在するとの考えに基づき検出する。

【 0 2 3 8 】

本実施形態では、最適平滑化スプライン曲面  $x(s, t)$  をライン毎に（たとえば  $s$  を  $s=p$  に固定し）処理している。その場合、エッジは変数  $t$  に関するスプライン曲線  $x(p, t)$  の 1 次

50



導関数の極値として求めればよい。

【 0 2 3 9 】

図 6 は、本実施形態の最適平滑化スプライン曲面を概略的に示す図である。図 6 A は、各ピクセルの位置（座標）及びその輝度の分布から求められた最適な制御点（重み係数）の分布を示す図であり、図 6 B は、各ピクセルの空間における離散データに基づいて生成された最適平滑化スプライン曲面 $x(s, t)$ 及びライン毎に生成された関数 $x(s, q)$ 、 $x(p, t)$ を示す図である。

【 0 2 4 0 】

図 1 3 は、本実施形態の最適平滑化スプラインによる極値検出方法及び極値検出プログラムを用いたデジタル画像のエッジ検出処理装置を示すブロック図である。

10

【 0 2 4 1 】

本実施形態の画像のエッジ検出処理装置 2 1 は、CPU 2 5（演算手段）とメモリ装置 2 4（RAM や ROM 等：記憶手段）を有し、このメモリ装置 2 4 に格納されたプログラムを実行し、図 1 4 ~ 1 6 のフローチャートに記載した動作を実行することにより、図 1 3 のブロック図に示すように、CPU 2 5（演算手段）と、メモリ装置 2 4（記憶手段）と、マウス、キーボード、スキャナ、データロガー等の入力装置 2 6（入力手段）と、バス 2 3 と、ディスプレイ装置 2 7（出力手段）と、プリンタ装置 2 8（出力手段）と、外部ネットワーク 3 0 と接続するネットワーク装置 2 9（入力手段・出力手段）を備えた処理装置として機能する。

【 0 2 4 2 】

20

図 1 4 は、本実施形態の最適平滑化スプラインによるデジタル画像のエッジ検出方法の処理の概要を示すフローチャートである。

【 0 2 4 3 】

ステップ 3 0 1 では、ピクセルデータ $(i, j, f_{i, j})$ 、 $i = 1, 2, \dots, I, j = 1, 2, \dots, J$ 、を、画像のエッジ検出処理装置 2 1 に入力する工程である。ここで、 $f_{i, j}$  は、ピクセルの座標 $(i, j)$ におけるピクセルの輝度値を示す。ピクセルデータは、例えば、不図示の光学的読取手段にてデータとして取得されうるものである。また、ピクセルデータは、ネットワーク装置 2 9 を介して不図示の外部装置から取得してもよい。そして、しきい値 をメモリ装置 2 4 に格納する。しきい値 は、極値がエッジ部分かどうか判別するためのパラメータである。 $x^{(1)}(s, t)$ の極値（極大、極小）の絶対値がある程度以上の大きさの箇所をエッジとみなす。その程度は、画像の状態やエッジ検出条件により異なるため、エッジ検出においてしきい値 を設定する。しきい値 よりも極値の絶対値が大きければ、その箇所をエッジ部分としてみなし、それ以外の場合は、極値であってもエッジ部分とはみなさない。

30

【 0 2 4 4 】

ステップ 3 0 2 では、CPU 2 5 は、上記入力ピクセルデータ $(i, j, f_{i, j})$ に基づいて、最適平滑化スプライン曲面を生成する。具体的には、CPU 2 5 は、最適な重み係数 $w_{ij}$  ( $i = -k, -k+1, \dots, l-1, j = -k, -k+1, \dots, J-1$ )を求め、数式 2 8 に示す最適平滑化スプライン曲面を生成する。最適平滑化スプライン曲面を生成する方法については、本発明の第 2 実施形態で説明した通りである。その際のパラメータは  $s_0 = t_0 = 1, m_1 = l - 1, m_2 = J - 1$  とする。

40

【 0 2 4 5 】

最適平滑化スプライン曲面を生成したのち、その曲面をライン毎に（たとえば $s = q$ に固定し）処理する。図 6 の  $s$  軸方向に対する処理をするために、ステップ 3 0 3 へ進み、図 6 の  $t$  軸方向の処理をするためにステップ 3 0 4 へ進む。

