

逐次型ファジィスプライン曲線生成法

佐藤 洋一

佐賀 聡人

室蘭工業大学 情報工学科

抄録

我々は、一筆の手書き描画動作をあいまいな曲線のモデルである「ファジィスプライン曲線」として表現し、さらにこれを基に手書き曲線を同定する手法 FSCI を既に提案している。しかしこの手法では、修正インタフェースを必要とし描画動作のみの一貫したペン操作による図形入力インタフェースは実現できていない。そこで、本稿では、複数回の描画ストローク、いわゆる重ね書きによってファジィスプライン曲線を逐次修正しつつ生成する手法「逐次型ファジィスプライン曲線生成法 (S-FSCG)」を新たに提案する。また、この S-FSCG と FSCI を組み合わせた簡易重ね書き図形入力インタフェースを試作し、従来のような修正インタフェースを必要としない重ね書きによる手書き図形入力インタフェースの実現の可能性を示す。

Sequential Fuzzy Spline Curve Generator

Yoichi SATO

Sato SAGA

Department of Computer Science and Systems Engineering,
Muroran Institute of Technology

Abstract

We have proposed a method to express a stroke of freehand drawing as a vague curve model named fuzzy spline curve, and then proposed a freehand curve identification technique, FSCI, on the basis of the fuzzy spline curve. However, because this technique requires additional correcting interface, it cannot realize complete freehand interface for graphics input that is purely controlled by drawing manner. This paper newly proposes a sequential fuzzy spline curve generator, S-FSCG, which generates a fuzzy spline curve with modifying the previous fuzzy spline curve in every event of overlapping freehand drawing. This paper also realizes a simple prototype of graphics input interface using proposed S-FSCG and FSCI, which requires no additional correcting interface.

1 はじめに

手書き描画動作をあいまいな曲線のモデルである「ファジィスプライン曲線」として表現する手法が文献 [1] で提案され、またこれを基にして書き手の意図する幾何図形を推論する手法「ファジィスプライン曲線同定法 (Fuzzy Spline Curve Identifier, FSCI) が文献 [2] で提案されている。さらに文献 [3] では FSCI による描画動作認識にペン操作による修正インタフェースを付加することにより手書き図形入力インタフェースを実現した。

しかしこの修正インタフェースではピッキングやドラッグといった描画動作以外のペン操作を必要とし、全体として描画動作のみによる手書き図形入力インタフェースとはなっていない。これは文献 [1] のファジィスプライン曲線の生成法 (以下、これを Fuzzy Spline Curve Generator, FSCG と呼ぶ。) では 1 回の描画ストロークのみでファジィスプライン曲線が決定されてしまうことに起因する。

そこで本稿では、複数回の描画ストロークの重ね書きによってファジィスプライン曲線を逐次修正しつつ生成する手法「逐次型ファジィスプライン曲線生成法 (Sequential Fuzzy Spline Curve Generator, S-FSCG)」を新たに提案する。これにより従来のような付加的な修正インタフェースを必要しない、手書き描画動作のみによる図形入力インタフェースの実現が期待できる。

2 従来の FSCG と FSCI による図形入力インタフェースの概要

文献 [3] で提案された従来の図形入力インタフェースの概要 (図 1 参照) について述べる。

2.1 従来の FSCG の概要 [1]

従来の FSCG は一回の描画ストロークからその形状と描画の動特性をもとにして一本のファジィスプライン曲線を生成する。

まずタブレット上に描画された手書き曲線は離散的にサンプリングされ、サンプル点列 $a_k (k = 0, 1, \dots, n_a - 1)$ とそれらのタイムスタンプ列 $t_k (k = 0, 1, \dots, n_a - 1)$ として得られる。

FSCG は最初に各サンプル点 a_k に対して位置情報のあいまいさの程度を表わす量 r_{a_k} をファジィネスとして付加することでファジィ点列 $\tilde{a}_k (k = 0, 1, \dots, n_a - 1)$

