

制約付きドロネー三角形分割を用いた自由形状のイラスト美化手法 Freeform Illustration Beautification using Constrained Delaunay Triangulation

森本 有紀[†] 高橋 時市郎^{†‡}

Yuki MORIMOTO[†] and Tokiichiro Takahashi^{†‡}

[†] 東京電機大学

[†] Tokyo Denki University

[‡] UEI Research

[‡] UEI Research

E-mail: {yuki,toki}@vcl.im.dendai.ac.jp

概要

本稿では、自由形状のイラストをデザイン原理に基づき美化する手法を提案する。単純な描画を美化ルールに基づき美化する手法はこれまでに盛んに研究されてきた。しかし、それらの美化ルールの自由形状への適応は挑戦的な課題である。本手法では、制約付きドロネー三角形分割を用いて局所的に関連の強い形状を効率的に探索する。また、本手法では、平行・滑らかさ・形状類似性・共曲線性、のデザイン原理を美化ルールとして用い、自由形状を評価する。

本手法は、形状評価、拡散フィルタリング、形状フィッティングの三つのステップによって成り立つ。まず制約付きドロネー分割を行った入力に対して、デザイン原理に基づき形状を評価する。次に、形状評価値に、特徴保存の平滑化処理を行う。最後に、形状フィッティングでは、複数の異なるデザイン原理に対して形状エネルギーを最小化し、最適な形状を生成する。既存手法との比較アンケートからは、本手法はより適切な美化ができることがわかった。

1. はじめに

これまでに CG や HCI の分野で描画された内容を整形する、または美化するたくさんの研究が行われている [13,15]. これらの研究は主に 2 種類に分けられる。一つは線の平滑化に基づく方法、もう一つは 2D での線の関係に対して美化ルールを用いる方法である。ここで、描画内容の美化にあたり、多くの線の平滑化に基づく整形手法は、入力内容を高く信頼し入力内容を大きく変形しない想定である。また、2D の美化ルールを考慮した先行研究は、シンプルな図を対話的なスケッチシステムの中で美化するものが多い [2,5]. 本研究では、より積極的な変形を目指すために美化ルール、自由形状に対応するために線の平滑化に基づく整形手法のコンセプトを取り入れた新しい美化手法を提案する。美化ルールに関しては、本手法は [6] で記述されたデザイン原理から、特にイラスト (説明図) の描画に必

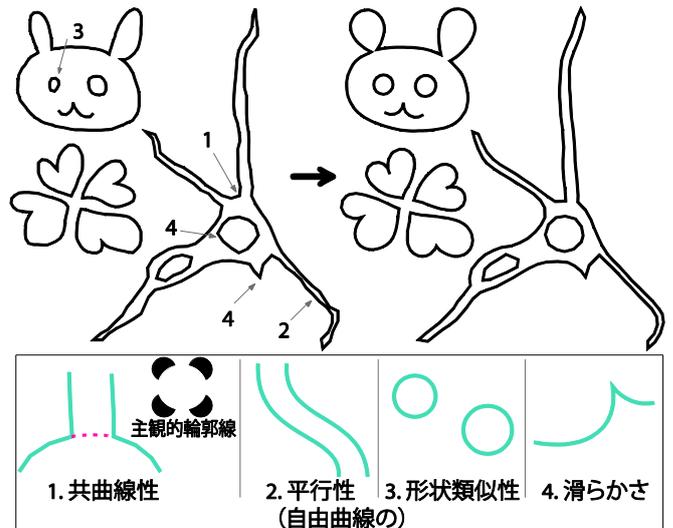


図 1: 入力したポリライン (左上), 本手法によって美化されたポリライン (右上), 考慮するデザイン原理 (下). なお、図は 3 つの画像に別々に美化を行った結果を集めたものである。

要なものとして、平行・滑らかさ・形状類似性・共曲線性を採用した。図 1 に本手法で扱うデザイン原理と本美化手法の結果を示す。

手で描画されたイラストや写真の輪郭線は多くの似た線分によって成り立つため、全探索で似た特徴の部分を見つけるのは難しい。イラストの美化での課題として、入力したベクター画像の中で関連する局所形状を探し、強調すべき特徴を決めることが挙げられる。

CDT (制約付きドロネー三角形分割: constrained Delaunay triangulation) は共曲線性や平行性のある頂点を連結しやすいため、本手法では制約付きドロネー三角形分割を入力画像に対して行い、2D 空間での連結関係を用いて、形状特徴の評価や平滑化に使用する。