【 0 2 4 6 】

ステップ 3 0 3 では、CPU 2 5 は、ステップ 3 0 2 で生成された最適平滑化スプライン曲面 $x(s, t)$ について、 $t$  座標を $q$  ( $q = 1, 2, \dots, J$ )で固定した複数の曲線 $x(s, q)$ （図 6 B 参照）を生成し、曲線 $x(s, q)$ の 1 次導関数 $x^{(1)}(s, q)$ について極値を検出する（ステップ 3 0 3 a）。そして、CPU 2 5 は、この極値に基づいて、各々エッジの判定の

50

処理を行う(ステップ303b)。

【0247】

ステップ304では、CPU25は、ステップ302で生成された最適平滑化スプライン曲面 $x(s, t)$ について、 $s$ 座標を $p(p=1, 2, \dots, l)$ で固定した複数の曲線 $x(p, t)$ (図6B参照)を生成し、曲線 $x(p, t)$ の1次導関数 $x^{(1)}(p, t)$ について極値を検出する(ステップ304a)。そして、CPU25は、この極値に基づいて、各々エッジの判定の処理を行う(ステップ304b)。

【0248】

ステップ305では、CPU25は、ステップ303とステップ304で得られた判定により検出されたエッジ箇所を総和としてまとめる。すなわち $s$ 方向、 $t$ 方向のうち少なくとも一方でエッジと判定された箇所をエッジと判定する。

10

【0249】

ステップ306では、CPU25は、ステップ305の結果に基づいて、入力画像に対するエッジ検出結果を画像に変換し、ディスプレイ装置27やプリンタ装置28に出力する。

【0250】

図15は、ステップ303における処理の詳細を示すフローチャートである。ステップ401~403が、図14のステップ303aに相当し、ステップ404~407が、図14のステップ303bに相当する。

【0251】

ステップ401は、CPU25に、図14のステップ301で求めたピクセルデータ( $i, j, f_{i,j}$ )について重み係数 $w_{i,j}$ ( $i=-k, -k+1, \dots, l-1, j=-k, -k+1, \dots, J-1$ )を入力する工程である。

20

【0252】

なお、以下に使用される数式に関して、 $\hat{\quad}$ を変数とする関数 $x$ には、すべて‘ハット( $\hat{\quad}$ )’がつくものであるが、明細書本文中では省略するものとする。

【0253】

ステップ402では、CPU25は、ステップ302で生成された最適平滑化スプライン曲面 $x(s, t)$ を、 $t$ 座標を $q(q=1, 2, \dots, J)$ で固定して生成された複数の曲線 $x(s, q)$ について、節点区間 $[s_j, s_{j+1})$ において正規化を行う。そして、単位区間 $[0, 1)$ における正規化された曲線 $x(\hat{s}, q)$ を生成し、1次導関数 $x^{(1)}(\hat{s}, q)$ を求め、多項式である $x^{(1)}(\hat{s}, q)$ について、各係数 $a_0, \dots, a_{k-1}$ の計算を行う。

30

【0254】

ステップ403では、CPU25は、多項式 $x^{(1)}(\hat{s}, q)$ の極値点 $s_{1}, s_{2}, \dots, s_m$ ( $m \leq k-1$ )の検出を行う。多項式 $x^{(1)}(\hat{s}, q)$ の極値点 $s_{1}, s_{2}, \dots, s_m$ は、 $x^{(2)}(\hat{s}, q)$ の根のうち、単位区間 $[0, 1)$ にある実根を検出して求めることができる。

【0255】

ステップ404、405では、CPU25は、すべての各極値点 $s_{1}, s_{2}, \dots, s_m$ について、 $x^{(1)}(\hat{s}_l, q) \cdot x^{(3)}(\hat{s}_l, q) < 0$ 、かつ、 $|x^{(1)}(\hat{s}_l, q)| > (0 \mid m)$ が否かを判断する。 $x^{(1)}(\hat{s}_l, q) \cdot x^{(3)}(\hat{s}_l, q) < 0$ 、かつ、 $|x^{(1)}(\hat{s}_l, q)| > (0 \mid m)$ の場合には、ステップ406へ進む。そうでない場合には、ステップ404に戻る。

