

手書き図形入力のための時系列情報を利用した逐次清書法

松 田 浩 一[†] 近 藤 邦 雄[†]

手書き図形の入力は、コンピュータの入力インタフェースとして重要な機能と考えられ、さまざまな手法が提案されている。しかし、人が紙面上にスケッチを描くような試行錯誤ができるような手書き入力手法はない。そこで本論文では、コンピュータによる図形入力支援を行うために、スケッチに用いられる手法である線分の重ね書きによる入力をコンピュータ上で可能にする、逐次清書法を提案する。本手法は、手書き線群を時系列順に処理し動的に新たな清書線分を出力することにより対話的な線分入力を可能にする手法である。本論文では逐次清書法を用いて直線抽出、曲線抽出を行う。直線抽出のために線分混合法を提案し、また、曲線抽出のために特徴点結合法および節点分割抽出法を提案し、制御点の再構成法について提案を行った。既存の制御点生成アルゴリズムと比較を行った結果、より少ない制御点での曲線表現が可能であり、概形の再現性にも優れていることを示した。本手法を用いることにより、コンピュータ上で試行錯誤を行いながら清書線分を描くことが可能となり、(1) 重ね書きによる複数線分による入力を清書可能、(2) 対話的に試行錯誤を行いながら清書を行うことができる、(3) 同一のアルゴリズムを用いて清書線分に対して加筆・修正を行うことが可能、(4) 少ない制御点で入力曲線を表現可能である、という利点が得られる。

Interactive Sequential Drawing for Freehand Sketch Inputting

KOICHI MATSUDA[†] and KUNIO KONDO[†]

This paper introduces a new method to deal with sketch for inputting drawings using a computer with a stylus pen and a tablet. Many study have been developed for improving user interaction with a stylus on a computer, particularly in the conceptual design stage. These are only useful when the designer knows exactly what the object should look like. Our method affords the designer to draw overlapped segments like as drawing sketch. Characteristics of our research are as follows. (1) Making a clean copy from freehand lines in realtime, (2) Trial and error is possible, (3) Retouch is possible using a same algorithm, (4) Freehand curves can be expressed with a few control point. The aim of our method is to simplify the process of drawings for helping the designer. He can keep his drawing with a stylus on a tablet at the same manner using a pencil on paper.

1. はじめに

本論文の目的は、ペン入力による手書き入力インタフェースのための線分の重ね書きによる逐次清書法の提案である。本研究では、コンピュータ上でのスケッチ感覚でのペン入力による形状入力や形状修正を可能にすることを目標としている。

デザイナーが形状をイメージするときには頭の中に3次元形状がある。頭の中にあるイメージは直接ははっきりとした形として表すことができないため、概形を描き、徐々に具体化していくことが一般的に行われている。このとき、重ね書きによる描画は発想の段階で多く用いられる。この描画作業による思考をコンピュー

タ上で生かすため、ペンを用いた手書きによる図形入力のための試行錯誤が可能なインタフェースが望まれている¹⁾。

手書きによる図形入力の研究では、極大・極小・変曲点を用いた一筆書き曲線の清書²⁾がある。直接操作による曲線の形状変形が特徴であり、大域的な変形が可能である。また、手書きにより描かれた線や図形がどのような幾何形状であるかを識別する研究もあり、ファジィ理論を用いて補正する Chen ら³⁾の手法などがある。この研究では楕円や円弧や直線などを柔軟に認識することが可能である。そして、歪みを含んだ直線の補正を行う手法として、Ramer⁴⁾、近藤ら⁵⁾の手法がある。両者ともに、折れ線の端点を結び、端点を結ぶ線から距離最大点を選んで区間分割し直線近似する手法である。しかし、距離最大点の選び方に違いがあり、前者は端点を結ぶ直線からの距離を用いるのに

[†] 埼玉大学理工学研究科

Graduate School of Science and Engineering, Saitama University

対し、後者は端点間の距離と端点を結ぶ直線からの距離の比を用いている。上記の研究においては、描かれた線分をどのように処理するか主眼がおかれ、どのように描くかについては言及されていない。描き方に関する研究としては、拘束条件などを用いて清書図形を容易に描く五十嵐ら⁶⁾の研究がある。平行や垂直、対称性などを利用して自動的に次に描く候補を予想することにより図形描画を可能にしている。しかし、以上の手書き入力に関する研究は、いずれも描画する線が1本であり、部分的な変更をラフに描きながら試行錯誤することが不可能である。