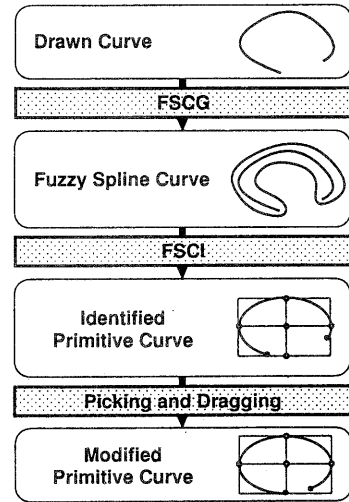


図 1: 従来の FSCG と FSCI による図形入力インタフェースの処理の流れ

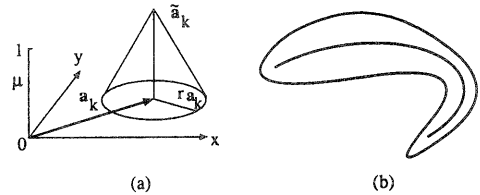


図 2: (a) ファジィ点 (b) ファジィスプライン曲線

を構成する。ただし、ファジィ点 \tilde{a}_k は図 2(a) に示す円錐型メンバシップ関数

$$\mu_{\tilde{a}_k}(v) = \left(1 - \frac{\|v - a_k\|}{r_{a_k}}\right) \vee 0 \quad (1)$$

によって特徴づけられるファジィ位置ベクトルである。ここで、ファジィネス r_{a_k} は、描画の動特性をもとに文献 [4] に従って、描画の雑な部分では大きく、逆に描画のていねいな部分で小さくなるように設定される。

FSCG は次に、ファジィ点列 $\tilde{a}_k (k = 0, 1, \dots, n_a - 1)$ をファジィスプライン補間することで描画曲線を、次数 n 、区間数 L のファジィスプライン曲線

$$\tilde{s}(u) = \sum_{i=0}^{L+n-1} \tilde{d}_i N_i^n(u) \quad (2)$$

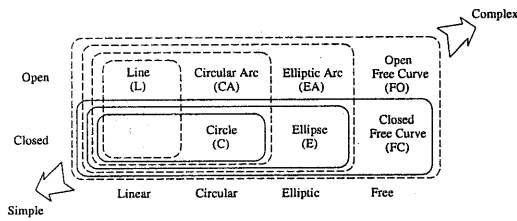


図 3: 曲線プリミティブ間の包含関係

として表現する. ただし \tilde{d}_i はファジィ制御多角形, $N_i^n(u)$ は n 次の正規化 B-スプラインである.

このファジィスプライン曲線は図 2(b) のように, あいまいさを変化させながら移動するファジィ点モデルの移動軌跡とみなすことができ, 雑に書かれたところではあいまいさが大きく, 逆に丁寧に書かれたところではあいまいさが小さくなる.

2.2 FSCI の概要 [2]

FSCI は, FSCG で生成されたファジィスプライン曲線を基に書き手の意図する曲線クラスを推定し, 描画ストロークを一般的な CAD 図形の構成要素として必要不可欠な 7 種類の幾何曲線プリミティブ, 「線分 (L)」, 「円 (C)」, 「円弧 (CA)」, 「楕円 (E)」, 「楕円弧 (EA)」, 「閉自由曲線 (FC)」, 「開自由曲線 (FO)」のいずれかとして同定する. しかしここで, これら曲線プリミティブ間には図 3 のような包含関係が存在するため, 描画形状のみからこれらを分類しようとすると, 必然的により表現能力の高い曲線プリミティブが選択されてしまうことになる. この為 FSCI の推論は, あいまいさの広がりを持ったファジィスプライン曲線の中からなるべく単純な曲線プリミティブを見い出そうとするように設計されており, ユーザーはこの性質を利用して, 雑な描画によって単純な曲線プリミティブを認識させたり, 逆に丁寧な描画によって複雑な曲線プリミティブを認識させる事ができる. したがって, FSCI では描画形状とともにファジィスプライン曲線のファジネスの大きさが同定結果に直接的な影響を与える.

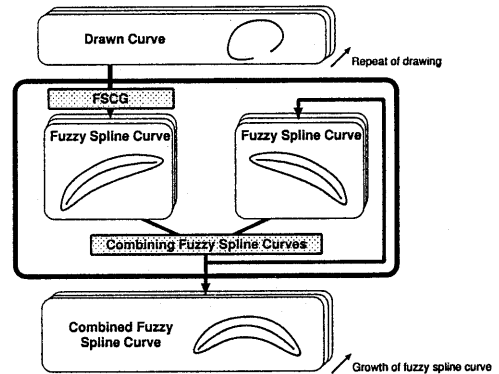


図 4: S-FSCG の処理の流れ

2.2.1 修正インタフェース

FSCI の同定結果が意図したものでない場合, 候補補選択をペンによるピッキングにより行なう. また位置や形状の微調整をペンによるドラッキングにより行なう.