40

【0256】

ステップ406では、CPU25は、 $x^{(1)}(\hat{s}_l, q) \cdot x^{(3)}(\hat{s}_l, q) < 0$ 、かつ、 $|x^{(1)}(\hat{s}_l, q)| > (0 \mid m)$ の場合には、点 $(p+(1/\hat{\quad}) \hat{s}_l, q)$ においてエッジがあるものと判断する。これは、 $x^{(1)}(t, q)$ の極値の箇所が必ずしもエッジの位置に対応しないことがあり、 $x^{(1)}(\hat{s}_l, q) \cdot x^{(3)}(\hat{s}_l, q) < 0$ を満たす箇所が、実際のエッジに対応するからである。

【0257】

50

ステップ407では、CPU25は、 $p < I - 1$ の場合には、ステップ402～ステップ406の各工程を繰り返し、ステップ408では、 $q < J$ の場合には、ステップ402～ステップ407の各工程を繰り返し、ステップ302で生成された最適平滑化スプライン曲面 $x(s, t)$ の $t$ 座標を $q(q=1, 2, \dots, J)$ で固定して生成された複数の曲線 $x(s, q)$ のすべてについてエッジを判断する。

【0258】

図16は、ステップ304における処理の詳細を示すフローチャートである。

【0259】

ステップ501は、CPU25に、図14のステップ301で求めたピクセルデータ $(i, j, f_{i, j})$ について重み係数 $w_{i, j}$  ( $i = -k, -k+1, \dots, l-1, j = -k, -k+1, \dots, J-1$ )を入力する工程である。

10

【0260】

ステップ502では、CPU25は、ステップ302で生成された最適平滑化スプライン曲面 $x(s, t)$ を、 $s$ 座標を $p(p=1, 2, \dots, l)$ で固定して生成された複数の曲線 $x(p, t)$ について、区間 $[t_j, t_{j+1})$ において正規化を行う。そして、単位区間 $[0, 1)$ における正規化された曲線 $x(p, t)$ を生成し、1次導関数 $x^{(1)}(p, t)$ を求め、多項式である $x^{(1)}(p, t)$ について、各係数 $b_0, \dots, b_{k-1}$ の計算を行う。

【0261】

ステップ503では、CPU25は、多項式 $x^{(1)}(p, t)$ の極値点 $t_1, t_2, \dots, t_n$  ( $n = k-1$ )の検出を行う。多項式 $x^{(1)}(p, t)$ の極値点 $t_1, t_2, \dots, t_n$ は、 $x^{(2)}(p, t)$ の根のうち、単位区間 $[0, 1)$ にある実根を検出して求めることができる。

20

【0262】

ステップ504、505では、CPU25は、すべての各極値点 $t_1, t_2, \dots, t_n$ について、 $x^{(1)}(p, t_{1l}) \cdot x^{(3)}(p, t_{1l}) < 0$ 、かつ、 $|x^{(1)}(p, t_{1l})| > (0 \leq l < n)$ か否かを判断する。 $x^{(1)}(p, t_{1l}) \cdot x^{(3)}(p, t_{1l}) < 0$ 、かつ、 $|x^{(1)}(p, t_{1l})| > (0 \leq l < n)$ の場合には、ステップ506へ進む。そうでない場合には、ステップ504に戻る。

【0263】

ステップ506では、CPU25は、 $x^{(1)}(p, t_{1l}) \cdot x^{(3)}(p, t_{1l}) < 0$ 、かつ、 $|x^{(1)}(p, t_{1l})| > (0 \leq l < n)$ の場合には、点 $(p, q+(1/l) \cdot t_{1l})$ においてエッジがあるものと判断する。

30

【0264】

ステップ507では、 $q < J - 1$ の場合には、CPU25は、ステップ502～ステップ506の各工程を繰り返し、ステップ508では、 $p < I$ の場合には、ステップ502～ステップ507の各工程を繰り返し、ステップ302で生成された最適平滑化スプライン曲面 $x(s, t)$ を、 $s$ 座標を $p(p=1, 2, \dots, l)$ で固定して生成された複数の曲線 $x(p, t)$ のすべてについてエッジを検出する。

【0265】

(デジタル画像のエッジ検出例)

40

図17及び図18は、本発明の第3実施形態の最適平滑化スプラインによる極値検出法に用いたデジタル画像のエッジ検出方法を用いて、実際にエッジ検出を行った例を示す図である。図17は、原画像(a)にノイズが含まれない例を示し、図18は、原画像(a)にノイズが含まれる例を示す。