本研究の主な新規性を以下にあげる。

- 2D の美化ルールと自由形状を整形する線の特徴保存平滑化手法を組み合わせる

- 2D の特徴評価にあたり，CDT による効率的な探索方法を提案する
- 事前計算やデータベースなどが不要である

表 1 は本手法と先行研究の扱う形状と基本アルゴリズムを比較するものである。

表 1: 対象形状と基本手法の先行研究との比較

手法	対象形状	基本アルゴリズム
線の平滑化系[1,8,9,13]	自由形状	特徴保存平滑化
2D 美化系[2,5,11] ¹⁾	プリミティブ(線, 楕円, 文字 ²⁾)	美化ルールに従った変形
[15]	自由形状	形状の平均化
本手法	自由形状	美化ルール+特徴保存平滑化

¹⁾ 例外として[11]では自由形状の連結のルールを扱う。

²⁾ 文字はプリミティブではないが、予め用意した形状を用いている。

2. 関連研究

前章で述べた、2D のシンプルな形状の美化に関する先行研究の例外として、[13]の手法がある。この研究では、複雑な手描きの内容を美化することができる。しかし完璧な形状、例えば完璧な円などに美化するのは手法の性質上難しい。

いくつかの対話的な描画整形手法ではクロソイド曲線(またはオイラー曲線)を使用している[1,8,9]。クロソイド曲線は美的曲線として知られ、曲率が線の長さに対して線形に変化する曲線である。これらの研究では、手描きの描画線を部分的なクロソイド曲線と弧、直線の三つで近似する。また、Thiel らは Elasticurves という描画速度によって描画線の平滑化の度合いを決める手法[13]を提案した。これら手法では、特徴を保持した線の平滑化が基本となっている。工業デザインの観点では、曲線のよさの分析やその生成について研究されているが、曲率の滑らかさに関するものが多い[14]。

本手法も曲線の性質を滑らかにする先行研究と似たコンセプトに基づいている。しかしながら本手法では、魅力的なイラストは、曲率だけでなく、滑らかな複数種類の形状評価値の変化を持つ線によって構成されているという仮定に基づく。

本手法では、形状評価値の特徴保存平滑化に大竹らの非線形拡散[10]を参考にする。また、平滑化した値を用い、形状フィッティングを行う。この処理によって値を強調する効果があり、また、いくつかの先行研究で必要な線の分割をする必要がない。

共曲線性は錯視として知られる主観的輪郭線に似ている。これは独立した線や輪郭線が連結しているよ

うに見える現象である。曲線性は平行性や近接性などとともにラスター画像をベクトル化するために使われる特徴である[4]。Fu らは線画の静止画からその適切な描画順を自動で生成する手法を示した[3]。本手法では、認知的に連結している線を見つけるために、Fu らの Transition cost の計算方法をベースとしている。

3. 本手法

まず、ユーザのマウスによる描画または画像からの輪郭線抽出によって、2D の連続した点を入力する。次に以下の流れで処理を行う。

1. 再サンプリング：入力したポリラインは鋭角な特徴点間で等間隔になるように再サンプリングを行う。特徴点は閾値を下回る角度を持つ点とする。
2. CDT：ポリラインの頂点を入力として、CDTを行う。このとき、入力線の外側も処理の領域に含めるかどうか、選択する。
3. 形状評価：すでに述べた複数のデザイン原理によって各頂点で形状を評価する。
4. 拡散フィルタリング：形状評価値を重み付きフィルタリングで拡散する。
5. 形状の最適化：入力形状を拡散した形状評価値に当てはまるように最適化する。

1~5 はユーザが満足するまで繰り返す。このとき、鋭角の抽出は再サンプリングのためだけに行われ、拡散や最適化の処理には必要ない。

3.1 形状の評価方法

よいイラストには、局所的にデザイン原理を満たす部分がよく見られる。以下に図 2 に即して、それらの原理とその評価および評価値を特徴保存平滑化する隣接空間を示す。

- A) 滑らかさ(鋭さ)：曲率として、入力線上で評価および特徴保存平滑化を行う。
- B) 形状の類似性：曲率の絶対値として、入力線上で評価し、入力線と CDT エッジ上で特徴保存平滑化を行う。
- C) 共曲線性：曲率として、CDT エッジ上で評価され、抽出された共曲線部分で特徴保存平滑化を行う。
- D) 平行性：距離として、CDT エッジ上で評価され、入力線上でフィルタリングを行う。

