

視覚的な自然さに基づくグラフクラスタリングを用いた 形状セグメンテーションのための 類似度重みのパラメータ調整

佐藤 信¹

概要：視覚的な自然さを表現する形状特徴量に基づいた、既提案の形状セグメンテーション手法について、形状の類似度を表現するグラフのエッジの重みを調整するための手法を提案する。グラフラプラス行列の固有ベクトルに基づいて、重みのパラメータの調整をおこない、生成される形状セグメントについて検討する。構造化されていない曲線セグメントの集合を基準形状として類似形状を生成して、機械学習のデータとするのが目的である。

Parameter Adjustment of Similarity Weights for Shape Segmentation Using Graph Clustering Based on Visual Naturality

MAKOTO SATOH¹

Abstract: This paper presents a method to adjust the edge weights of graphs, represent the similarity of shapes, for a previously proposed shape segmentation method based on the shape features to represent visual naturality. In the method, the weight parameters are adjusted based on the graph Laplacian eigenvectors. The shape segments generated using the adjusted parameters are analyzed. The method will be used in the learning set generation for machine learning, by the generation of the similar shapes from base shapes constructed from set of unstructured curve segments.

1. はじめに

本稿では、視覚的な自然さに基づいた形状セグメンテーションのための既提案の手法 [7], [8] について、形状の類似度を表現する重みの計算に用いるパラメータを調整するための手法を提案する。

ある形状が与えられた場合に、その形状を洗練化 [2] する領域を選択するための手法として、どのような手法が適当であるのかは、適用分野に依存することである。そのため、多くの手法が提案されている。本稿でパラメータを調整する手法は、平面上の Bezier 曲線の集合により表現された形状を対象とする手法である。その手法は、形状の視覚的な自然さをセグメンテーションの基準として用いることが特徴であり、形状の一部分を直感的に選択して形状を洗

練化するような作業に適する。既に作成してある形状の一部分を形状洗練化する場合ばかりではなく、形状を作成する過程での形状洗練化にも有効である。その理由は、形状が完成するまでの作成過程では、その形状が何を表現するのが不明確な場合があるので、視覚的な自然さを表現する形状特徴量が有効であるからである。

なお、その既提案の手法の適用分野としては、機械学習の学習データの生成がある。基準とする形状から、それに類似な形状を生成することにより学習データを生成する場合に、類似形状の生成に用いる非均一相似変換 [9], [10] をおこなう領域を選択するために用いることが可能である。なお、非均一相似変換 [9], [10] は、Fowler 等 [3] の提案した曲線の形状洗練化のための手法を拡張したものである。非均一相似変換を用いると、平面上の Bezier 曲線で表現した基準形状を、曲線の通過点などの変形のための制約条件

¹ 岩手大学
Iwate University, Ueda, Iwate 020-8551, Japan

を満たしながら、可能な限り基準形状の特徴を維持して形状を洗練化することが可能である。

これ以降の構成について、簡単に説明する。2節では、関連研究について説明をおこなう、そして、既提案のグラフクラスタリングを用いた形状セグメンテーション手法の概要と特徴について述べる。そして、3節では、既提案のセグメンテーション手法について、類似度重みのパラメータの調整手法を提案する。提案手法の実装と結果の検討について4節で説明する。そして最後に、5節で本稿のまとめと今後について述べる。

2. 関連研究と既提案手法の概要

2.1 関連研究

セグメンテーションは、多くの分野で必要とされる技術であり、多くの手法が提案されている。それらがセグメンテーションの対象としているのは、ソーシャル・ネットワーク、コンピュータ・ネットワーク、画像を含む2次元形状、そして、3次元形状などであり、非常に広い分野である。

例えば、Katz等[5]は、3次元メッシュのセグメンテーション手法について研究している。ここでは、セグメンテーションの基準として、メッシュを構成するポリゴンフェース間の形態的な距離、および、隣接するポリゴンフェースのなす角度を用いている。このように、Katz等[5]の手法では、セグメンテーションをおこなう要素どうしの隣接関係についてのデータを用いている。本稿でパラメータの調整をおこなう既提案の手法は、構造化されていないBezier曲線セグメントの集合からのサンプリング点についてセグメンテーションをおこなうのが特徴であり、それらのサンプリング点どうしの接続に関するデータをアルゴリズムの入力として必要としない。

つぎに、形状を構造化するための関連研究について述べる。Hammond等[4]は、複数ストロークにより入力したストローク点を、図形として認識するための研究をおこなっている。ここでは、ストロークの本数と順序が自由であるが、ストロークを接続するためのストローク間の距離などを予め与えている。本稿では既提案のセグメンテーション手法について、セグメンテーションする形状に合わせて、形状の類似度を表現する特徴量を計算するためのパラメータを調整する手法を提案する。