そこで本論文では、スケッチの描画手法をコンピュータへの入力として扱うことを目的とした逐次清書法を提案する。本論文で提案する逐次清書法は、ラフに描かれた手書き線群を時系列順に処理し、動的に新たな清書線分を出力することにより対話的な線分清書を可能にする手法である。

本逐次清書法の特長は以下の4つである。(1) 重ね書きによる複数線分による直線・曲線入力が可能、(2) 対話的に試行錯誤を行いながら作画を行うことができる、(3) 同一のアルゴリズムを用いて抽出線分に対して加筆・修正を行うことが可能、(4) 少ない制御点で入力曲線を表現可能である。

本論文では、まず2章で逐次清書についての概要を述べる。3章において直線の逐次清書のための線分混合法について述べ、4章では曲線の逐次清書のための特徴点結合法と節点分割抽出法のアルゴリズムについて述べる。そして5章では、逐次清書法を用いた実験例を示し、比較を行った結果を述べる。

2. 逐次清書法の概要

逐次清書法はラフに描かれた線群から動的に直線および曲線を得る手法である。提案手法では、ユーザの定めた時間内に描画活動が行われない場合その時間を思考時間と見なし、その間に線群からの芯線抽出を自動的に行う。

逐次清書法は、図1のように構成される。ここで、本論文では直線と曲線の自動判別は行わず、どちらを入力するかはユーザが選択する。直線入力を行う場合、(1) 入力された線群は、(2) 節点抽出法⁵⁾により直線近似される。次に(3) 本論文で提案する線分混合法により直線抽出され、(6) 抽出結果を出力する。また、曲線入力の場合は、(2) で得られた制御点を用いて(4) 本論文で提案する特徴点結合法による制御点列生成を行い、(5) スムージングおよび制御点削減のために節点分割抽出法を用いて(6) 抽出結果を出力する。上

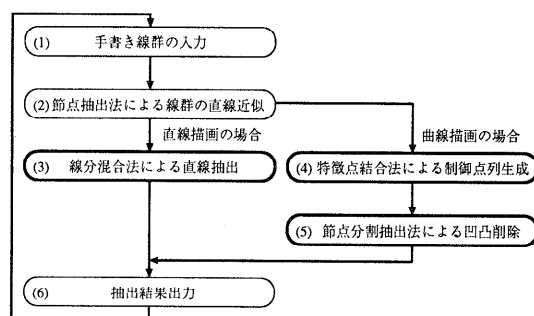


図1 逐次清書法の構成

Fig. 1 Component of the interactive sequential drawing.

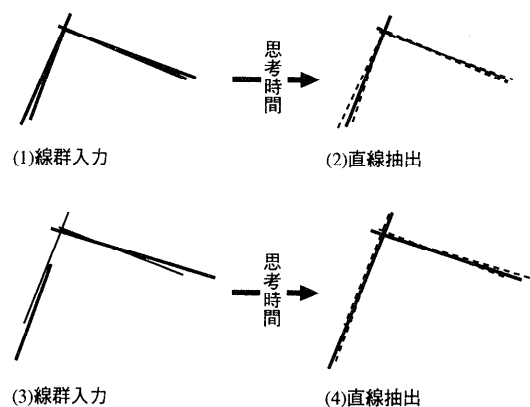


図2 逐次清書法の作業の流れ

Fig. 2 Process of the interactive sequential drawing.