形状評価は、2D の連結した各頂点 v_i に対して行う。このとき $i=1,2,3,\dots,m$ で m は頂点の総数である。 $N_{all}(i)$, $N_{in}(i)$, $N_{CDT}(i)$ は v_i に連結する頂点の集合であり、それぞれ、連結する全ての頂点、入力線で連結する全ての頂点、CDT で連結する全ての頂点、とする。 N_{all} は N_{in} と N_{CDT} の和集合である。共曲線性および平行性は、これらの特徴の高い部分を見つけ出し、それぞれ $N_{co}(i)$,

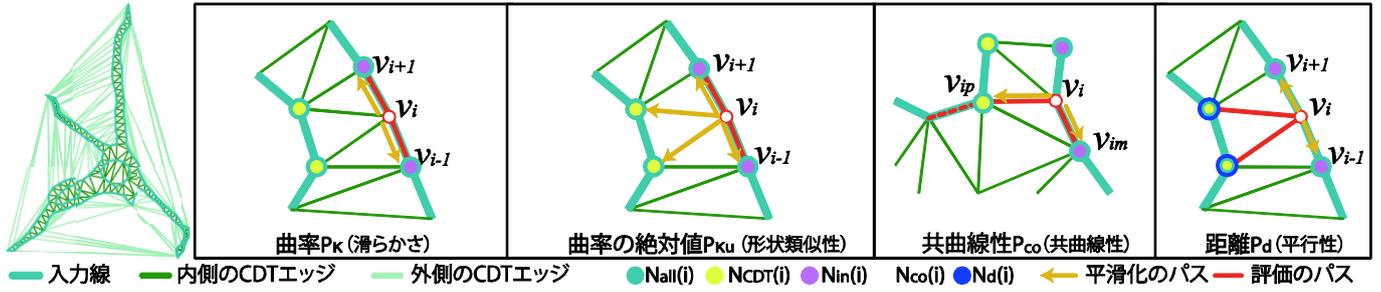


図 2: CDT を用いた形状特徴の評価とフィルタリングのための頂点 v_i におけるパス。括弧内は対応するデザイン原理を示す。赤い点線は同じ共曲線部分の残りの頂点を示す。

$N_d(i)$ として処理に適した近傍頂点の集合を抽出する。

3.1.2 曲率

各頂点の曲率は3点 v_{i-1}, v_i, v_{i+1} の外接円の半径の逆数として、正弦定理より、離散曲率 $P_k(i) = 2|\overline{v_i v_{i+1}} \times \overline{v_i v_{i-1}}| / (|v_i v_{i+1}| |v_i v_{i-1}| |v_{i+1} v_{i-1}|)$ と計算する。ここで、 \times は外積を表し、 $P_k(i)$ は形状の凹凸方向を示す符号付きスカラー値である。形状の見かけは符号なし曲率 $P_{ku}(i) = |P_k(i)|$ によるので、 $P_{ku}(i)$ は形状の類似性の評価に適している。

3.1.2 共曲線性

まず全ての頂点で CDT エッジのみで連結した近傍に対し連結スコアを計算し、次に条件を満たす部分を共曲線として求める。Fu らの Transition cost の計算[3]を参考に、連結スコアは $connect(i, j) = (\theta_i - \theta_j)^2 |v_i - v_j|, j \in N_{CDT}(i)$ で計算する(図3左)。このとき、 v_i と CDT エッジのみで連結する頂点を v_j とし、角度 θ_i は $N_{in}(i)$ の頂点、 $v_i, N_{CDT}(i)$ の頂点の角度であり、角度 θ_j も同様の v_j における角度である。

このとき以下を一つでも満たす v_j は共曲線から除外し、残りを共曲線として抽出する。1. $|v_i - v_j| > thr_{close}$, 2. $\angle v_i > thr_{sharp}$, 3. $connect(i, j) > thr_{detect}$, 4. v_i に複数の共曲線がある場合、連結スコアが最小のもの以外。