形状のシルエットを表現するサンプリング点から NPR スケッチを作成するための Lewis 等 [6] の研究では、グラフ Laplace 行列を用いて形状をセグメンテーションしている。その手法では、形状の接線の傾きの類似度に基づいて、サンプリング点の集合の2点どうしの類似度を計算している。その場合に、2つのサンプリング点の類似度への接線の傾きの感度を、2点間の距離に関して、パラメータにより調節することが可能である。しかし、その調節方法につ

いてはふれていない。本稿では、2点間の距離、形状の接線の傾きそして曲率を用いてサンプリング点の2点どうしの類似度を計算する既提案の手法について、類似度の計算において用いる、距離測度のスケーリングパラメータを調整する手法を提案する。

また、グラフのクラスタリングは、広く用いられている手法であるが、本稿では、グラフ Laplace 行列を用いてクラスタリングをする。その理由は、形状の特徴を大域的にとらえるためには、グラフ Laplace 行列によりセグメンテーションをすることが有効であると考えたからである。グラフ Laplace 行列を用いるセグメンテーション手法は、Fiedler[1]の研究などを基に研究されているものであり、広く用いられている。これまでに、複数の Laplace 行列が提案されているが、それらには非正規化 Laplace 行列と正規化 Laplace 行列がある。本稿でパラメータの調整をする既提案手法では、Laplace 行列を生成するためのグラフの重みの計算手法を提案している。その重みのクラスタリングへの影響について検討をおこなう始めの段階として、非正規化 Laplace 行列を用いている。非正規化 Laplace 行列の詳細については、Mohar [11], [12]などに記述がある。

2.2 既提案手法の概要

パラメータの調整手法を提案する既提案の形状セグメンテーション手法 [7], [8] の概要を説明する。その手法が対象とするのは、曲線セグメントどうしの接続関係などの構造に関する情報が与えられていない、Bezier 曲線セグメントの集合により構成されている形状である。そのように構造化されていない形状について、その形状の視覚的な自然さを表現するように設計した形状特徴量を計算し、その形状特徴量をエッジの重みとするグラフをクラスタリングすることにより形状をセグメンテーションする。その手順の概要を示す。

Step 1: サンプリング 構造化されていない Bezier 曲線セグメントの集合から、曲線上の点をサンプリングする。ここで、各サンプリング点について、その位置座標、接線の傾き、そして曲率をサンプリングする。

Step 2: クラスタリング サンプリング点集合の2点どうしについて、その点での形状の類似度を表現する重みを計算する。この重みを用いて、グラフ Laplace 行列を作成する。そして、その行列の固有値と固有ベクトルに基づいてクラスタリングをする。

3. 形状セグメンテーションでのパラメータの調整

2.2 節で概要を説明した既提案の形状セグメンテーション手法 [7], [8] について、パラメータを調整するための手法を説明する。

3.1 クラスタリングで用いる類似度

グラフを用いて形状を表現するために、形状を構成する曲線からのサンプリング点を節点 V 、そして、各節点を結ぶエッジを E として、グラフ $G = \langle V, E \rangle$ を作成する。ここで、各エッジに、そのエッジが接続する2つの節点の形状の類似度を表現する重み w を与える。

$$w = \exp\left(-\left(\frac{d_p^2}{\alpha_p^2} + \frac{d_T^2}{\alpha_T^2} + \frac{d_C^2}{\alpha_C^2} + \frac{d_U^2}{\alpha_U^2}\right)\right) \quad (1)$$

ここで、距離 d_p, d_T, d_C そして d_U は、そのエッジが接続する2つの節点についてのものである。それぞれ、位置座標、接線の傾き、符号付き曲率、そして、符号なし曲率を基にして計算する。また、 $\alpha_p, \alpha_T, \alpha_C$ そして α_U は、スケールリングのためのパラメータである。この重みをすべてのサンプリング点について計算する。そして、グラフ G のグラフ Laplace 行列の Fiedler ベクトルを基にして、形状をセグメンテーションする。詳細については、[7], [8] に説明がある。

なお、これまでに提案した類似度重み [7], [8] からの変更点は、(1) 式では、スケールリングパラメータ $\alpha_p, \alpha_T, \alpha_C$ そして α_U を、それぞれ二乗しているという点である。その理由は、3.3 節で説明をするパラメータの調整手法において、最適化の過程でパラメータを変化させる場合に、パラメータの符号を制限する必要がないようにするためである。ただし、最適化の結果として得られるパラメータの値の範囲は、3.3 節で説明する (9) 式で調整をするようにしている。

3.2 形状セグメンテーションでのパラメータ

既提案の手法での重み計算のためのパラメータを、表 1 に示す。なお、詳細については、[7], [8] に説明がある。

表 1 既提案の形状セグメンテーション手法でのパラメータ
Table 1 The parameters in previously proposed shape segmentation method

parameters	description	
α_p	scaling	positional difference
α_T		tangential difference
α_C		signed curvature difference
α_U		unsigned curvature difference
β_p	thresholding	positional difference
β_T		tangential difference
β_C		signed curvature difference
β_U		unsigned curvature difference