記の直線手法と曲線入力においては、ともに抽出結果に対して繰り返し入力を行うことにより線分の修正を行うことができる。

図2に直線入力における作業の流れを示す。図2(1)は直線近似された入力線群を示し、思考時間と判定された場合には自動的に直線抽出が行われ、図2(2)の状態になる。図2(2)は、複数本あった線分から新たな直線が抽出された状態を示す。実線は抽出結果を表し、破線は抽出前の入力線群を表している。図2(3)は(2)で抽出された線分(図中の細線)に対し、直接修正線群を描画した(図中の太線)図であり、図2(4)は、修正線分からさらに抽出した結果を示す。

このように本手法では、その清書された図形に対してさらに修正を行うことができる。本手法はこのように入力作業を何度も繰り返し行いながら、重ね書きによる線群を扱うことが可能である。次章以降で直線・曲線それぞれの抽出手法について述べる。

3. 直線の抽出

本章では入力線群から直線を抽出するための線分混

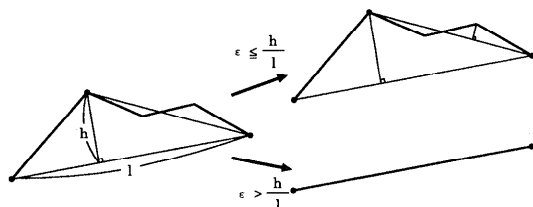


図3 節点抽出法のアルゴリズム

Fig. 3 Algorithm of the line approximate.

合法について述べる。

まずはじめに、入力線群に対し節点抽出法を用いて直線近似を行う。そして、直線近似された線群に対し本論文で提案する線分混合法を用いて直線抽出を行う。

3.1 節点抽出法による直線近似

入力線群は、手書きによる手振れを持っているため、線分混合の前処理として、入力線群の直線近似を行う必要がある。

直線近似を行う手法として、Ramer⁴⁾、近藤ら⁵⁾の手法が提案されている。本研究では図形の大きさに依存しないという点から、近藤らが提案した節点抽出法を利用する。

図3にそのアルゴリズムを示す。探索区間において、端点間の距離と、端点を結んだ線から最も遠い点と端点間を結ぶ直線からの距離の比が閾値 ϵ 以上の値が得られた場合に節点として残す。さらに再帰的に区間分割を行い、同様の探索を行う。

この節点抽出法を用い、手振れを持つ入力線分を直線近似し入力直線として扱う。

3.2 線分混合法による直線抽出

まず、線分に関する定義について述べ、次にそれらの定義を用いた線群からの直線抽出アルゴリズムを示す。

3.2.1 線分に関する定義

任意の2線分について、(1) 角度が近く、(2) 線分間の距離が近い場合は同一線分、と定義する。ここで、角度が近いとは、2本の直線の角度差が閾値より小さい線分とし(図4(1))、線分間の距離が近いとは、一方の線分の中点と他方の線分との距離が閾値より小さいこととする(図4(2))。図中の α , β は水平線からの角度、 d は2線分の距離を表す。以上の2条件を満たす場合に、同一直線を描こうとしたと判断する。なお、閾値はユーザが設定可能であり、その閾値により2線分が1本として描かれたか、2本として描かれたかを判定する。

3.2.2 線分混合アルゴリズム

直線抽出のための線分混合アルゴリズムは以下のと

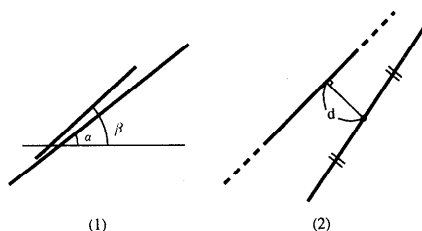


図4 角度と距離を用いた同一線分の判定

Fig. 4 Judgement of same lines.

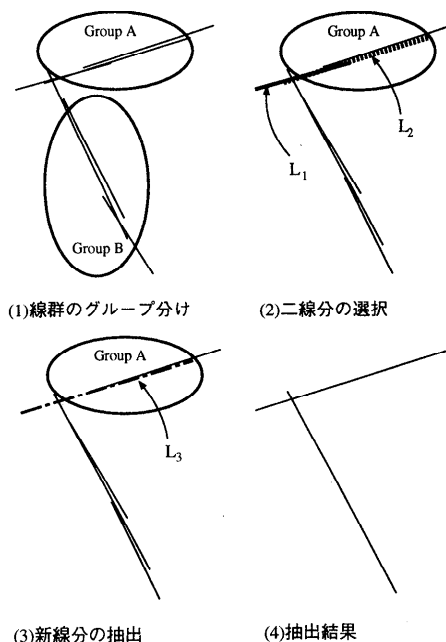


図5 直線抽出手順

Fig. 5 The thinning algorithm.