$\angle v_i$ は入力線上で連続する v_{i-1}, v_i, v_{i+1} の角度である。また、複数の入力した線の間での CDT エッジをここでの計算対象にするかどうかをユーザが選ぶ。残った共曲線部分は処理に必要な近傍を含めた頂点番号の集合 $c(k, l) \in k = 1, 2, 3, \dots, r, l = 1, 2, 3, 4$ として格納する。一つの集合は平滑化のため2点の曲率が必要なため、4点が必要であり、 l は4つとする。 r は抽出する共曲線の数である(図3右)。 $N_{co}(i)$ は $c(k, l)$ から i の両隣を抽出したものである。本稿の結果生成では $thr_{close} = 10$ (pixel), $thr_{sharp} = 0.8\pi$ (radian), $thr_{detect} = 35$ としている。共曲線性は共曲線部分の曲率として $P_{co}(i) = 2|\overline{v_i v_{ip}} \times \overline{v_i v_{im}}| / (|v_i v_{ip}| |v_i v_{im}| |v_{ip} v_{im}|)$ と計算する。このとき $im = c(k, l - 1), i = c(k, l), ip = c(k, l + 1)$ である(図3右)。

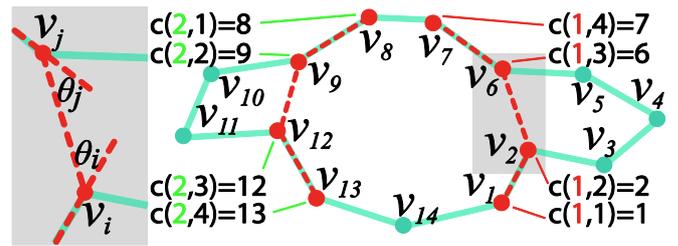


図 3: 左図は連結スコアの計算、右図は共曲線部分の頂点番号の集合 $c(k, l)$ の例を示す。ここで $r=2$ である。

3.1.3 距離

距離は平行性を高めるために評価する。まず、平行な近傍頂点の集合 $N_d(i)$ を $N_{CDT}(i)$ から、以下を満たす頂点として見つける(図4)。

$$I = \begin{cases} |\angle \overline{v_i O_i} \overline{v_i v_j}| & (|\angle \overline{v_i O_i} \overline{v_i v_j}| < \pi/2) \\ \pi - |\angle \overline{v_i O_i} \overline{v_i v_j}| & (\text{上記以外}) \end{cases}$$

$$\angle J = \begin{cases} |\angle \overline{v_j O_j} \overline{v_j v_i}| & (|\angle \overline{v_j O_j} \overline{v_j v_i}| < \pi/2) \\ \pi - |\angle \overline{v_j O_j} \overline{v_j v_i}| & (\text{上記以外}) \end{cases}$$

$$\angle 0 = \begin{cases} |\angle \overline{O_i v_i} \overline{O_j v_j}| & (|\angle \overline{O_i v_i} \overline{O_j v_j}| < \pi/2) \\ \pi - |\angle \overline{O_i v_i} \overline{O_j v_j}| & (\text{上記以外}) \end{cases}$$

のとき、 $\angle I < thr_{dist}$, $\angle J < thr_{dist}$, $\angle 0 < thr_{dist}$,

図の O_i は v_{i-1}, v_i, v_{i+1} の外接円の中心であり、 O_j も同様である。本稿では、 $thr_{dist} = \pi/3$ (radian) とした。距離 $P_d(i)$ は頂点 v_i と $N_d(i)$ に含まれる頂点 v_j のユークリッド距離の平均として以下の式で計算する。

$$P_d(i) = \sum_{j \in N_d(i)} \frac{|v_i - v_j|}{|N_d(i)|}$$

このとき $|N_d(i)|$ は $N_d(i)$ の頂点数を示す。

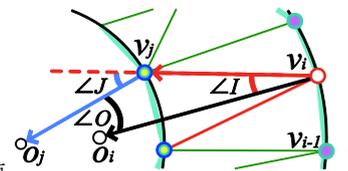


図 4: 平行な頂点の抽出

3.2 特徴保存平滑化

本ステップでは Ohtake らの非線形平滑化[10]を法線ではなく本手法で評価した値に適応し、値がその近傍の頂点と近い場合平滑化され、逆に値が近傍のものと差がある場合はそのまま残される。また、鋭角な頂