3.3 パラメータの調整

形状の視覚的な自然さに基づいて、類似度を表現する (1) 式のパラメータの値を調整するための手法について述べる。ここでは、形状からサンプリングした n 点を用いて、形状をセグメンテーションする場合について、(1) 式のパラメータ α_p, α_T そして α_C を調整するための手法を説明する。

サンプリング点の集合について、(1) 式により生成したグラフ Laplace 行列の固有ベクトルなどを基準として、評価関数を最適化することによりパラメータを探索する。提案する評価関数は、

$$f(\mathbf{p}) = \underbrace{g_E(\mathbf{p})}_{\text{eigenvector range}} + \underbrace{g_M(\mathbf{p})}_{\text{more eigenvector range}} + \underbrace{g_R(\mathbf{p})}_{\text{parameter range}} + \underbrace{g_N(\mathbf{p})}_{\text{parameter norm}} + \underbrace{g_Z(\mathbf{p})}_{\text{zero eigenvector element}} \quad (2)$$

である。ここで、 $\mathbf{p} = \{\alpha_p, \alpha_T, \alpha_C\}$ は、調整しようとするパラメータを要素とするベクトルである。また、 $g_E(\mathbf{p})$, $g_M(\mathbf{p})$, $g_R(\mathbf{p})$, $g_N(\mathbf{p})$ そして $g_Z(\mathbf{p})$ は、それぞれ、クラスタリングに用いた固有ベクトルの要素の値の範囲、クラスタリングに用いない固有ベクトルの要素の値の範囲、パラメータを探索する値の範囲、パラメータベクトルのノルム、そして、値が 0 である固有ベクトルの要素の個数に関する項である。また、 p 以外のパラメータには、予め値が与えられているものとする。

以降において、(2) 式の右辺の各項について述べる。始めに、用いる記号について説明する。Fiedler ベクトルを $\mathbf{v}_f = \{v_1, v_2, \dots, v_n\}$ 、そして、Fiedler ベクトルに対応する固有値の値の次に値が大きい m 個の固有値に対応する固有ベクトルを $\mathbf{v}_{e_j} = \{v_{e_{j1}}, v_{e_{j2}}, \dots, v_{e_{jn}}\}$ ($j = 1, \dots, m$) とする。また、 \mathbf{v}_f の要素を、値が負、0、または正であるかによりクラスタリングすることにより生成されるクラスタを、それぞれ、 $\{C_{f1}\}$, $\{C_{f2}\}$ そして $\{C_{f3}\}$ とする。 \mathbf{v}_{e_j} についても、同様に生成されるクラスタを、 $\{C_{e_{j1}}\}$, $\{C_{e_{j2}}\}$ そして $\{C_{e_{j3}}\}$ とする。さらに、Fiedler ベクトル \mathbf{v}_f の要素の最小値と最大値を、それぞれ $v_{f(\min)}$ と $v_{f(\max)}$ とする。固有ベクトル \mathbf{v}_{e_j} についても同様とする。そして、クラスタ $\{C_{f1}\}$ の要素の最小値と最大値を、それぞれ $c_{f1(\min)}$ と $c_{f1(\max)}$ とする。他のクラスタについても同様とする。

これらを用いて、(2) 式の右辺の各項を、以下のように計算する。

$$g_E(\mathbf{p}) = \frac{c_{f1(\text{range})} + c_{f3(\text{range})}}{v_{f(\max)} - v_{f(\min)}} \quad (3)$$

$$c_{f1(\text{range})} = c_{f1(\max)} - c_{f1(\min)} \quad (4)$$

$$c_{f3(\text{range})} = c_{f3(\max)} - c_{f3(\min)} \quad (5)$$

$$g_M(\mathbf{p}) = \sum_{j=1}^m \frac{c_{e_{j1}(\text{range})} + c_{e_{j3}(\text{range})}}{v_{e_j(\max)} - v_{e_j(\min)}} \quad (6)$$

$$c_{e_{j1}(\text{range})} = c_{e_{j1}(\max)} - c_{e_{j1}(\min)} \quad (7)$$

$$c_{e_{j3}(\text{range})} = c_{e_{j3}(\max)} - c_{e_{j3}(\min)} \quad (8)$$

$$g_R(\mathbf{p}) = \sum_{k=1}^3 \begin{cases} -\exp(-\frac{\gamma_L - p_k}{\gamma_S}) & \text{if } p_k < \gamma_L, \\ \exp(\frac{p_k - \gamma_U}{\gamma_S}) & \text{if } \gamma_U < p_k. \end{cases} \quad (9)$$

$$g_N(\mathbf{p}) = \gamma_N |\mathbf{p}|. \quad (10)$$

$$g_Z(\mathbf{p}) = \gamma_Z (|C_{f_2}| + \sum_{j=1}^m |C_{e_{j_2}}|). \quad (11)$$

ここで, $\gamma_L, \gamma_U, \gamma_S, \gamma_N$ そして γ_Z は, 予め与えられた値である. なお, ここでは, 形状をセグメンテーションするためのクラスタリングにおいて, Fiedler ベクトルのみを用いている.