おりである。

- (i) 前述の定義に従い、角度・距離により線群をグループ分けする(図5(1))。
- (ii) グループ内で描かれた順に2本の線分 L_1 , L_2 を取り出す(図5(2))。
- (iii) 直線は1点と傾きにより一意に決まるので、 L_1 , L_2 の角度の平均および後に描かれた線分 L_2 の中点を用いる。得られた直線に対し2線分の端点の存在範囲でクリッピングを行い線分 L_3 を生成する(図5(3))。
- (iv) グループ内に未処理の線分がある場合、(iii) で生成された線分を L_1 、次に取り出した線分を L_2 とし、(iii) を実行する。
- (v) すべてのグループに対して (ii)~(iv) を行う。

描き手は、描き足していくことによりイメージに近付けることから、本アルゴリズムでは、後に描かれる

線分の比重を高くするために描いた順に線分抽出を行う。

4. 曲線の抽出

曲線のスケッチ描画においても直線と同様に1本で描画されることは少ない。また、描き始める位置や方向が一定でなく、描き足すなどの様々な描画が考えられ、単に入力順に制御点を結合し自由曲線を生成しても意図どおりに生成されない場合が多い。そこで本章では入力線群に対し逐次に制御点を再構成する手法を提案する。

曲線抽出においては、線群から節点抽出法を用いて制御点候補となる点を求めた後、(1) それらを接続する手法である特徴点結合法、(2) 凹凸を生じさせる制御点候補を削除し、滑らかに接続するための制御点を残す手法である節点分割抽出法から成る。以下ではそれぞれの手法について述べる。

4.1 特徴点結合法

本論文では入力線群から得られた制御点候補となる点を特徴点と呼ぶ。本手法は、入力された線群から特徴点を線分単位で抽出し、特徴点の結合により入力された線群を清書するための曲線の制御点を求める。

4.1.1 特徴点の抽出

重ね書きにより入力される線分は一般に複雑な形状を持たないため、1つの線分から特徴点となる点を端点を除き1点を選ぶ。そのために、再帰的な探索は行わない節点抽出法の特殊型を利用する。以下にそのアルゴリズムを示す。なお、図6(1)の曲線上の点はタブレットによる入力を一定時間間隔でサンプリングした点である。

- (i) 端点間を結んだ直線から最も遠い点を選ぶ(図6(1))。
- (ii) 端点間の距離 L と端点間を結ぶ直線と点の距離 h が ε よりも大きければ特徴点とする(図6(2)-1)。
- (iii) 端点以外の点がなくなった場合には、中点を特徴点とする(図6(2)-2)。

上記の端点間選ばれた特徴点および端点を特徴点列として扱う。

4.1.2 特徴点列の結合

線群に含まれるそれぞれの線分から得られた特徴点列を結合することにより新たな特徴点列を生成する。その規則は以下のとおりである。

- (i) 2つの特徴点列に重なりが生じている場合には、新たに描かれた線分の特徴点を優先して挿入する。特徴点列の重なりの有無は図7のように

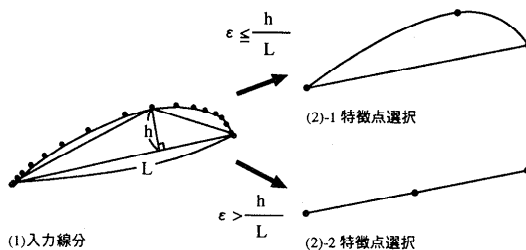


図6 特徴点選択
Fig. 6 Choice a characteristic point.

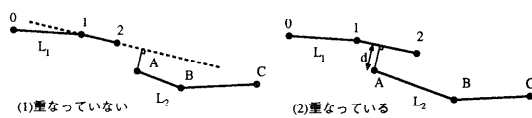


図7 点列の重なり判定
Fig. 7 Judgement of overlapping segments.

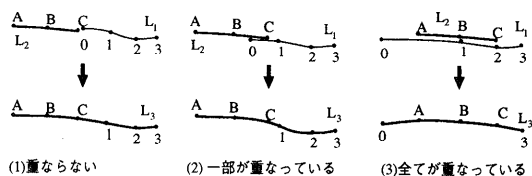


図8 特徴点の結合
Fig. 8 Connecting characteristic points.