【0266】

図17(b)と図18(b)は、本実施形態のエッジ検出法を適用して生成した画像である。また、比較例として、cannyフィルタ及びsobelフィルタを用いてエッジ検出した画像を、図17(c)と図18(c)(cannyフィルタ)及び図17(d)と図18(d)(sobelフィルタ)に示した。

【0267】

50

図からもわかるように、特に、原画像(a)にノイズが含まれる場合において、他のフィルタと比較して、精度良くエッジの検出ができることがわかる。よって、本発明の最適平滑化スプラインによる極値検出方法に用いたデジタル画像のエッジ検出方法の有効性が証明された。

【0268】

以上説明したように、本実施形態の最適平滑化スプラインによる極値検出方法によれば、スプライン曲面の設計が画像情報全体の情報に基づいた曲面近似となるため、デジタル画像のエッジ検出は大域的な方法となる。従来の局所的な(画像の一部の情報に基づいた)方法とはまったく異なる方法であるといえ、本実施形態の最適平滑化スプラインによる極値検出方法は、結果的に画像のノイズの影響を受けにくい。

10

【0269】

なお、上記各実施形態においては、スプライン曲線(1次元)の場合、スプライン曲面(2次元)の場合について説明をしたが、本発明はこれに限られず、さらなる高次元の場合に拡張して適用することが可能である。

【図面の簡単な説明】

【0270】

【図1】最適平滑化スプライン曲線を概略的に示す図である。

【図2】本発明の第1実施形態の最適平滑化スプラインによる極値検出方法に用いられる処理装置を示すブロック図である。

【図3】本発明の第1実施形態の最適平滑化スプラインによる極値検出方法の処理の流れを示すフローチャートである。

20

【図4】本発明の第1実施形態の最適平滑化スプラインによる極値検出方法の処理の流れにおいて、最適平滑化スプラインが3次である場合を示すフローチャートである。

【図5】本発明の第1実施形態の最適平滑化スプライン曲線とその極値を示す図である。

【図6A】空間における離散データから求められた最適な制御点(重み係数)の分布を示す図である。

【図6B】最適平滑化スプライン曲面を概略的に示す図である。

【図7】本発明の第2実施形態の最適平滑化スプラインによる極値検出方法に用いたデジタル画像のエッジ検出処理装置を示すブロック図である。

【図8】本発明の第2実施形態の最適平滑化スプラインによる極値検出方法及び極値検出プログラムの処理の概要を示すフローチャートである。

30

【図9】図8のフローチャートのステップ203における処理の詳細を示すフローチャートである。

【図10】図8のフローチャートのステップ206における処理の詳細を示すフローチャートである。

【図11】図8のフローチャートのステップ211における処理の詳細を示すフローチャートである。

【図12】本発明の第2実施形態の最適平滑化スプラインによる極値検出方法及び極値検出プログラムを用いて、実際に極値検出を行った例を示す図である。

【図13】本発明の第3実施形態の最適平滑化スプラインによる極値検出方法及び極値検出プログラムを用いたデジタル画像のエッジ検出処理装置を示すブロック図である。

40

【図14】本発明の第3実施形態の最適平滑化スプラインによるデジタル画像のエッジ検出方法の処理の概要を示すフローチャートである。

【図15】図14のフローチャートのステップ303における処理の詳細を示すフローチャートである。

【図16】図14のフローチャートのステップ304における処理の詳細を示すフローチャートである。

【図17】本発明の第3実施形態の最適平滑化スプラインによる極値検出方法を用いたデジタル画像のエッジ検出方法を用いて、実際にエッジ検出を行った例を示す図である。

【図18】本発明の第3実施形態の最適平滑化スプラインによる極値検出方法を用いたデ

50

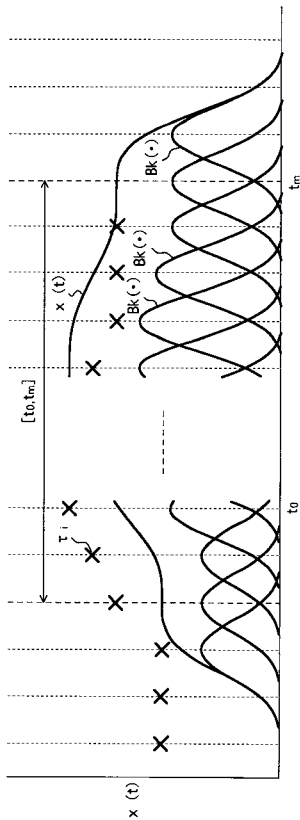
デジタル画像のエッジ検出方法を用いて、実際にエッジ検出を行った例を示す図である。