4. 実装と結果の検討

4.1 実装

提案のアルゴリズムを, Java 言語により実装した. 形状の表現には, 平面上の 3 次 Bezier 曲線を用いた. そして, それらの曲線のグラフィックスデータとしての表現形式には, SVG を用いた.

なお, 図中の形状および印は, 実装プログラムで調整した重みパラメータを用いたセグメンテーションの結果と固有ベクトルなどを, その実装プログラムから SVG 形式としてファイル出力したものである. 本稿では, その SVG 形式のファイルを EPS 形式に変換したものをを用いている.

4.2 セグメンテーションの各段階

形状セグメンテーションのためのアルゴリズムでの各段階を, 図 1 に示す. 図の始めの行は, 基準とする形状であり, 3 次 Bezier 曲線で構成されている. 次に, 図の 2 番目の行では, この基準形状を表現する曲線の全長を 50 等分した値をサンプリング間隔として, [7] で述べた方法により, 55 個のサンプリング点 (solid blue circle) を求めている. そして, 図の 3 番目の行では, サンプリング点をクラスタリングすることにより, サンプリング点を形状セグメント 1 (open blue circle) と 2 (green cross) に分類している.

ここでは, 3.1 節で提案したグラフの重みを用いて, サンプリング点をノードとするグラフを作成し, その Laplace 行列の固有値と固有ベクトルを基にしてクラスタリングしている. その Laplace 行列の固有値と固有ベクトルを, 図 2 に示す. なお, 第 2 固有ベクトルを用いて, クラスタリングしている. また, グラフの重みの計算に用いたパラメータを表 2 に示す. ここでのパラメータ値は, アルゴリズムにより調整したものではなく予め決定しておいた値を用いている. なお, $\beta_C = 0.1, \beta_U = 1$ であることから分かるように, 曲率の特徴量として, 符号付き曲率を用いている.

この結果から, クラスタリングにより 2 つのクラスに分類されたサンプリング点の集合は, それぞれが, 形状の曲線部分または直線部分からサンプリングしたものである

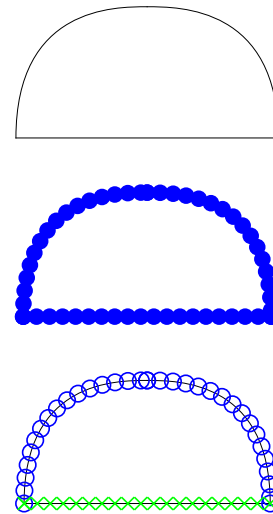


図 1 形状セグメンテーションの各段階

Fig. 1 Each step of shape segmentation. First row: a base shape is denoted by solid curves. Second row: the sampled points are denoted by solid blue circles. Third row: the clustered sampling points are denoted by: open blue circles the points included in shape segment 1; green crosses the points included in shape segment 2.

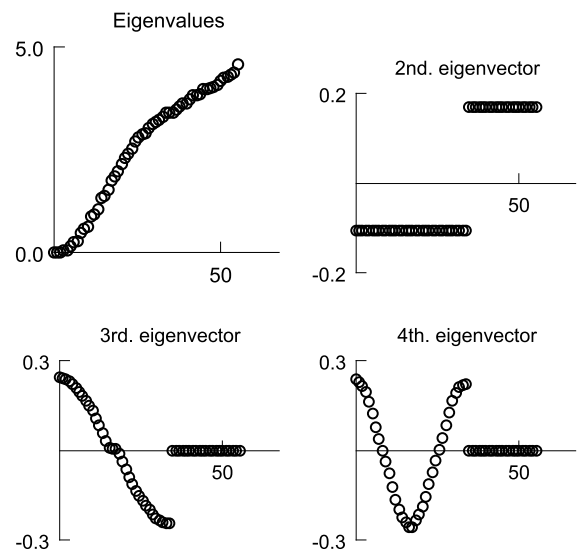


図 2 図 1 のセグメンテーションでの固有値と固有ベクトル

Fig. 2 Eigenvalues and eigenvectors of the graph Laplacian used in the segmentation presented in Fig. 1. The 2nd. eigenvector has been used for shape clustering.

表 2 図 1 のセグメンテーションで用いたパラメータ

Table 2 The parameters used in the segmentation in Fig. 1

α_P	α_T	α_C	α_U	β_P	β_T	β_C	β_U
0.1	0.33	0.33	0.1	0.1	0.1	0.1	1

ことが分かる．したがって，これらのクラスタリングしたサンプリング点は，形状を曲線部分と直線部分の2つのセグメントに分類するために用いることが可能であるといえる．

4.3 定数値を与えたパラメータ

これ以降では，(1) 式のパラメータ α_P, α_T そして α_C を調整する．そこで，アルゴリズムの入力として定数値を与えたパラメータを，表 3 に示す．

表 3 定数値を与えたパラメータ

Table 3 The parameters that are given constant values.