定義する。追加点列 L_2 の各点から追加元点列 L_1 への垂線を下ろしたとき、垂線と L_1 の交点が存在し、その距離 d が閾値以下であるならば重なりが生じていると判断する。

- (ii) 特徴点どうしの距離が閾値より小さい場合、新たに描かれた線分の制御点に置き換える。

上記の規則を用いての3つの結合例を以下に示す。ここで、図8における L_1 (特徴点 0, 1, 2, 3) は結合済みの特徴点列、 L_2 (特徴点 A, B, C) は新たに入力された特徴点列、 L_3 は新たに生成された特徴点列を表している。

- (1) L_1, L_2 に重なりがない場合はそのままつなぐ(図8(1))。0, Cの距離が閾値より小さかったので0が削除された。この結果、 L_3 (特徴点 A, B, C, 1, 2, 3) が生成された。
- (2) L_1, L_2 の一部が重なる場合は、重複部分のみに後に描かれた線分 L_2 の特徴点列に置き換える(図8(2))。ここでは0が削除される。この結果、 L_3 (特徴点 A, B, C, 1, 2, 3) が生成された。
- (3) L_2 の特徴点が L_1 の間ですべて重なっている場合は、重複部分を後に描かれた線分の特徴点

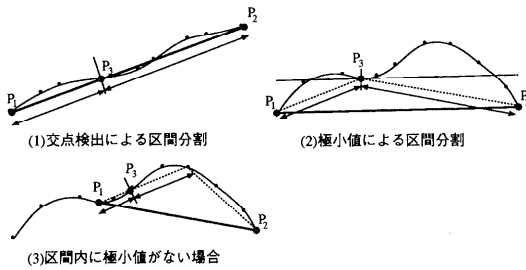


図9 節点分割抽出法による制御点選択

Fig. 9 Choice points by using the recursive amplitude through algorithm.

列に置き換える(図8(3)). ここでは1, 2が削除される. この結果, L_3 (特徴点0, A, B, C, 3)が生成される.

4.2 節点分割抽出法

本節では, 節点抽出法を曲線の制御点選択に適用するために拡張した, 節点分割抽出法について述べる. 本手法は, 振幅を残すべき区間を検出して全体を小さく分割し, 点列を複数の区間に分割する. そして, その分割区間で節点抽出法を利用することにより, 局所的な振幅に対する処理を行う. 以下に節点分割抽出法のアルゴリズムを示す.

- (i) 端点 P_1, P_2 を結ぶ直線と曲線との交点を検出する. 交点 P_3 を検出した場合, その点を分割点とし, その左右の区間で再帰的に同様の探索を行う(図9(1)).
- (ii) 区間 P_1, P_2 内で交点が見つからず, その区間に極小値が存在するならば, 最も小さな極小値を求める. その点を分割点 P_3 とし, (i)に戻る(図9(2)).
- (iii) 区間 P_1, P_2 内に交点が見つからず, 極小値が存在しない場合には, 区間端点から距離の最も遠い点から両端点に伸ばした直線との交点があるか調べ, 交点が存在するときはその点を分割点とする(図9(3)).
- (iv) 以上の区間分割が終了したのち, それぞれの区間で節点抽出法を用いて制御点削除を行う.

5. 実験結果

5.1 逐次清書法による直線描画

図10に逐次清書法を利用した図形の入力例を示す. 図10(1)は線群による入力例を示す. 図10(2)は, 描画を休む間に自動的に線分混合法により抽出された直線, 図10(3), (4)は, 抽出直線に対し, さらにスケッチを続けた結果を示す.

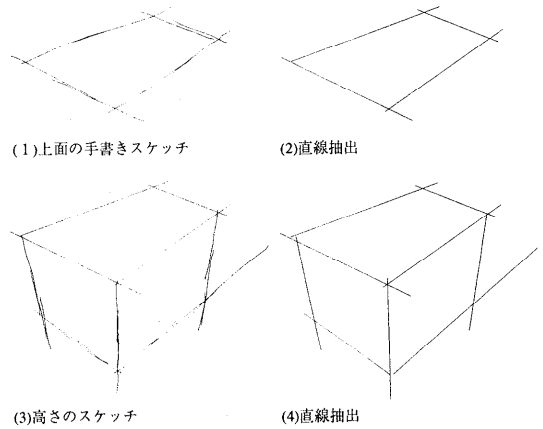


図10 線分混合法による直線描画

Fig. 10 Drawing parallelepiped with freehand lines.