3 逐次型ファジィスプライン曲線生成法 S-FSCG の提案

複数の描画ストロークによって生成される複数のファジィスプライン曲線から 1 本のファジィスプライン曲線を生成する逐次型ファジィスプライン曲線生成法 (S-FSCG) を提案する. 実際には多くのファジィスプライン曲線を一度に融合し 1 本を生成するのではなく, 図 4 のように逐次的に 1 本ずつ融合していくことにより, 最終的に複数本から 1 本を得る. 従って, 以下では 2 本のファジィスプライン曲線を融合し, 1 本のファジィスプライン曲線を生成する手順を示す.

3.1 ファジィスプライン曲線のファジィ点時系列化

1 本目のファジィスプライン曲線 $\tilde{a}(s)$ と, 2 本目のファジィスプライン曲線 $\tilde{b}(t)$ が与えられた時, これらをそれぞれ等距離間隔で評価しファジィ点時系列

$$\{(\tilde{a}_i, s_i) | i = 0, \dots, n_a - 1\}$$

$$\{(\tilde{b}_j, t_j) | j = 0, \dots, n_b - 1\}$$

を求める.

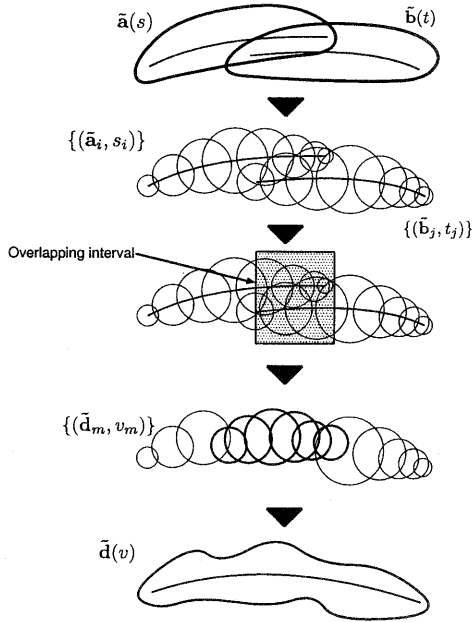


図 5: 2本ファジィスプライン曲線の融合手順

3.2 重複区間の検出

$\bar{a}(s)$ と $\bar{b}(t)$ の間の融合するべき重なり部分を重複区間として検出する。ただし、融合した結果が一本のファジィスプライン曲線となるように重複区間の数は、ただかひとつとする。また、分岐や交差となるような融合すべきでない重なり部分は、重複区間とはしない。

3.2.1 重複状態行列の生成

$\bar{a}_i (i = 0, \dots, n_a - 1)$ と $\bar{b}_j (j = 0, \dots, n_b - 1)$ の全ての組合せについて、2つのファジィ点が重複している度合を可能性値

$$U(i, j) = \sup_v (\mu_{\bar{a}_i}(v) \wedge \mu_{\bar{b}_j}(v)) \quad (3)$$

として評価し、重複状態行列 (OSM)

$$U = \begin{pmatrix} U(0,0) & U(0,1) & \dots & U(0,n_b-1) \\ U(1,0) & U(1,1) & \dots & U(1,n_b-1) \\ \vdots & \vdots & \ddots & \vdots \\ U(n_a-1,0) & U(n_a-1,1) & \dots & U(n_a-1,n_b-1) \end{pmatrix} \quad (4)$$