α_U	β_P	β_T	β_C	β_U
0.1	0.1	0.1	0.1	1
γ_L	γ_U	γ_S	γ_N	γ_Z
0.001	100	1	1	1

4.4 4.2 節で用いた形状でのパラメータの調整

4.2 節で用いた形状について，提案の評価関数 ((2) 式) を用いて求めたパラメータを，表 4 に示す．これらの値は，最適化計算の結果を小数点第 4 位で四捨五入した値である．

図 3 に，形状セグメンテーションの結果を示す．この例では，評価関数において，クラスタリングに用いない固有ベクトルの要素の値の範囲を調整するための項 ((6) 式) を用いていない．図の始めの行は，それぞれのサンプリング点についてのセグメンテーションの結果である．ここでは，サンプリング点を形状セグメント 1 (open blue circle) と 2 (green cross) に分類している．図の 2 番目の行は，グラフ Laplace 行列の第 2 固有ベクトルと第 3 固有ベクトルである．形状セグメンテーションにおいては，第 2 固有ベクトルを用いている．なお，サンプリング数は，174 である．

図 4 は，評価関数において，クラスタリングに用いない固有ベクトルの要素の値の範囲を調整するための項 ((6) 式) で $m = 1$ として，第 3 固有ベクトルを用いた結果である．なお，形状セグメンテーションにおいては，第 2 固有ベクトルを用いている．

これらの結果から，形状の曲線部分と直線部分に合わせて，サンプリング点がクラスタリングされていることが分かる．アルゴリズムにより調整したパラメータを用いる場合においても，4.2 節で示した形状セグメンテーションと同様の結果が得られているといえる．また，図 3 と 4 を比較すると，評価関数において，クラスタリングに用いない固有ベクトルの要素の値の範囲を調整するための項を用いない場合でも，第 3 固有ベクトルの範囲が調整されていることが分かる．

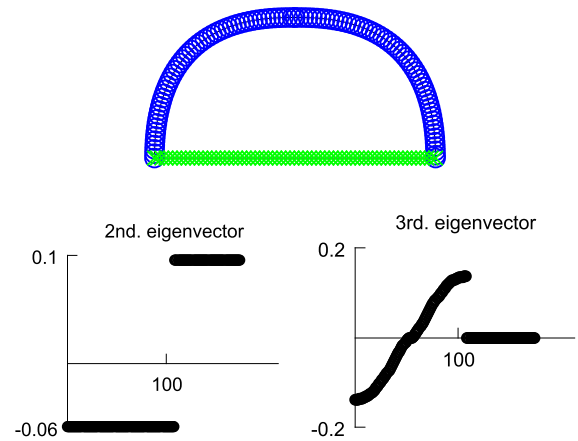


図 3 4.2 節で用いた形状でのパラメータの調整 ($g_M(\mathbf{p})$ を使用しない場合)

Fig. 3 Adjustment of the parameters for the segmentation of the shape used in subsection 4.2 ($g_M(\mathbf{p})$ does not be used.). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

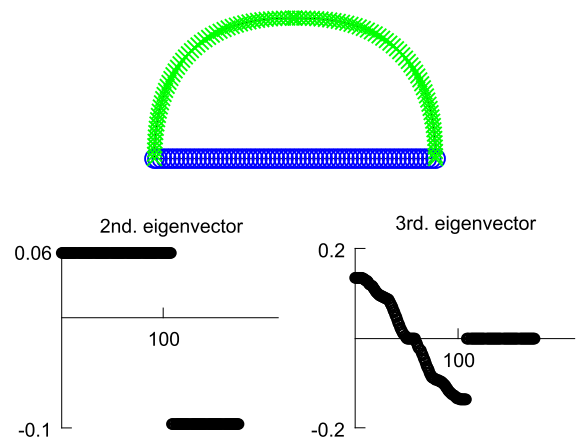


図 4 4.2 節で用いた形状でのパラメータの調整 ($m = 1$)

Fig. 4 Adjustment of the parameters for the segmentation of the shape used in subsection 4.2 ($m = 1$). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

表 4 図 3, 4 で用いた調整したパラメータ

Table 4 The adjusted parameters used in the segmentation in Fig. 3, 4. The parameters are rounded off to three decimal places.

figure	α_P	α_T	α_C
Fig. 3	0.141	0.234	0.338
Fig. 4	0.148	0.162	0.090

4.5 サンプリング数に関する頑健性

4.4 節で用いた形状についてサンプリング数を変化させた場合に、4.4 節で求めた類似度重みパラメータを用いて形状セグメンテーションをおこなった結果を、図 5, 6 に示す。サンプリング数は、それぞれ 255 と 155 である。

これらの結果では、サンプリング数が 255 の場合には、良好なセグメンテーションがおこなわれているが、サンプリング数が 155 の場合には、良好なセグメンテーションがおこなわれていないことが分かる。

