

Dynamic Programming によるセル画アニメーション内 均等色領域輪郭線符号化の検討

宮澤 敏記¹⁾ 中神 央二¹⁾ 亀山 渉²⁾ 渡辺 裕²⁾ 富永 英義^{1),2)}

¹⁾ 早稲田大学 理工学部 電子・情報通信学科
〒169-8555 東京都新宿区大久保 3-4-1
miyazawa@tom.comm.waseda.ac.jp

²⁾ 早稲田大学大学院国際情報通信研究科
〒169-0051 東京都新宿区西早稲田 1-3-10 早大 29-7 号館

あらまし

セル画アニメーション内には多数の均等色領域が存在する。この領域の輪郭線を有意味点及び均等色を代表する色で近似することで圧縮率改善が図られる。そこで、本稿では輪郭線近似のための有意味点を dynamic programming を基に抽出し、輪郭を近似するアルゴリズムを提案する。Dynamic programming を用いることで、誤差関数を基に、近似関数に最適な有意味点を繰り返し抽出可能となる。Dynamic programming は繰り返し演算に伴って、計算時間が大きくなることが問題である。そこで、本アルゴリズムの実行時間評価を行い、計算時間に影響を与える要素について検討を行った。一方、本アルゴリズムは、1) 点数削減率を正確に制御できない、2) 閾値によって輪郭線内に局所的に抽出点が発散してしまう問題がある。これらの問題を解決するために Feedback algorithm を提案する。

キーワード 画像符号化, アニメーション, スプライン曲線, 均等色領域, dynamic programming
feedback algorithm

A Study on Contour Approximation methods for Homogeneous Color Region in Animation Image

Toshinori MIYAZAWA¹⁾ Ouji NAKAGAMI¹⁾ Wataru KAMEYAMA²⁾ Hiroshi WATANABE²⁾ Hideyoshi TOMINAGA^{1),2)}

¹⁾ Dept. of Electronics, Information and Communication Engineering, WASEDA University
3-4-1 Ohkubo, Shinjuku-ku, Tokyo, 169-8555 JAPAN
miyazawa@tom.comm.waseda.ac.jp

²⁾ Global Information and Telecommunication Study, WASEDA University
29-7 Bldg, 1-3-10 Nishi-Waseda, Shinjuku-ku, Tokyo, 169-0051 JAPAN

abstract:

There are many homogeneous color regions in animation images. By approximating the regions by significant points and representative colors, coding efficiency can be improved. In this paper, we propose an approximation method for the regions based on dynamic programming. An approximation criterion used in the method is an error value defined by the distance between an original and an approximated line. If the error value is smaller than a threshold, a significant point that provides the minimum error value is extracted from the line. This process is iteratively executed until the error value exceeds the threshold. Since the the number of iteration tends to be large, we evaluate the processing time of significant elements, which is dominant for computational complexity. The remaining problems, in the proposed approximation, are 1) lack of point reduction ratio control and 2) excess extraction of significant points at some threshold. We also propose feedback algorithm in order to solve the problems.

keywords Image Coding, Animation Image, spline curve, homogeneous color region
dynamic programming, feedback algorithm

1. はじめに

DCT ベースの符号化方式は高周波成分を多く含むセル画アニメーションコンテンツには適していない。そこで我々は、セル画アニメーションに最適な符号化方式の検討を行ってきた。セル画アニメーションにおける高周波成分として、1) 線画、2) 均等色領域の輪郭線の2種類が存在する。本稿では、均等色領域の輪郭線の符号化方式を検討する。第一に、輪郭線の形状を符号化するために、Dynamic Programming により補間処理に最適な有意点（標本化点）を抽出するアルゴリズムを提案する。Dynamic Programming の問題として計算時間が大きいことが挙げられる。そこで第二に計算時間の評価を行った。第三には、Dynamic programming に用いる閾値に依存し、データ量が発散する問題を解決するために、冗長な有意点を削除及びサンプリング制御を実現する Feedback algorithm を提案した。

2. セル画アニメーションにおける均等色領域

セル画アニメーションにおける均等色領域例を図1に示す。1フレーム内の各均等色領域は既に提案した抽出アルゴリズムにより、取得可能となっている [1][2]。本稿ではこの均等色領域の輪郭線（閉曲線）を誤差を最小とする近似曲線を得るためのアルゴリズムを検討する。