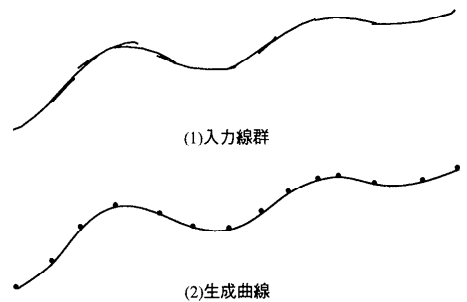


図11 重ね書きによる曲線描画

Fig. 11 Drawing a freehand curve with overlapped lines.

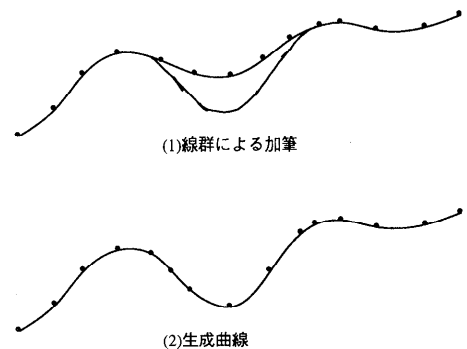


図12 清書曲線に対する加筆・修正

Fig. 12 Retouching a freehand curve.

5.2 特徴点結合法による曲線生成

図11は特徴点結合法を用いた重ね書きによる曲線の描画例である. 図11に, 8本の線分が入力され, 14の制御点により曲線が生成された例を示す. 図から分かるように, 一度に描かれた複数の線分から成るラフな曲線に対して, 本提案手法を用いることにより, 1

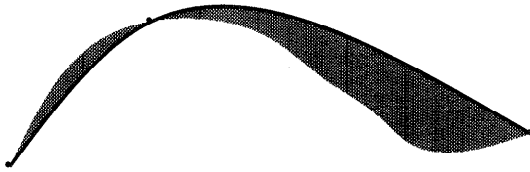


図13 差分面積の算出
Fig. 13 Difference.

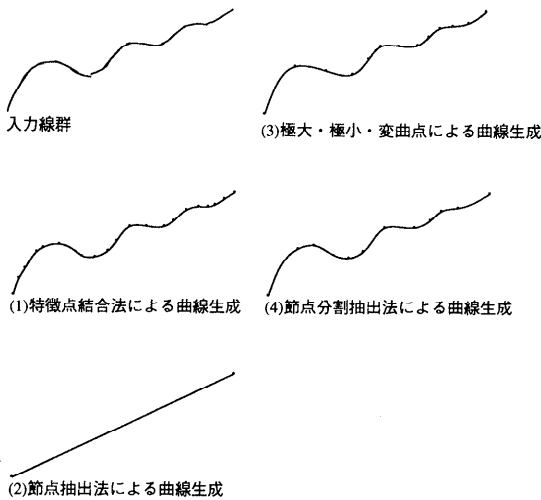


図14 曲線生成例1
Fig. 14 Generation of curve 1.

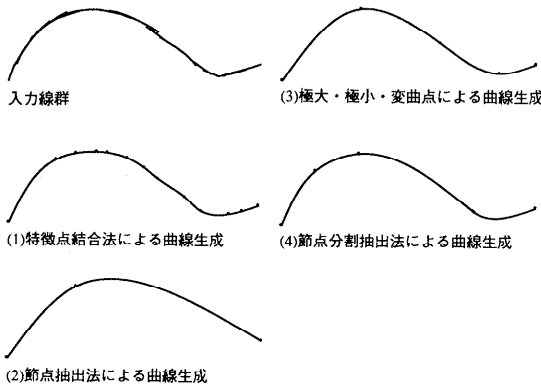


図15 曲線生成例2
Fig. 15 Generation of curve 2.