4.6 円形のセグメンテーション

図 7 に、ほぼ円形の形状の形状セグメンテーションの結果を示す。表 5 が、調整したパラメータの値である。サンプリング数は、76 である。半径の異なる 2 つの円形からのサンプリング点が、それぞれ、形状セグメントに分類されていることが分かる。

4.7 交差する形状のセグメンテーション

形状が交差している部分を含む形状について、パラメータを調整して形状をセグメンテーションする例を、図 8, 9, 10, 11 に示す。サンプリング数は、それぞれ、33,104,105 そして 103 である。また、用いたパラメータを、表 6 に示す。アルゴリズムにより調整した類似度重みのパラメータを用いて、交差する形状を分離して、セグメンテーションが可能であることが分かる。

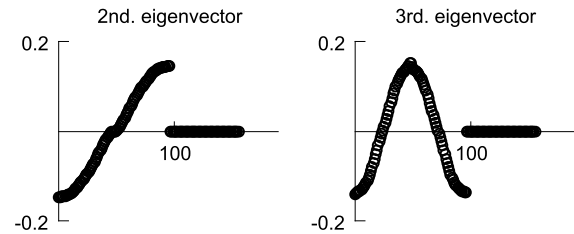
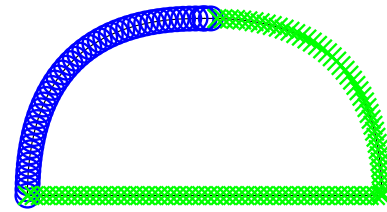


図 5 サンプリング数に関する頑健性 (サンプリング数 255)
Fig. 5 Robustness in terms of sampling size (sampling size 255). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

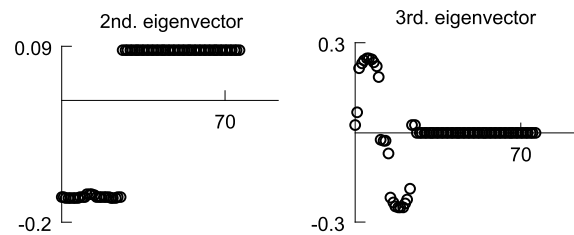
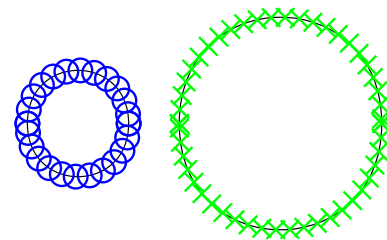


図 7 円形のセグメンテーション
Fig. 7 Segmentation of circular shapes. First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

表 5 図 7 で用いた調整したパラメータ

Table 5 The adjusted parameters used in the segmentation in Fig. 7. The parameters are rounded off to three decimal places.

α_P	α_T	α_C
0.100	0.104	0.133

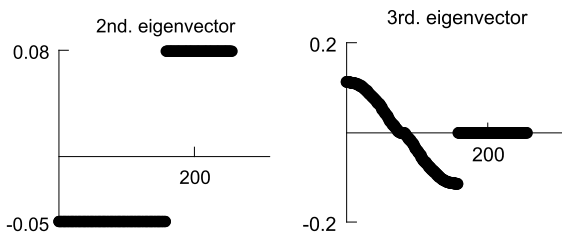
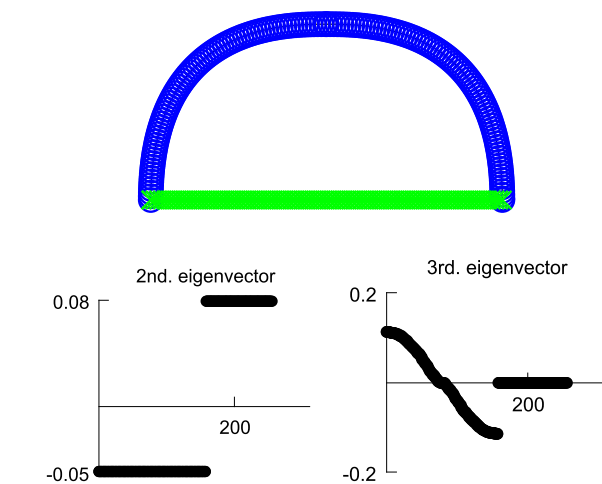


図 6 サンプリング数に関する頑健性 (サンプリング数 155)
Fig. 6 Robustness in terms of sampling size (sampling size 155). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