点是最適化の処理と合わせて強調され、元より値が大きくなる傾向にある。例えば、曲率の場合、更に鋭角になる（図 5）。処理は次式に従う。

$$P'_x(i) = \sum_{j \in N_x(i)} \omega(i,j) P_x(i) / \sum \omega(i,j), \quad \omega(i,j) = \exp(-\alpha K^2),$$

$$K = |P_x(i) - P_x(j)| / |v_i - v_j|, \quad (x \in \{\kappa, co, d, \kappa_u\})$$

このとき $P'_x(i)$, $P_x(i)$, α は平滑化された値、評価値、平滑化の調整パラメータ（任意の正の定数）である。

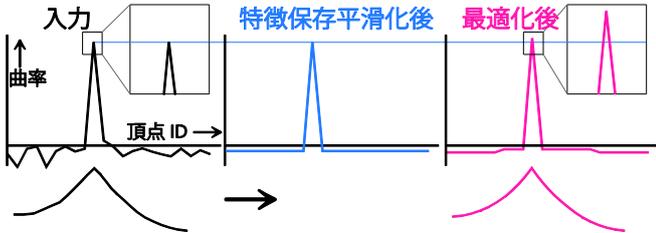


図 5: 特徴保存平滑化と最適化によって強調される形状評価値の例。黒線は入力線とそれに一致する曲率を示す。青線は特徴保存平滑化された曲率、ピンクの線は平滑化された値に最適化した形状とその曲率である。

3.3 形状の最適化

最適化では曲率、距離、共曲線性、類似性のエネルギーを $E_\kappa, E_d, E_{co}, E_{\kappa_u}$ を以下のように定義する。

$$E_\kappa = \sum_{v_i \in V} (P_\kappa(i) - P'_\kappa(i))^2, \quad E_d = \sum_{v_i \in V} (P_d(i) - P'_d(i))^2,$$

$$E_{co} = \sum_{v_i \in V_{co}} (P_{co}(i) - P'_{co}(i))^2, \quad E_{\kappa_u} = \sum_{v_i \in V} (P_{\kappa_u}(i) - P'_{\kappa_u}(i))^2.$$

また、入力したポリラインの頂点位置と頂点間隔のエネルギー E_p および E_{in} を他のエネルギーと同様に定義する。ここで $\{V\} = \{v_i\}$ は頂点の集合、 $\{V_{co}\} = \{v_{c(i,j)}\}$ は各共曲線部分とすると、全てのエネルギー E は次式で定義する。

$$E = [k_\kappa k_d k_{co} k_{\kappa_u} k_p k_{in}] [E_\kappa E_d E_{co} E_{\kappa_u} E_p E_{in}]^T$$

このとき、最適化の係数 $k_\kappa, k_d, k_{co}, k_{\kappa_u}, k_p, k_{in}$ はユーザによる任意の値である。美化後の頂点 $\{V'\} = \{v'_i\}$ は $V' = \operatorname{argmin}_{V'} E$ によって求める。ここでは非線形最小二乗問題を解く手法である Levenberg-Marquardt 法[7]によって $\sum_{i=1}^m \|v_i - v'_i\| < \epsilon$ となった場合に計算を収束する。

4. 結果

提案アルゴリズムは Windows Vista デスクトップ PC, Intel Core i7 3.2 GHz processor, 3.0 GB メモリで実装されている。実験での平滑化と最適化を合わせた 1 回の処理は本稿の結果生成に関して 0.2 秒以内であったが、計算時間は入力頂点数と適用する美化ルールの数に依存することが観察された。表 2 に結果と対応する入力超点数、計算時間および実験に関するパラメータの設

定を示す。

開曲線の平行性 (設定 A, B, C) この実験は、ほとんど平行な 2 本の曲線に対して設定 A, B, C を用いて行った（図 6）。曲率の最適化を行う設定 A では、入力線が十分滑らかであることから、ほとんど変形しなかった。距離の最適化を行う設定 B では、美化後の距離がほぼ等しくなったが、滑らかでないため、見かけの平行性は改善されていない。曲率と距離の最適化を行う設定 C では、距離と曲率の両方が滑らかになり、平行な見かけが得られた。

共曲線性 (設定 F, G, H) うさぎの顔のシルエットのような入力線に対し、共曲線性に関して設定 F, G, H を用いて実験を行った（図 9）。曲率の最適化を行う設定 F では、曲率が滑らかになり、角が鋭角になっているが、耳の根本部分の共曲線の滑らかさは失われた。共曲線性の最適化を行う設定 G では共曲線部分のみが滑らかになったが、共曲線性と曲率の両方の美化を行う設定 H では、共曲線部分とその他の部分が滑らかに連結された結果となった。