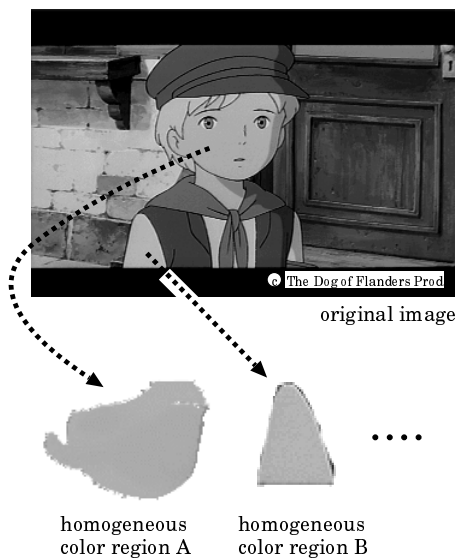


図 1: 均等色領域例

3. Dynamic Programming による輪郭線符号化

3.1 Dynamic Programming 定義

DP(Dynamic Programming)とは、取り得る全ての事象を評価関数に入力し、最適解を再帰的に導出する手法である。DPは輪郭線を近似するための最適な有意点抽出処理に適用することが可能である。DPの定義を輪

郭線近似に拡張すれば、次のように表現できる。輪郭線上の全ての点は輪郭線近似のための有意点と成り得る。この全ての点の中から1点を有意点と仮に決定し、新たな有意点を追加して得られた近似曲線を近似誤差関数に入力する。全ての対象点について仮の近似曲線の誤差値を算出し、その中で最小の誤差値を与える点を最終的に有意点と決定する。この処理はある条件（誤差値の閾値を満たす等）を満たすまで再帰的に繰り返す。

3.2 評価式定義

輪郭線近似において取り得る事象とは輪郭線上の全ての点であり、評価関数とは輪郭線近似の際の近似誤差算出式である。近似誤差算出式としては、従来研究 [3][4] より1) 面積誤差、2) 最小距離誤差の2つに分類することができる。前者は近似線と原曲線のなす面積値であり、後者は原曲線上の点から最も近い位置にある近似曲線上の点までの距離の総和である。従来からこのような特徴ある点周辺の近似は面積誤差に基づく手法では正確でない場合が多かった。後者の誤差算出式は原曲線上の点の特徴（曲率値、鋭角点など）に基づいて重み付けが可能であり、より高度な制御が可能である。ゆえに本稿では、誤差算出法として最小距離誤差を用いることとする。算出式、算出法を式(1)及び図2に示す。

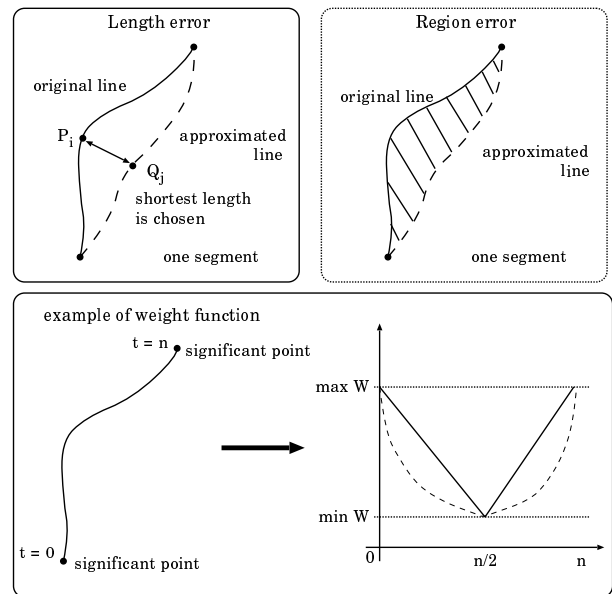


図 2: 誤差定義及び重み関数の例

$$E = \frac{1}{l} \sum_{i=1} \omega_i |P_i - Q_j|^2 \quad (1)$$

ただし、 E は近似誤差値、 l は輪郭線のセグメントにおける点数、 ω_i は原曲線上の点 i の重み関数であり、 P_i が同点の座標を示している。また Q_j は点 i から近似曲線上の最小距離を与える点 j の座標を示している。距

離は計算時間短縮のため距離の2乗を用いる。近似誤差はセグメントの距離で正規化されているため、誤差関数を用いるための誤差閾値は、あらゆる大きさの閉曲線に対し共通に用いることができる。