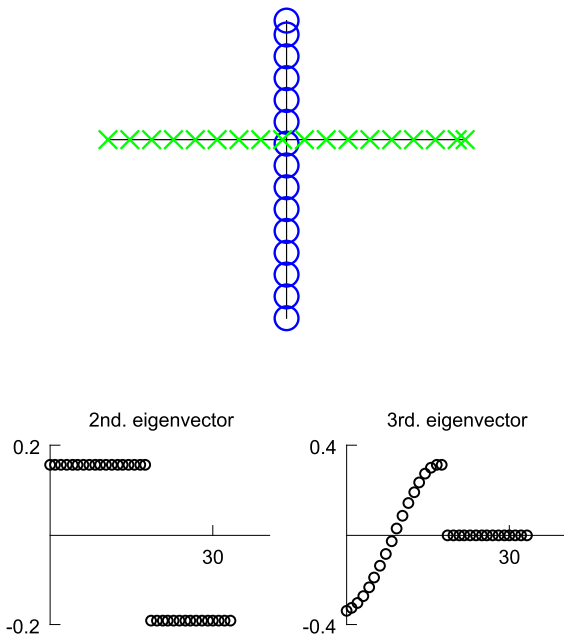


図 8 交差する形状のセグメンテーション (直線と直線)
Fig. 8 Segmentation of intersecting shapes (straight lines). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

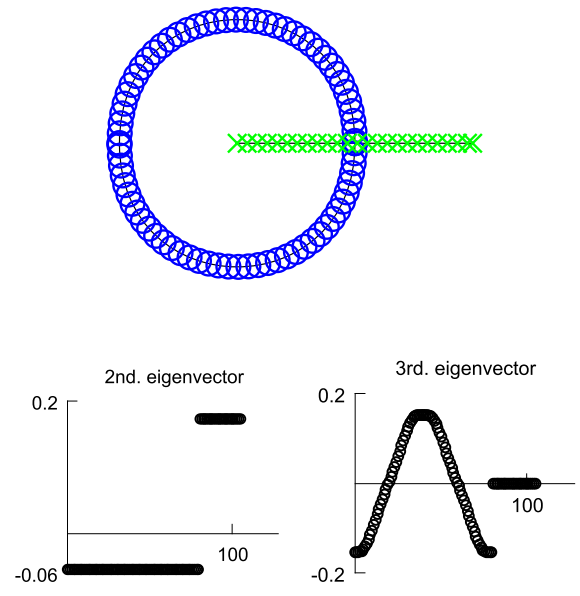


図 10 交差する形状のセグメンテーション (直線と円形)
Fig. 10 Segmentation of intersecting shapes (a straight line and a circular shape). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

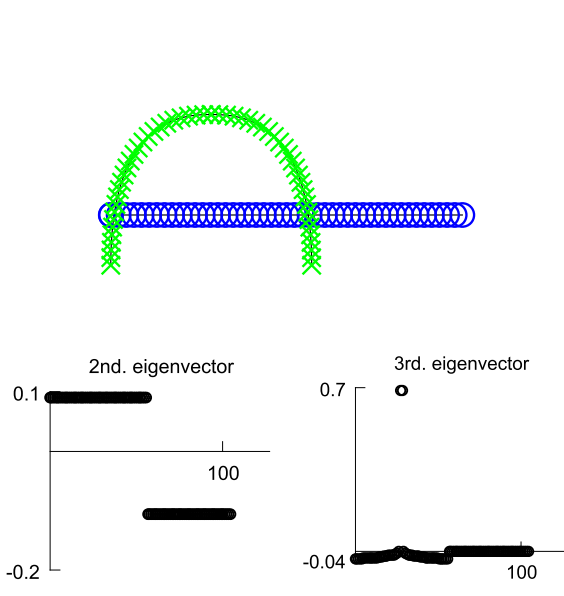


図 9 交差する形状のセグメンテーション (直線と曲線)
Fig. 9 Segmentation of intersecting shapes (a straight line and a curve). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

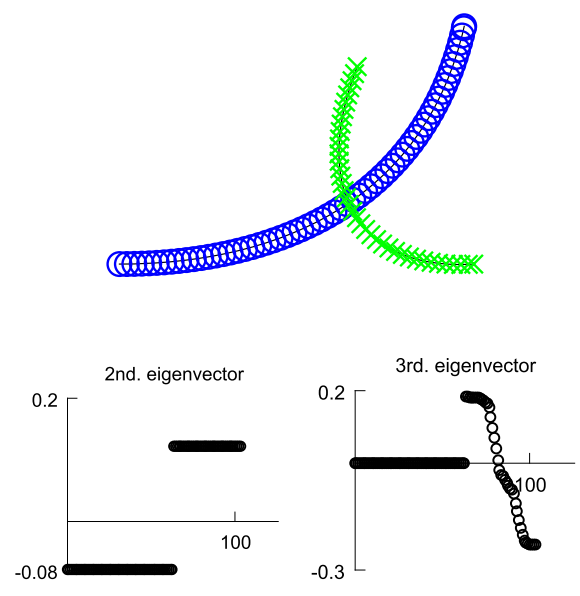


図 11 交差する形状のセグメンテーション (曲線と曲線)
Fig. 11 Segmentation of intersecting shapes (curves). First row shows the segmentation result: shape segment 1 (open blue circles); shape segment 2 (green crosses). Second row shows the 2nd. and 3rd. eigenvectors of the graph Laplacian.