形状類似性 (設定 I, J) 半径の異なる手描きの 2 つの丸を入力線として設定 I, J を用いて実験を行った（図 10）。曲率の最適化を行う設定 I は単に滑らかな結果を生成するのに対し、曲率の絶対値の最適化を行う設定 J では、滑らかになると同時に、2 つの丸がほぼ同じサイズとなった。曲率の絶対値の最適化は離れた領域の形状の類似性を高めることがわかった。

ベクター画像への適用 図 1 のニューロンは写真から輪郭線を抽出したものに本手法を適用したものである。本手法を写真に適用する際、本手法の前準備として、OpenCV の `cv::findContours` により、写真から輪郭線を抽出した。また、閉曲線の平行性の実験と同様に、表 2 に示された複数の設定をユーザが適宜選択して適用した。結果画像はデザイン原理に基づき美化されていることがわかる。

比較 図 9 は先行研究との結果の比較である。先行研究の代表として、クロソイド曲線によるフィッティングを用いた先行手法[1]と本研究による結果を比較した。本手法が形状の平滑化と強調を適切に行っていることを確かめるため、先行研究による結果を 2 種類用意した。一つはより平滑化されたもの、一つはより詳細を残したものである。本稿の Baran ら[1]による結果では曲線補間によって解像度が変化しているが、本手法では元の解像度を保ったため、Baran らの結果

は本手法の3倍以上の解像度である。しかしながら結果からは、本アルゴリズムはイラストの美化において有効であることがわかる。他に Adobe Illustrator を用いた比較実験結果でも同様の結果が得られた。

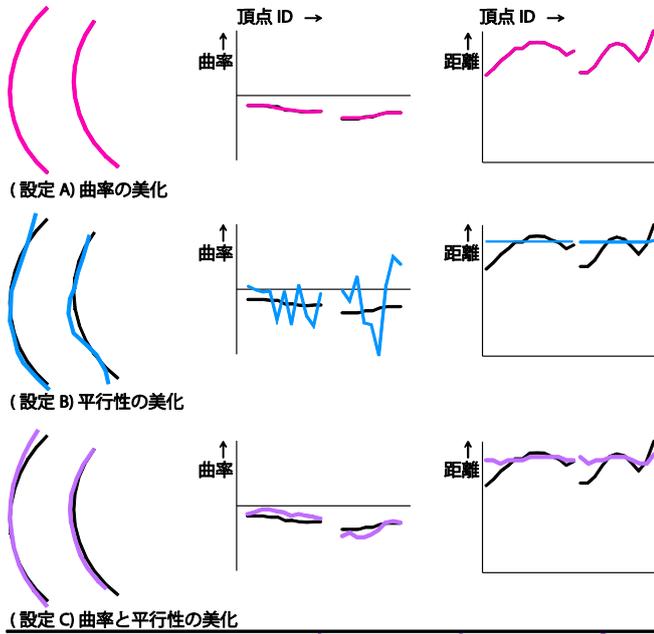


図 6: 平行性美化の実験結果と比較 (上図) と曲率と平行の美化における処理回数による形状の変異(下図)。上図の黒線は入力線, 中と右の列のグラフはそれぞれ頂点の順 (描画順) の曲率と距離を示す。

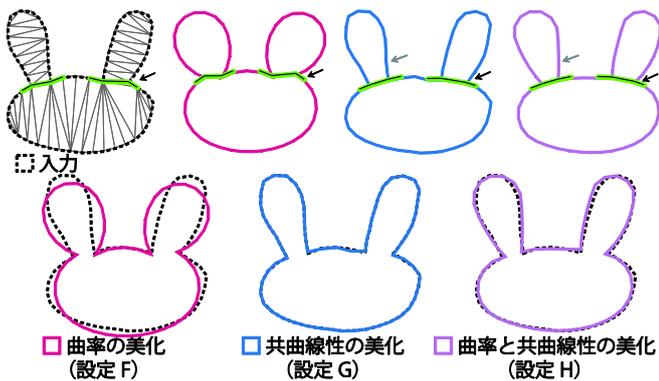


図 7: 共曲線性美化の実験結果と比較. 設定 F, G, H を用い, 滑らかさ, 共曲線性, 滑らかさと共曲線性の両方, を最適化した結果である. 黒い実線は CDT のエッジ, 緑の線は共曲線として抽出された部分である。