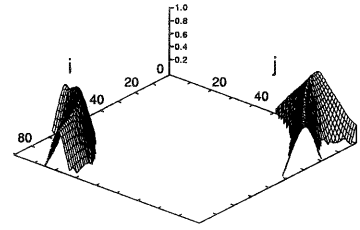


図 6: OSM U のプロット例

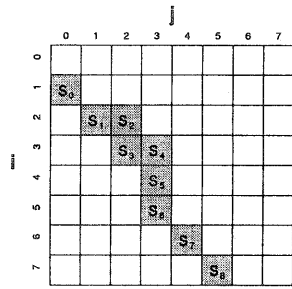


図 7: OSM U 上の R の例

を生成する。これをプロットすると例えば図 6 のようになる。

3.2.2 重複ファジィ点ペア列群の抽出

OSM U をもとにファジィ曲線 $\bar{a}(s)$ と $\bar{b}(t)$ の間の重複区間を抽出する。この重複区間は具体的には重複ファジィ点ペア列

$$R = (S_0, S_1, \dots, S_{n_R-1}) \quad (5)$$

として抽出する。ただし、 $S_k = (i_{S_k}, j_{S_k}) (i_{S_k} \in \{0, \dots, n_a - 1\}, j_{S_k} \in \{0, \dots, n_b - 1\})$ は、 $\bar{a}_{i_{S_k}}$ と $\bar{b}_{j_{S_k}}$ のペアを表し、R は行列 U 上の任意の辺から任意の他の辺へ単調に至る 1 本の経路を成すものとする。以下では一般性を失う事なしに図 7 の例のような $j = 0$ の辺から $i = n_a - 1$ の辺に至る場合について議論することにする。

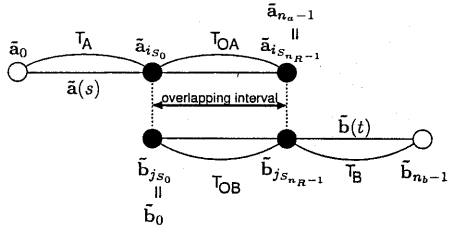


図 8: 各点の時間的位置関係の模式図

3.2.3 重複ファジィ点ペア列の評価と決定

$\tilde{a}(s)$ と $\tilde{b}(t)$ 間の重複区間は一般に複数存在するため、上述の R も複数存在することになる。ここでは以下の評価関数 $O(R)$ によって個々の R を評価し、最良のものを求めることにより重複区間をただ一つ選出する。すなわち、まず重複区間 R 全体にわたっての重複可能性値を評価する関数として、

$$P(R) = \min(U(S_0), \dots, U(S_{n_R-1})) \quad (6)$$

を定義する。次に 2 曲線全体に対しての重複区間の占める時間的な割合の大きさを評価する関数として、

$$T(R) = T_O/T_T \quad (7)$$

を定義する。ただし、 $T_{OA} = (s_{i_{s_{n_R-1}}} - s_{i_{s_0}})$ 、 $T_{OB} = (t_{j_{s_{n_R-1}}} - t_{j_{s_0}})$ 、 $T_O = (T_{OA} + T_{OB})/2$ 、 $T_A = (s_{i_{s_0}} - s_0)$ 、 $T_B = (t_{n_b-1} - t_{j_{s_{n_R-1}}})$ 、 $T_T = T_A + T_O + T_B$ である (図 8 参照)。このとき、 $P(R)$ と $T(R)$ の合成

$$O(R) = \alpha P(R) + \beta T(R) \quad (8)$$

によって R の重複度を評価し、これを最大にする R をただ一つ選出しこれを重複区間とする。これは図 6 からなるべく高く長い峰を R として抽出することを意味する。

3.3 重複区間の融合

抽出された重複区間を平均操作によって融合する。

3.3.1 時刻情報の調整

融合に先だって $\tilde{a}(s)$ と $\tilde{b}(t)$ の時刻情報の調整を行なう。すなわちまず、 \tilde{b}_0 の時刻を $\tilde{a}_{i_{s_0}}$ の時刻に一致さ

せるために、 $t_j \leftarrow t_j + T_A$ ($j = 0, \dots, n_b - 1$) と $\tilde{b}(t)$ の時刻情報をシフトする。

次に、重複区間での経過時間を一致させるために、

$$s_i \leftarrow s_{i_{s_0}} + (s_i - s_{i_{s_0}}) \frac{T_O}{T_{OA}} \quad (9)$$

$(i = i_{s_0}, \dots, i_{s_{n_R-1}})$

$$t_i \leftarrow t_{j_{s_0}} + (t_j - t_{j_{s_0}}) \frac{T_O}{T_{OB}} \quad (10)$$