表 6 図 8, 9, 10, 11 で用いた調整したパラメータ

Table 6 The adjusted parameters used in the segmentation in Fig. 8, 9, 10, 11. The parameters are rounded off to three decimal places.

figure	α_p	α_T	α_C
Fig. 8	0.100	0.288	0.001
Fig. 9	0.001	0.262	0.254
Fig. 10	0.207	0.100	0.276
Fig. 11	0.061	0.100	0.001

5. おわりに

視覚的な自然さに基づいた形状セグメンテーションのための既提案の手法 [7], [8] について, 形状の類似度を表現する重みの計算に用いるパラメータを調整するための手法を提案した. その手法の特徴は, 形状を表現するグラフ Laplace 行列の固有ベクトルが明確に分離するように設計した評価関数を最適化することにより, パラメータを探索するという点である. そのため, 特定の形状に関する事前知識を必要としないという特徴も併せ持つ手法である.

そして, 提案の手法により調整したパラメータを用いて, 形状セグメントを生成することが可能であることを示した. 提案したパラメータ調整のための手法は, 特定の形状に関する事前知識を必要としないので, 形状を構成する Bezier 曲線のパラメータ以外の事前知識を必要としない既提案の形状セグメンテーション手法の特徴を活かした手法といえる. また, 本稿で示した例では, セグメンテーションに用いる固有ベクトルの要素の値の範囲を小さくするように最適化をすると, セグメンテーションに用いない固有ベクトルにも形状の特徴を表現する情報を含むようにすることが可能であった. セグメンテーションに用いない固有ベクトルについても, 評価関数で最適化をする必要があるのかどうかについては, さらに検討をおこないたい.

今後の課題には, 複雑な形状から形状セグメントを生成するための類似度重みのパラメータの調整, 提案した類似度重みの調整手法の他のグラフクラスタリング手法への適用, そして, 形状セグメンテーションと非均一相似変換を用いた機械学習のためのデータ生成に関する研究を挙げることができる.

参考文献

[1] Fiedler, M.: Algebraic connectivity of graphs, *Czechoslovak Mathematical Journal*, Vol. 23, No. 98, pp. 298–305 (1973).

[2] Forsey, D. R. and Bartels, R. H.: Hierarchical B-spline refinement, *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, ACM, pp. 205–212 (online), DOI: <http://doi.acm.org/10.1145/54852.378512> (1988).

[3] Fowler, B. and Bartels, R.: Constraint-based curve

manipulation, *Computer Graphics and Applications, IEEE*, Vol. 13, No. 5, pp. 43–49 (online), DOI: 10.1109/38.232098 (1993).

[4] Hammond, T. and Paulson, B.: Recognizing sketched multistroke primitives, *ACM Trans. Interact. Intell. Syst.*, Vol. 1, No. 1, pp. 4:1–4:34 (online), DOI: 10.1145/2030365.2030369 (2011).

[5] Katz, S. and Tal, A.: Hierarchical mesh decomposition using fuzzy clustering and cuts, *ACM SIGGRAPH 2003 Papers, SIGGRAPH '03*, New York, NY, USA, ACM, pp. 954–961 (online), DOI: 10.1145/1201775.882369 (2003).

[6] Lewis, J. P., Fong, N., XueXiang, X., Soon, S. H. and Feng, T.: More optimal strokes for NPR sketching, *Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia, GRAPHITE '05*, New York, NY, USA, ACM, pp. 47–50 (online), DOI: 10.1145/1101389.1101398 (2005).

[7] 佐藤 信: 非均一相似変換を用いた形状洗練化のためのグラフクラスタリングによる形状セグメンテーション, 情報処理学会研究報告, Vol. 2013-HCI-155, No. 7, pp. 1–8 (2013).

[8] 佐藤 信: 学習データ生成のための符号なし曲率を用いたグラフクラスタリングによる形状セグメンテーション, 情報処理学会研究報告, Vol. 2014-ICS-173, No. 6, pp. 1–8 (2014).

[9] 佐藤 信, 三輪譲二: 導関数ベクトルの非均一相似性制約に基づく曲線洗練化法, 情報処理学会研究報告, Vol. 2011-CG-142, No. 12, pp. 1–6 (2011).

[10] 佐藤 信, 三輪譲二: 平面曲線形状洗練化のための導関数ベクトルの非均一相似性制約を用いた鏡映対称変換, 情報処理学会研究報告, Vol. 2012-CG-146, No. 34, pp. 1–6 (2012).

[11] Mohar, B.: The Laplacian spectrum of graphs, *Graph Theory, Combinatorics, and Applications*, Wiley, pp. 871–898 (1991).

[12] Mohar, B.: Some applications of Laplace eigenvalues of graphs, *Graph Symmetry*, NATO ASI Series, Vol. 497, Springer Netherlands, pp. 225–275 (1997).