3.3 処理概要

DP 処理を開始するためには、基準となる点を複数選択する必要がある。ここで、基準となる点は原曲線（閉曲線）上の曲率の大きい点とする手法が提案されている。なぜならば、曲率の大きい点はスプライン補間する際、誤差が大きくなるためである。そこで DP 処理開始に当たって、第一に、曲率の大きい点から数点を有意点として選択する。第二に、連続する有意点間をセグメント化する。なお、有意点間の距離が閾値以上である場合、セグメントを分割する処理を加える。これは、誤差関数の出力はセグメントの長さで正規化されることにより、セグメント長が大きい場合、細かな形状を正確に近似できなくなる現象が発生するためである。第三に、セグメント間の誤差が閾値以上であるならば、セグメント間の全ての点を有意点候補とし、各々近似曲線を得る（閾値以下であるならば、そのセグメントにおける DP は終了する）。第四に、各々得た近似曲線について最小距離誤差が最小となる点を有意点として最終決定する。第三、第四の処理を誤差が閾値未満となるまで再帰的に行うことで、原曲線を忠実に再現した曲線を近似可能である。

3.4 実験 - 近似曲線結果 -

簡単な形状と複雑な形状の2つのサンプル閉曲線（図3）に対し提案アルゴリズムを実行し、近似画像を評価する。本検討では、あるセグメントに対し近似処理を実行するか否かを判断する閾値をパラメータとして用い、近似関数としてスプライン補間を用いた。

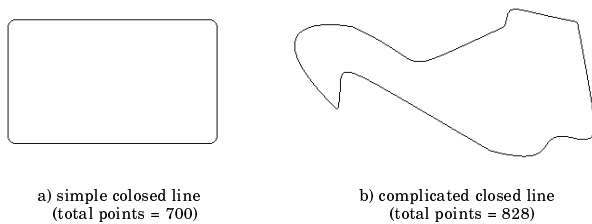


図 3: 輪郭線サンプル-1

図4に示した結果より分かることは以下の通りである。近似誤差関数を用いるための誤差閾値が大きい場合、抽出された有意点数は少なく近似曲線は大きく歪む。一方、誤差閾値を小さくするにつれ、抽出された有意点数は増加し、近似曲線は原曲線に近づくことが主観評価により認められる。

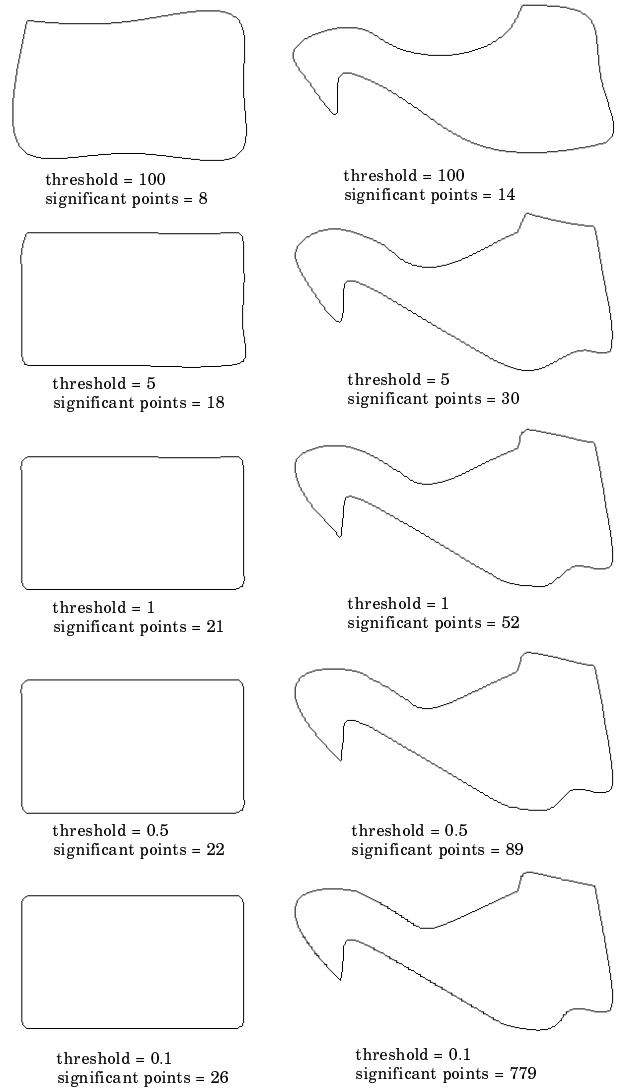


図 4: 近似曲線結果

3.5 実験 - 近似誤差評価 -

次に前実験の結果を客観的に評価するために、点数削減率と誤差値の関係を求める。サンプル画像として図3に加えて図5を用いた。式(1)で求まるセグメント間の誤差値を全セグメントにおいて加算した値を閉曲線の誤差値として用いる。また点数削減率は閉曲線上の有意点数を閉曲線上の全点数で割った値を%表示した値とする。実験結果を図6に示す。