$(j = j_{s_0}, \dots, j_{s_{n_R-1}})$

なる操作により、 $\tilde{a}(s)$ 、 $\tilde{b}(t)$ それぞれの重複区間の時間の伸縮を行なう。

3.3.2 平均操作による重複ファジィ点ペア列の融合

重複区間から平均操作によりファジィ点時系列 $\{(\tilde{c}_l, u_l) | l = 0, \dots, n_R - 1\}$ を式 (11) により構成する。

$$(\tilde{c}_l, u_l) = \left(\frac{\tilde{a}_{i_{s_l}} + \tilde{b}_{j_{s_l}}}{2}, \frac{s_{i_{s_l}} + t_{j_{s_l}}}{2} \right) \quad (l = 0, \dots, n_R - 1) \quad (11)$$

3.4 ファジィスプライン曲線の再構成

ファジィ点時系列 $\{(\tilde{d}_m, v_m)\}$ を

$$(\tilde{d}_m, v_m) = \begin{cases} (\tilde{a}_m, s_m) & (m = 0, \dots, i_{s_0-1}) \\ (\tilde{c}_{m-i_{s_0}}, u_{m-i_{s_0}}) & (m = i_{s_0-1}, \dots, i_{s_0} + (n_R - 1)) \\ (\tilde{b}_{m-(i_{s_0}+n_R)}, t_{m-(i_{s_0}+n_R)}) & (m = i_{s_0} + n_R, \dots, (i_{s_0} + n_R) + (n_b - 1 - j_{s_{n_R-1}})) \end{cases}$$

と構成し、これに新ためてファジィスプライン補間を施すことにより $\tilde{a}(s)$ と $\tilde{b}(t)$ を融合したファジィスプライン曲線 $\tilde{d}(v)$ を生成する。

3.5 S-FSCG の動作実験例

これまでに説明した S-FSCG の実際の動作実験例を以下に示す。図 9 と図 10 は、2 曲線の融合が可能な場合である。図 10 では、複数の重複区間の候補が認められるが重複度の比較によって、ただ一つの重複区間が選出されている。また図 11 と図 12 は、分岐及び交差で融合すべきでない例であり、この時は OSM 上に辺から辺に至る峰は求められず、重複区間が検出されないという事がわかる。この場合は融合処理は行なわず、2 本目の曲線は破棄する。

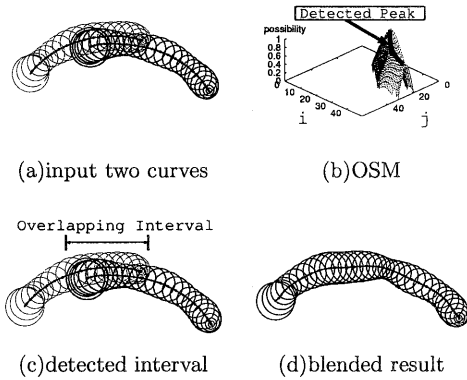


図 9: 1 本目の途中から 2 本目書かれた場合

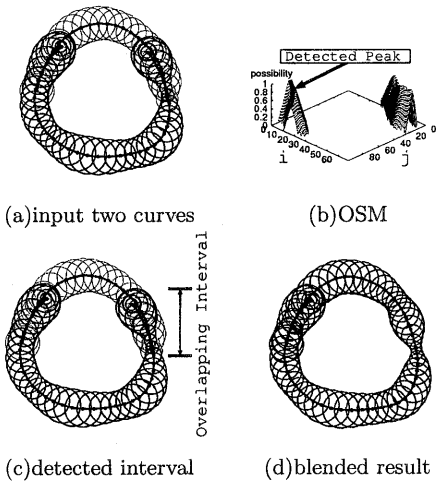


図 10: 複数の融合区間の候補がある場合

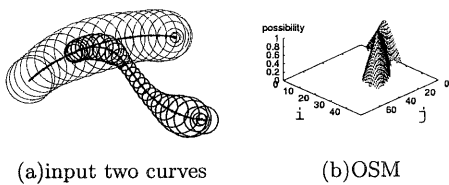


図 11: 融合しない場合 1 (分岐)