アンケート評価 比較結果に対し主観評価アンケートを行った結果では、「きれい・美しい」や「個人的

に好み」などの他, 特に「図として整っている」や「構成がわかりやすい」というイラストとして重要な項目にて, 本手法はよい評価を得た (アンケートの詳細および動画結果は添付資料を参照)。

考察と今後の課題 CDTはオブジェクトの領域分割に採用されるところなどからも, 共曲線部分を検出しやすいと言える[12]. また, 平行線においても, 他点を含まない外接円上の3点を連結する性質上, 平行線を連結する場合をよく観察できた. 本研究の実験において, 線の振動が非常に大きいなど以外では, あまり問題は観察されなかった. しかしながら, 必ずしも連結がうまくいく保証はない. 今後, 線の振動が大きい場合などにも対応したい。

サンプリング間隔に関しては, 大きくし過ぎると形状の特徴を捉えることができないが, 小さすぎると平滑化の影響範囲が小さくなり変形が小さくなる. サンプリング間隔により平滑化の影響範囲を変更するなどの対処が考えられる. 他に微妙な間隔の違いが変形の結果に大きく影響するなど課題である. 曲線を近似できる最低限の解像度で近似を行い, 曲線補間をするなどの対策が考えられる。

その他の課題として, 共曲線や平行の抽出の精度を上げることと, 最適化の要素同士の衝突を回避するアルゴリズムの開発が挙げられる. また, 最適化のパラメータの無次元化, 部分的に異なるパラメータを設定する, などもパラメータ調整を行いやすくする上での課題である。

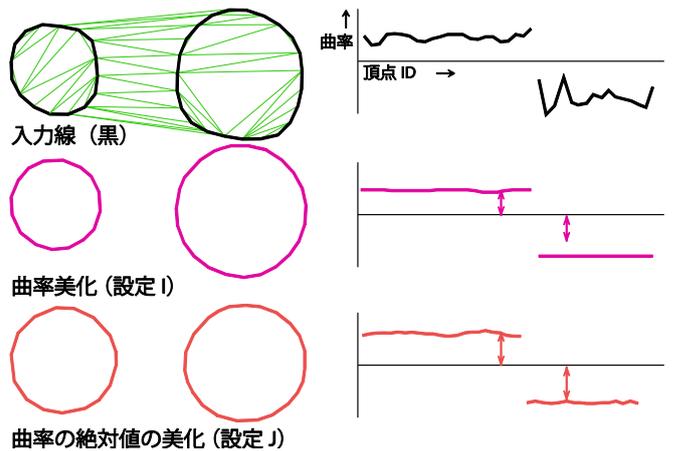


図 8: 類似性美化の実験結果と比較. 緑の線は CDT エッジを示す。

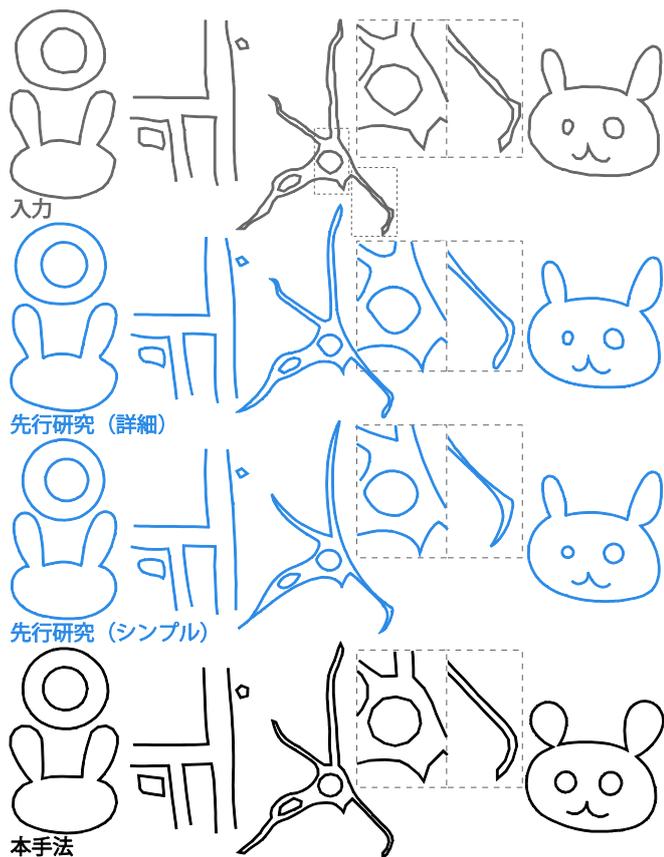


図 9: 先行研究との比較. 入力画像 (灰色), 手法[1]による詳細を残した結果 (青色上段), 同, 滑らかな結果 (青色下段), 本手法による結果 (黒色). 先行手法は本手法の3倍以上の解像度であるが, 本手法による平行, 共曲線, 形状類似の美化結果は顕著である.