点数削減率と近似誤差の関係は、点数削減率が大きくなるに従い、 $E = 0$ を漸近線として誤差値は単調減少する。一方、点数削減率が小さくなるに従って、誤差値は発散する。データ点数削減の観点からは、図6に示した曲線の凸点が最適点であると判断できる（以後最適値は凸点と定める）。この特徴として、1) データ量が発散する、2) 最適値、3) 誤差値が発散の3つの状態を挙げる事ができる。これらの状態を決定するパラメー

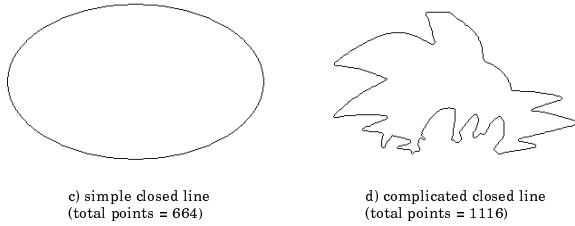


図 5: 輪郭線サンプル-2

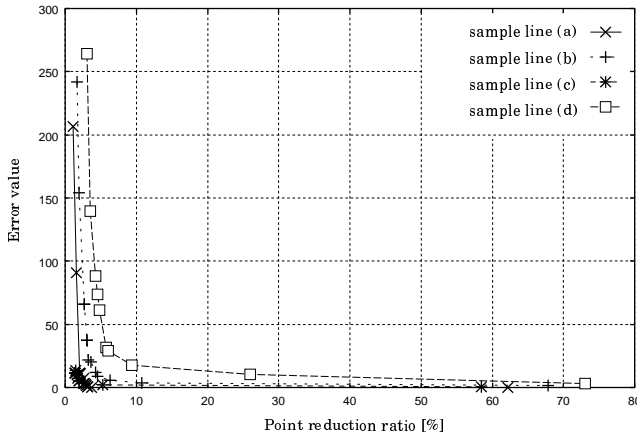


図 6: 点数削減率に対する近似誤差

タは誤差閾値であるので、表 1 に各状態を与える誤差閾値を記す。

サンプル	データ量発散	最適値	誤差値発散
(a)	0.01	1	50
(b)	0.2	1	20
(c)	0.2	3	-
(d)	0.5	1	20

表 1: 各状態を与える誤差閾値

図 6 及び表 1 から導かれることを 2 点挙げる。第一に、最適値を与える誤差閾値はサンプル閉曲線 (c) を除いて、全て 1 と判断できる。この場合、(c) の誤差閾値が 3 と他に比べて大きかったのは、形状がスプライン補間に適していたためであると考えられる。(c) がスプライン補間に適していることは有意点数が少なくても誤差値が発散しないことから分かる。誤差の閾値をあらゆる形状の輪郭線に対して共通の値を用いることは最適化の観点から困難であるが、おおよそ 1 周辺の値であると言える。第二に、複雑な形状の誤差の最適値は簡単な形状の誤差の最適値よりも大きい。2 つ理由が考えられる。

- 複雑な形状では鋭角点などの特徴点が多く、輪郭線のセグメントが増えるためである。重み関数を用い

ているため、輪郭線の距離で正規化しているにもかかわらず誤差値が増える。

- 形状が複雑になるに従って近似曲線の歪みが増るため、誤差の増加につながる。

最後にサンプル閉曲線 (d) は現実のセル画アニメーションにおける均等色領域の輪郭 (特に複雑な例を選択した) の近似結果を図 7 に示す。

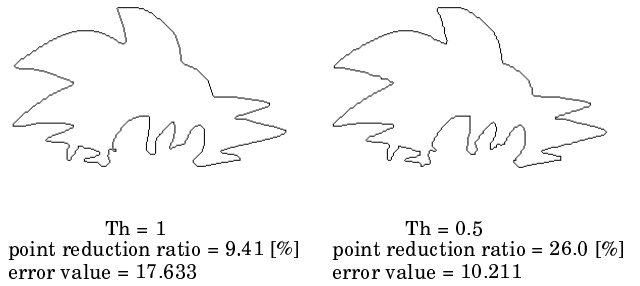


図 7: sample closed line (d) の近似結果

図 7 では、最適誤差値を与える閾値 1 とそれよりも小さい閾値 0.5 の場合の近似結果を示している。閾値が 1 である場合のサンプリングレートは 9.41 [%] であり、実際の有意点数は 105 点である。一方閾値が 0.5 である場合、点数削減率は 26.0 [%] であり、実際の有意点数は 209 点である。誤差値は前者の方が 7.422 大きい、近似曲線はほぼ図 5 の sample line (d) を再現している。