本の曲線を得ることができる。このように本手法を用いれば、入力する曲線を一筆書きで描く必要はない。

また、図12に、特徴点結合法により得られた曲線に対し、ペンを用いた直接修正を行った例を示す。ここでは5本の線分を入力した結果、6つの制御点が生成され、制御点列に挿入された。なお、制御点を結ぶ曲線の生成には自然スプライン曲線を用いている。

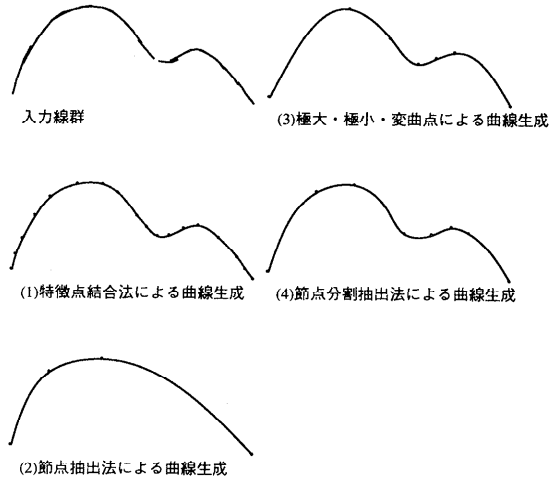


図16 曲線生成例3
Fig. 16 Generation of a curve 3.

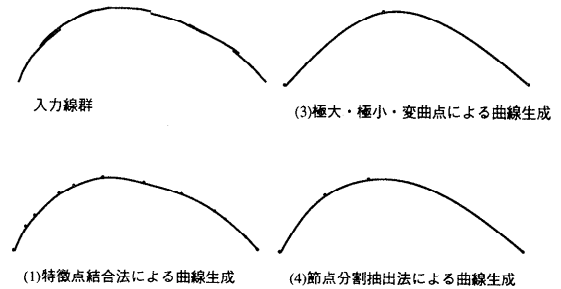


図17 曲線生成例4
Fig. 17 Generation of a curve 4.

5.3 生成制御点の比較

本節では入力線群に対する曲線抽出を行った結果を図14~17に示す。比較のために用いた手法は、(1)特徴点結合法、(2)節点抽出法、(3)極大値・極小値・変曲点に制御点を取る手法、(4)本論文で提案した節点分割抽出法である。なお、節点抽出法および節点分割抽出法で用いた閾値は $\varepsilon = 0.13$ である。

表1~4は、それぞれの生成曲線についての評価である。表に、その曲線の持つ制御点の数、特徴点結合法からの制御点削減率、差分面積を算出した結果を示す。この表における差分とは、特徴点結合法による曲線との間にできる面積を表している。特徴点結合法により生成された曲線は、入力曲線に最も近い形状と考えられるので、その差を再現性の基準とした。図13は図15においての特徴点結合法と節点抽出法による

表 1 図 14 における曲線の評価
Table 1 Evaluation of fig. 14.

	制御点数 (個)	制御点削減率	差分面積 (Dot)
特徴点結合法	21	/	/
節点抽出法	2	90%	15526
極大・極小・変曲点	13	38%	1011
提案手法	12	43%	436

表 2 図 15 における曲線の評価
Table 2 Evaluation of fig. 15.

	制御点数 (個)	制御点削減率	差分面積 (Dot)
特徴点結合法	13	/	/
節点抽出法	3	73%	18584
極大・極小・変曲点	5	62%	3126
提案手法	6	54%	556

表 3 図 16 における曲線の評価
Table 3 Evaluation of fig. 16.

	制御点数 (個)	制御点削減率	差分面積 (Dot)
特徴点結合法	18	/	/
節点抽出法	4	78%	9047
極大・極小・変曲点	7	61%	2970
提案手法	9	50%	751

表 4 図 17 における曲線の評価
Table 4 Evaluation of fig. 17.