【符号の説明】

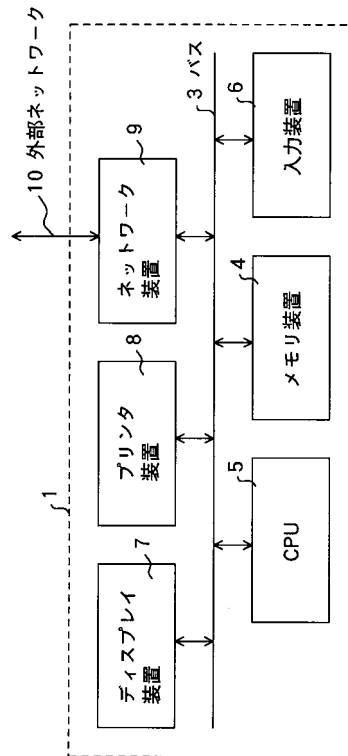
【0271】

- 1：最適平滑化スプラインによる極値検出処理装置
- 3：バス
- 4：メモリ装置
- 5：CPU
- 6：入力装置
- 7：ディスプレイ装置
- 8：プリンタ装置
- 9：ネットワーク装置

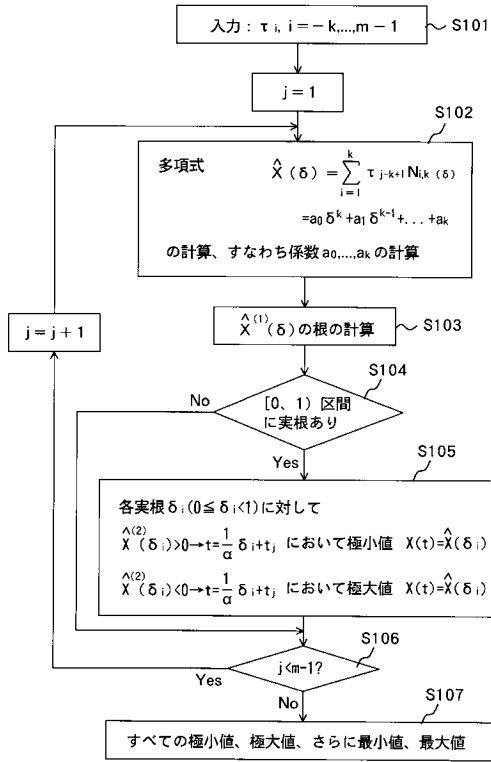
【図1】



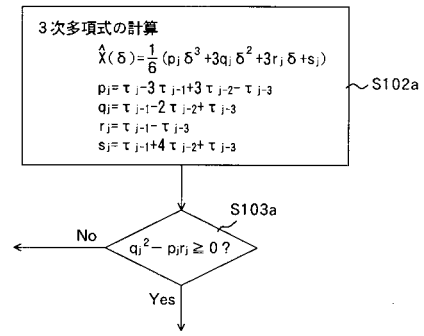
【図2】



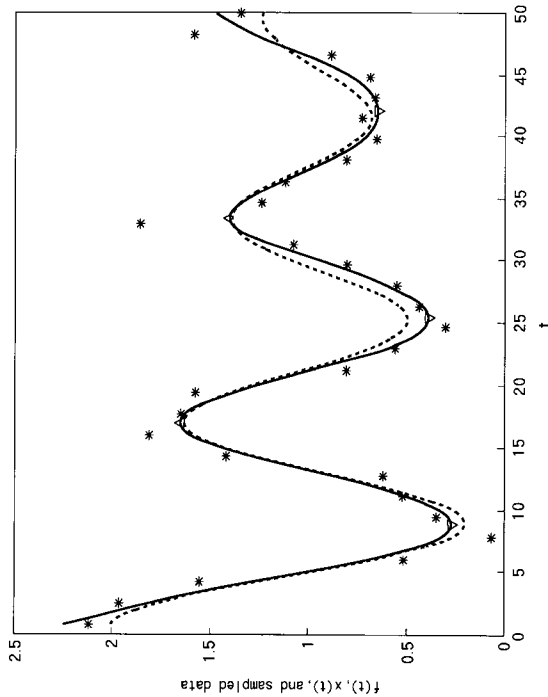
【図3】



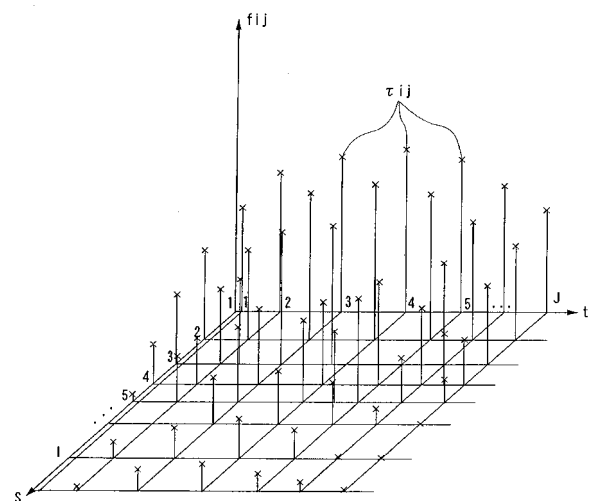
【図4】