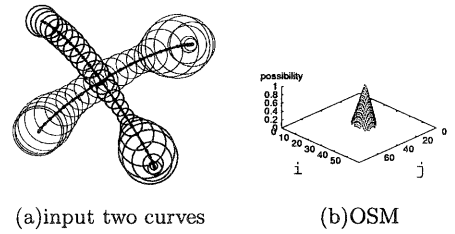


図 12: 融合しない場合 2 (交差)

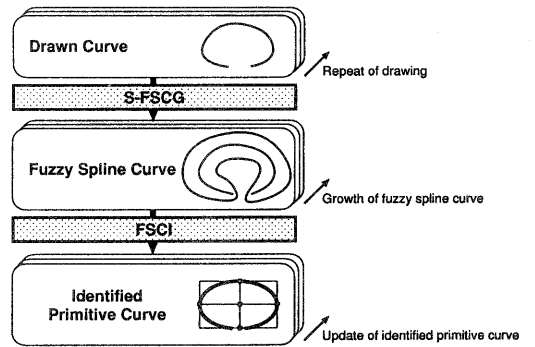


図 13: S-FSCG を用いた簡易重ね書きインタフェースの処理の流れ

4 S-FSCG と FSCI による重ね書き図形入力インタフェースの構成例

「S-FSCG」と「FSCI」を組み合わせることにより簡易的な重ね書き図形入力インタフェースを試作した。

4.1 簡易重ね書き図形入力インタフェースの構成

図 13 に示す簡易的重ね書き図形入力インタフェースを試作した。本インタフェースでは描画動作が行なわれる都度に、ファジィスプライン曲線が修正され、これに伴ない同定結果が更新される。従ってユーザーは、同定結果を見ながら重ね書き描画を繰り返し同定結果に修正を加える事ができる為、従来のような修正インタフェースの付加は必要なくなる。

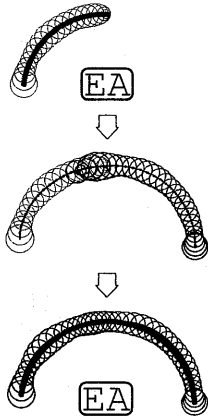


図 14: 長さに対する修正例

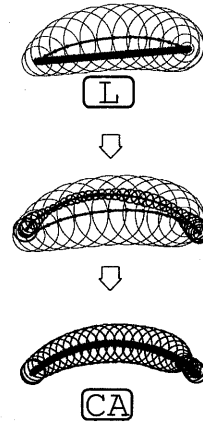


図 15: 曲線クラスの修正例 1 (線分から円弧へ)

4.2 重ね書き修正の例

図 14 から図 18 までに簡易インタフェースによる代表的な重ね書き修正の例を 4 種類示す。図はそれぞれ 3 段に分かれており、上段が 1 本目の曲線、中段が 1 本目の曲線及び重ね書きされた 2 本目の曲線、下段が中段の 2 曲線を融合した結果である。尚、上段と下段については、同定された曲線を太い実線とその種類を示すラベルを付与した。

4.2.1 同定結果の長さに対する修正例

同定結果の楕円弧が意図した長さで得られなかった場合の修正例を図 14 に示す。重ね書きによって楕円弧長を伸ばしている。

4.2.2 同定結果の曲線クラスに対する修正例

意図した種類の曲線プリミティブが同定されなかった場合の修正例を図 15 と図 16 に示す。図 15 の例では、初め雑に書きすぎて線分と同定されたものを、より丁寧な描画で円弧形状を重ね書きをすることで円弧と同定されるように修正している。図 16 の例では、逆に円弧と同定されたものを、線分と同定されるように修正している例である。

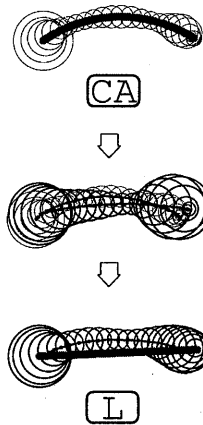


図 16: 曲線クラスの修正例 2 (円弧から線分へ)

4.2.3 分割点に対する修正例

図 17 は、分割点に対する修正の例である。本稿では触れなかったが、FSCI には描画動作の停止性に基づいて、曲線を複数のセグメントに分割する機能がある [1]。図の左側の例では、意図しない分割が行なわれた後に、停止動作を含まない重ね書きをすることで、1 本のセグメントへと接続を行なっている。また同図の右側の例では、意図と反して分割が行なわれなかった場