	制御点数 (個)	制御点削減率	差分面積 (Dot)
特徴点結合法	12	/	/
節点抽出法	4	67%	1324
極大・極小・変曲点	3	75%	4161
提案手法	4	67%	1324

曲線の差分を求めた図である。ここで、図のグレーの部分が差分面積となる。単位は画素数で表す。

(i) 曲線生成例 1 (図 14)

制御点削減率だけを見れば節点抽出法が最も優秀であるが、原形をまったく残さない形状になっており、差分面積も最大となっている。制御点削減において極大・極小・変曲点法に対し、本手法はわずかな優位であるが、差分面積が 2 倍以上と再現性においても本手法が優れている。また、形状を比較しても、本手法が一番左の振幅の形状をよく残している。

(ii) 曲線生成例 2 (図 15)

制御点削減率において (i) と同様に節点抽出法が数値が高いが、再現性が低い。差分面積において提案手法は最も優れており、形状においても入力線群の形状を最もよく再現していることは図 15 から明らかである。

(iii) 曲線生成例 3 (図 16)

制御点削減率において (i) と同様に節点抽出法が数値が高いが、再現性が低い。差分面積において提案手法は最も優れており、入力曲線の外形を維持した制御点選択が可能となっている。

(iv) 曲線生成例 4 (図 17)

節点抽出法と提案手法では、アルゴリズム上抽出結果はまったく同じになる。制御点削減率において、極大・極小・変曲点法がわずかに優位であるが、差分面積において節点抽出法、提案手法が優れており、入力に沿った制御点選択が行われているといえる。

以上の結果から、本論文で提案する節点分割抽出法は、(1) 特徴点結合法で得られた制御点列に対し、制御点数を半減させることが可能、(2) 他の手法と比較し、入力線群の概形を残した制御点削除が可能であることが分かる。

6. おわりに

本論文では、スケッチによる線分の重ね書きを可能にする逐次清書法の提案を行った。ペンを用いた直接入力を可能にする本手法は、スケッチの手法をコンピュータ上で実現するために有効な手法であり、試行錯誤を可能にする入力インタフェースとして提案するものである。

提案手法は、直線・曲線の逐次清書を可能にするための(1)線分混合法、(2)特徴点結合法、(3)節点分割抽出法である。線分混合法は重ね書きによる直線生成を可能にする手法であり、特徴点結合法、節点分割抽出法は、手書きによる曲線描画に対し、清書を行うための手法である。

本手法の提案により、以下の有効な点が得られた。

- (1) 従来手法では不可能であった重ね書きによる入力が可能となる。
- (2) 対話的に試行錯誤を行いながら線分抽出を行うことができる。
- (3) 抽出線分に対しても同様のアルゴリズムで修正が可能である。
- (4) 少ない制御点で入力線群を曲線で表現することが可能である。

以上のことからスケッチによる重ね書き線分入力を可能にするインタフェースとして有効な手法であることを示した。

参 考 文 献

- 1) マルチメディアコンテンツ振興協会：高付加価値意匠デザインのための3次元形状モデリングに関する調査研究報告書，pp.5-9 (1998)。
- 2) 日本機械学会：形態とデザイン，pp.7-12，培風館 (1993)。

- 3) Chen, C.L.P. and Xie, S.: Freehand drawing system using a fuzzy logic concept, *Computer Aided Design*, Vol.28, No.3, pp.77-89 (1996)。
- 4) Ramer, U.: An Iterative Procedure for the Polygonal Approximation of Plane Curves, *Computer Graphics and Image Processing*, Vol.1, pp.244-256 (1972)。
- 5) 近藤邦雄，木村文彦，田嶋太郎：曲面の形状感の表現（第3報）手描き入力による図形作成法，精密工学会誌，Vol.53, No.4 (1987)。
- 6) 五十嵐健夫ほか：対話的整形と予測描画による幾何学図形的高速描画，インタラクション'98 論文集，pp.25-26 (1998)。

(平成 10 年 6 月 1 日受付)

(平成 10 年 12 月 7 日採録)

松田 浩一 (学生会員)



の研究に従事。

1971 年生。1997 年埼玉大学大学院理工学研究科情報工学専攻博士前期課程修了。現在，同大学院情報数理科学専攻博士後期課程在学中。ペンをを用いた形状入力インタフェース

近藤 邦雄 (正会員)



教授。コンピュータグラフィクス，ユーザインタフェース，形状モデリング，感性と知識をもとにした画像処理の研究に従事。日本図学会会員。

1954 年生。1979 年名古屋工業大学第2部機械工学科卒業。工学博士。名古屋大学教養部図学教室，1988 年東京工芸大学電子工学科講師を経て1989 年埼玉大学工学部情報工学科助