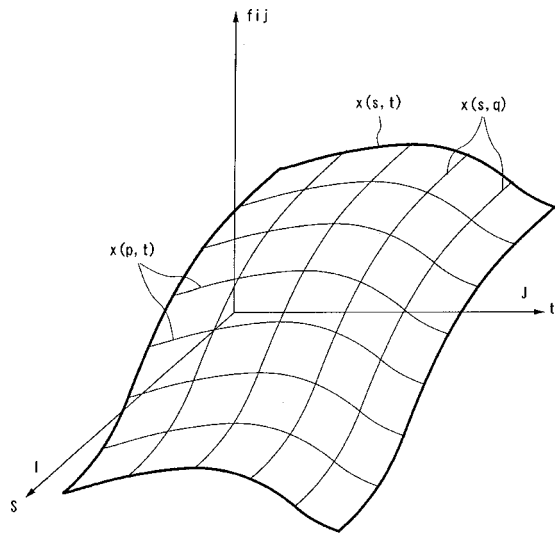
【図5】



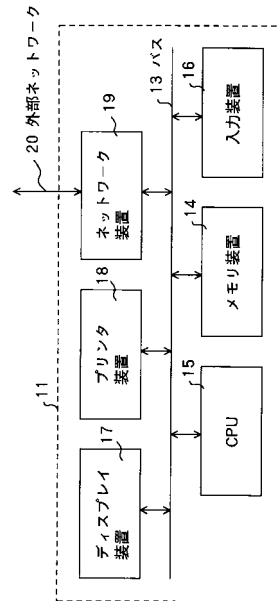
【図6A】



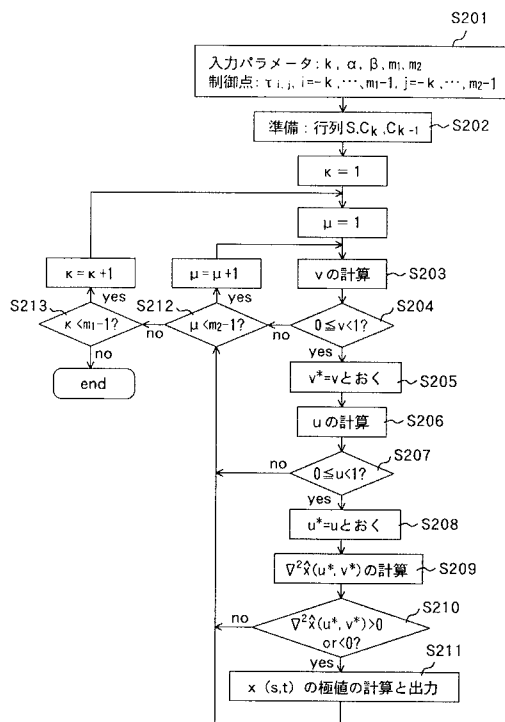
【図6B】



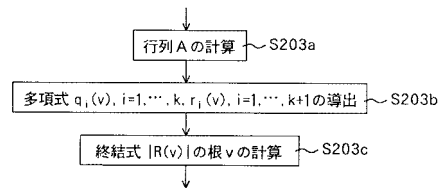
【図7】



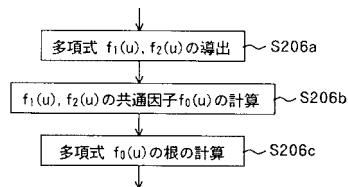
【図8】



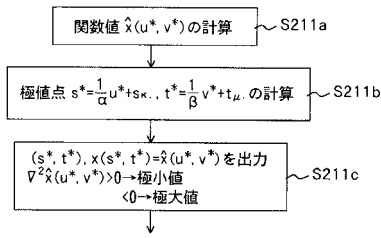
【図9】



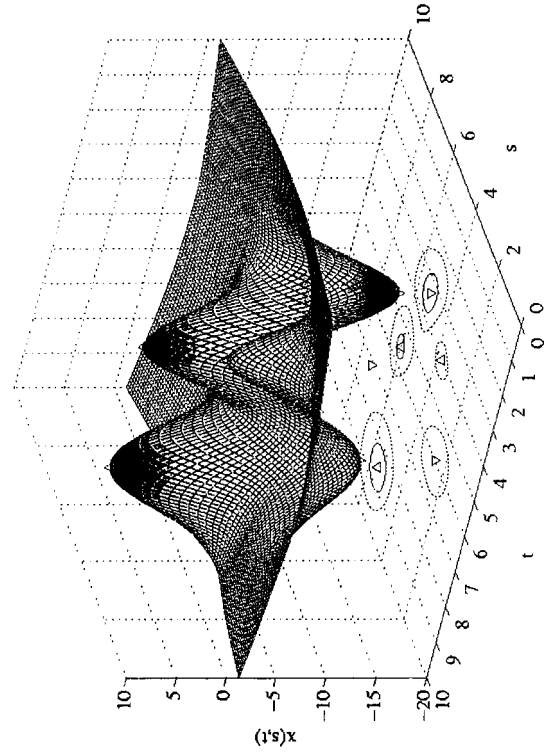
【図10】