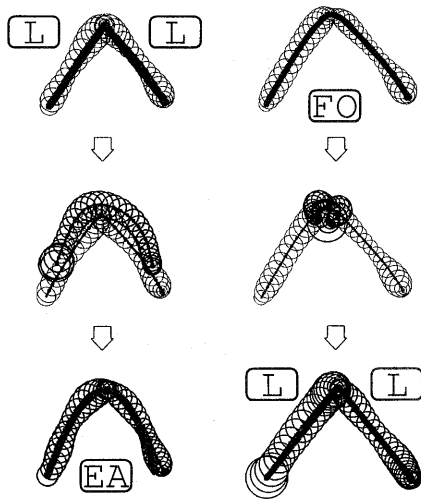


図 17: 分割点に対する修正例 (左)2本から1本へ (右)1本から2本へ

合, 分割を行ないたい箇所に停止動作を重ね書きすることで, 2本のセグメントへと分割を行なっている。

4.2.4 位置・姿勢の微調整の例

図 18 に, 位置・姿勢の微調整の例を示す。図形が意図した位置・姿勢に同定されなかった場合など, 位置をずらして重ね書きをする事により, 位置や姿勢に対する微調整を行なっている。図の左側の例では線分について, 右側の例では円弧についてそれぞれ似通った形状を近傍に重ね書きする事により, 微妙な位置的な修正を, 描画動作により行なっている。

5 おわりに

本稿では複数回の描画ストロークの重ね書きによってファジィスプライン曲線を逐次修正しつつ生成する手法 S-FSCG を提案した。また S-FSCG と FSCI を組み合わせた簡易重ね書き図形入力インタフェースを試作し, 従来のような修正インタフェースを必要としない重ね書きによる手書き図形入力インタフェースの実現可能性を示した。

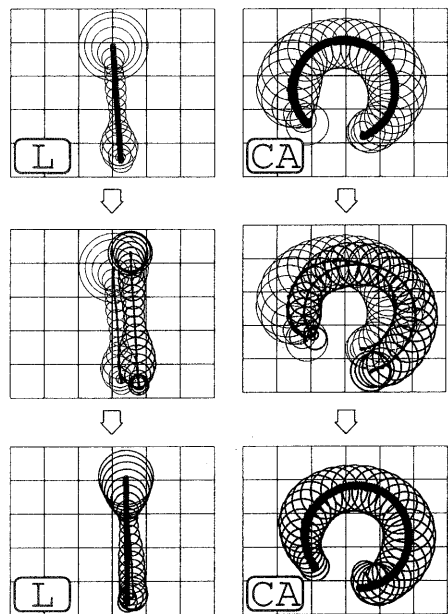


図 18: 微調整の例 (左)線分の場合 (右)円弧の場合

今後は, 文献 [3] のようなスナッピング処理の実装や複数セグメント入力への対応といった, さらに本格的な評価・改良を行ない実用的な手書き図形入力インタフェースの実現を図る予定である。

参考文献

- [1] 佐賀聡人, 牧野宏美, 佐々木淳一: 手書き曲線モデルの一構成法 — ファジィスプライン補間法 —, 電子情報通信学会論文誌, Vol.J77-D-2 No.8, pp.1610-1619, 1994.
- [2] 佐賀聡人, 牧野宏美, 佐々木淳一: ファジィスプライン曲線同定法, 電子情報通信学会論文誌, Vol.J77-D-2 No.8, pp.1620-1629, 1994.
- [3] 佐賀聡人, 佐々木淳一: ファジィスプライン曲線同定法を用いた手書き CAD 図形入力インタフェースの試作, 情報処理学会論文誌, Vol.36 No.2, pp338-350, 1995.
- [4] 大川哲也, 佐賀聡人,: 手書き曲線同定法 FSCI におけるファジィ生成モデルの精密化, 信学論 (D-I), vol.J82-D-I, no.5, pp.634-643, May.1999.
- [5] 松田浩一, 近藤邦雄: 手書き図形入力のための時系列情報を利用した逐次消画法, 情報処理学会論文誌, Vol.40 No.2, pp594-600, 1999.