5. まとめ

本稿では, 自由形状のイラストを美化する手法を提案した. また, 美化に必要な形状特徴の抽出に, CDTを用いた効果的な方法を提案した. 今後, 更に複雑なイラストを美化する手法へ発展したい.

文 献

[1] Baran, I., Lehtinen, J., Popovic, J.: Sketching clothoid splines using shortest paths. Eurographics

表 2: 実験に用いた設定. (/)は 0 を表す.

結果(1 ステップの計算時間/ ステップ数)	k_d	k_k	k_{co}	k_{k_u}	k_p	k_{in}	頂点数 頂点間隔の平均 平滑化の重み α
図 6	A (0.09sec /1)	/	10^3	/	/	/	24 20.13 900
	B (0.04sec /5)	10^{10}	/	/	/	/	
	C (0.06sec /5)	10^{-3}	10^3	/	/	/	
動画内	D (0.18sec /14)	/	10^8	/	/	/	57 24.93 900
	E (0.2 sec /5)	10^{10}	/	/	0.01	/	
図 7	F (0.16sec /40)	/	10^8	/	/	500	60 23.5 10000(共曲線性の平滑化には 1)
	G (0.19sec /4)	/	/	10^{10}	/	500	
	H (0.19sec /8)	/	10^8	10^{10}	/	500	
図 8	I (0.08sec /8)	D と同じ					39 14.52 9000
	J (0.08sec /27)	/	/	/	10^{10}	/	

2010 (2010)

[2] Cheema, S., Gulwani, S., LaViola, J.: Quickdraw: improving drawing experience for geometric diagrams. CHI '12, pp. 1037–1064. ACM (2012)

[3] Fu, H., Zhou, S., Liu, L., Mitra, N.J.: Animated construction of line drawings. ACM Trans. Graph. 30(6), 133:1–133:10 (2011)

[4] Hilaire, X., Tombre, K.: Robust and accurate vectorization of line drawings. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(6), 890–904 (2006)

[5] Igarashi, T., Matsuoka, S., Kawachiya, S., Tanaka, H.: Interactive beautification: a technique for rapid geometric design. UIST' 97. ACM (1997)

[6] Lidwell, W., Holden, K., Butler, J.: Universal Principles of Design. Rockport Publishers (2003)

[7] Madsen, K., Nielsen, H.B., Tingleff, O.: Methods for nonlinear least squares problems. Tech. rep. (2004)

[8] McCrae, J., Singh, K.: Sketching piecewise clothoid curves. SBIM '08.

[9] McCrae, J., Singh, K.: Neatening sketched strokes using piecewise french curves. SBIM '11, pp. 141–148. ACM, (2011)

[10] Ohtake, Y., Belyaev, A.G.: Nonlinear diffusion of normal for stable detection of ridges and ravines on range images and polygonal models. In: MVA, pp. 497–500 (2000)

[11] Pavlidis, T., Van Wyk, C.J.: An automatic beautifier for drawings and illustrations. SIGGRAPH Comput. Graph. 19(3), 225–234 (1985)

[12] Shapira, L., Shamir, A., Cohen-Or, D.: Consistent mesh partitioning and skeletonisation using the shape diameter function. Vis. Comput. 24(4), 249–259 (2008)

[13] Thiel, Y., Singh, K., Balakrishnan, R.: Elasticurves: exploiting stroke dynamics and inertia for the real-time neatening of sketched 2d curves. UIST '11, pp. 383–392. ACM (2011)

[14] Yoshida, N., Saito, T.: Interactive aesthetic curve segments. The Visual Computer 22(9-11): 896-905 (2006)

[15] Zitnick, C.L.: Handwriting beautification using tokens means. ACM Transactions on Graphics (TOG) 32(4) (2013)