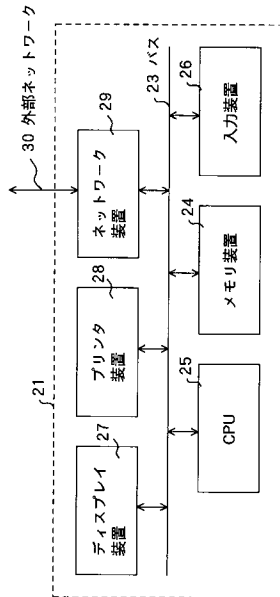
【図 1 1】



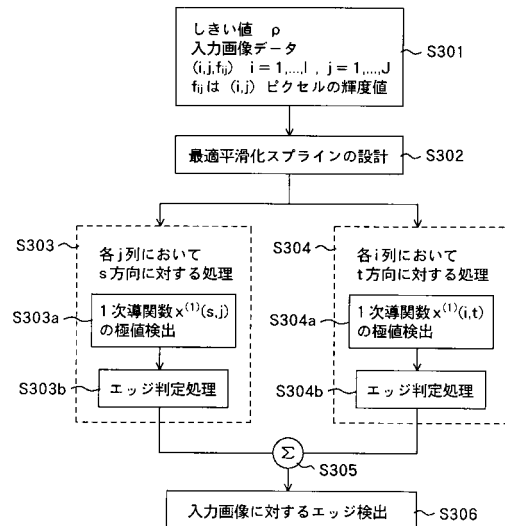
【図 1 2】



【図 1 3】

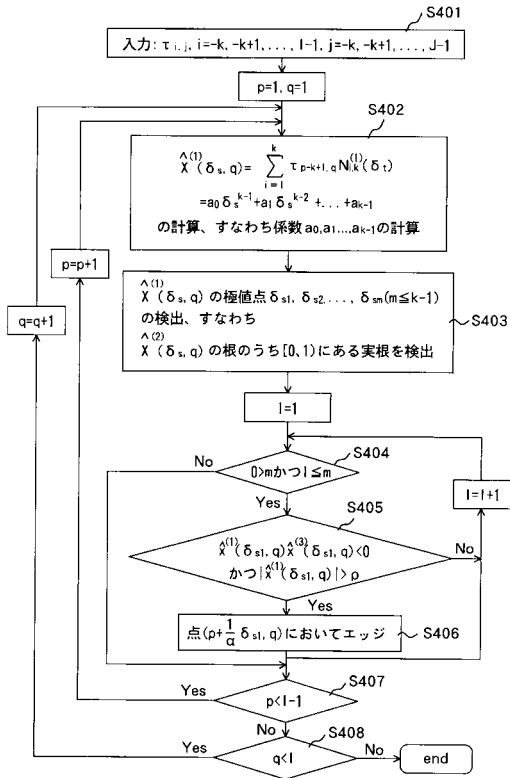


【図 1 4】

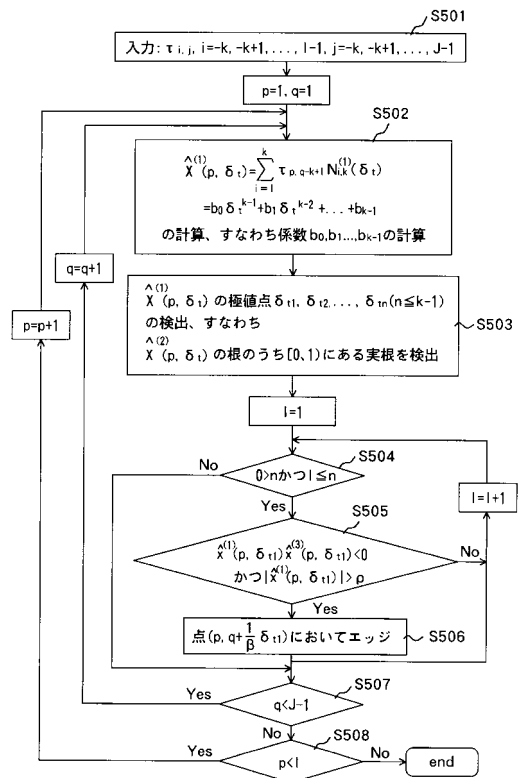




【図15】



【図16】



【図 17】



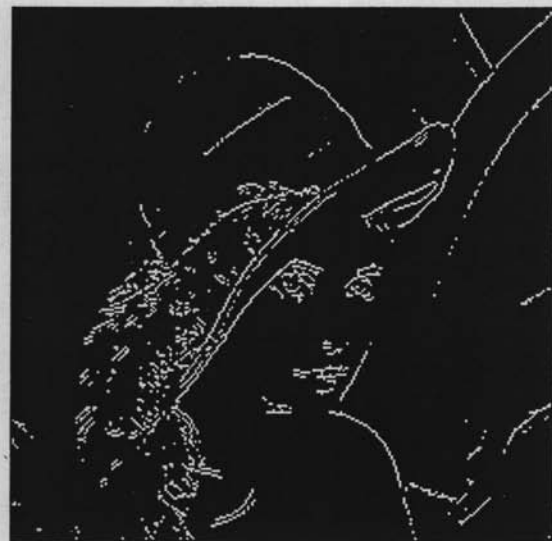
(a) 原画像



(b) 提案法



(c) canny フィルタ



(d) sobel フィルタ

【図 18】



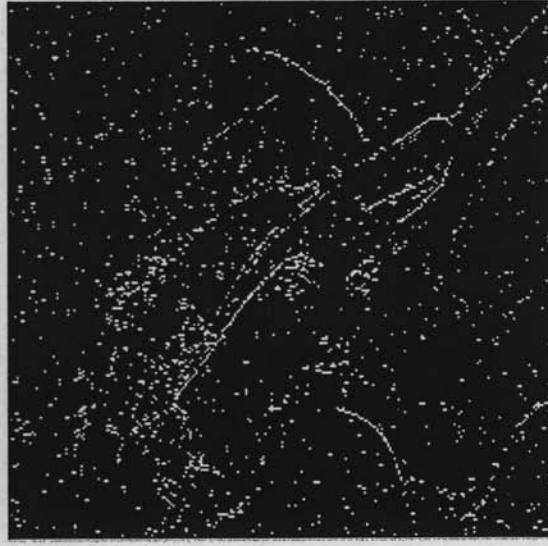
(a) ノイズを含む画像



(b) 提案法



(c) canny フィルタ



(d) sobel フィルタ

---

フロントページの続き

審査官 松永 稔

- (56)参考文献 特開平10-262941(JP,A)  
特開2005-326977(JP,A)  
特表2003-509748(JP,A)  
森川博由、坪川直宏、柳雄一郎、平滑化スプライン関数による音声ピッチパターンのモデル化と分析、電子情報通信学会技術研究報告. SP, 音声, 日本, 社団法人電子情報通信学会, 2000年 7月20日, 100(239), pp.17-24

- (58)調査した分野(Int.Cl., DB名)  
G06T 7/60