3.6 実験 - 計算時間評価 -

Dynamic programming によって、近似誤差関数を用いて最適な有意点を抽出可能となる。しかしながら、対象となる輪郭線セグメント上の全点を有意点候補として処理するため、計算時間が大きいことが問題である。よって前実験における計算時間を評価し、計算時間に与える要素に関して考察する。サンプル閉曲線 (a) ~ (d)、また閉曲線の大きさを比較するためにサンプル (c) を約 3 分の 1 に縮小した閉曲線 (e) に対し、本提案アルゴリズムの計算時間を測定した。サンプル閉曲線 (e) の全点の数は 208 点である。測定するのは基準点の探索開始から全有意点抽出終了までの時間である。結果を図 8 に示す。ただし、横軸は誤差の閾値である。また、評価を行った実行環境は、OS: linux (kernel 2.2.14), CPU: Pentium III 650MHz, RAM: 128MB である。

結果より分かる点は次の通りである。(e) 以外のサンプル閉曲線は、閾値が 0.5 より小さくなると計算時間が大きく上昇する。一方、閾値が 2 以上であると、計算時間はほぼ横ばいである。閾値が 0.2 以下になると、sample line (a) を除いて、複雑度に関わりなく計算時間が増加していることが分かる。閉曲線 (e) は他の 4 閉曲

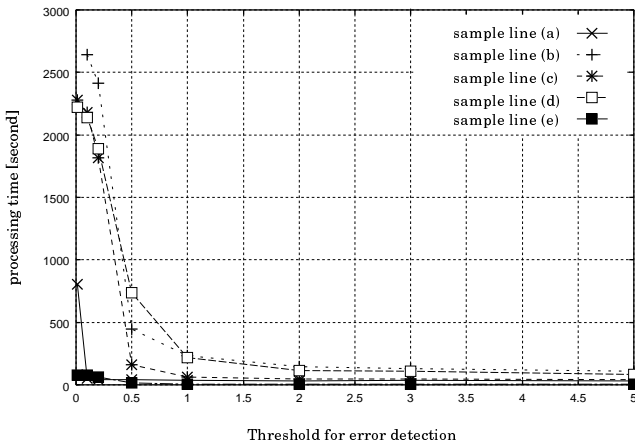


図 8: 計算時間評価結果

線と比較して、閾値が小さくなくても計算時間は大きく増加しない。このことからマクロな視点では、計算時間はデータ点数に依存すると考えられる。ミクロな視点で計算時間に着目すれば、必ずしもデータ点数に計算時間は依存していない。一例を挙げれば、閾値が 0.1 以下では、サンプル閉曲線 (c) の実行時間はサンプル閉曲線 (d) の実行時間よりも大きい。閉曲線の全点数は (c) が 664 点であるのに対し、(d) は 1116 点である。以下ミクロな観点で計算時間に影響を与える要素を考察する。

● 各セグメントの近似関数不適合

今回の近似関数としてスプライン関数を用いた。スプライン関数は緩やかな曲線の補間に適しているが、鋭角点、細かな形状などの近似に適していない。本アルゴリズムはセグメント単位で誤差関数による評価を繰り返すため、あるセグメントが近似関数に適している場合、繰り返し演算回数は大幅に削減される。ゆえに、輪郭線上の全点数が大きい場合でも計算時間は大幅に削減され得る。計算時間に影響を与える要素として、1) セグメント不適合 (誤差閾値の関数であると予測される)、2) 不適なセグメント数及び 3) 同セグメントにおける点数が挙げられる。

● スプライン補間限界

有意点を増加しても誤差値が減少しない減少がセグメント内で発生する場合がある。これはスプライン補間の精度限界が原因である。精度限界が発生局所的に発生するため計算時間が変動する。精度限界に伴う有意点が局所的に抽出される例を図 9 に示す。図内に示した正方形のマークの中心が有意点を表している。点線で囲む領域に有意点が集中的に抽出されていることが分かる。

セル画アニメーションの符号化を考えた場合、コンテンツ制作は 1 週間単位で行われるものであり、リアルタ

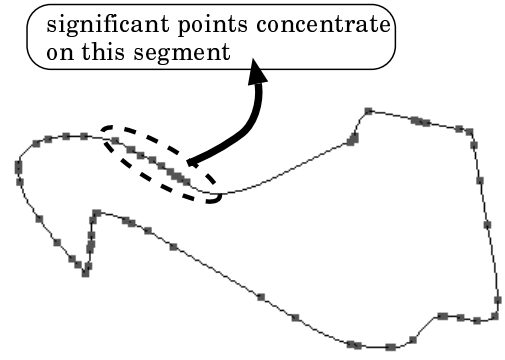


図 9: 有意点局所的抽出例

イム処理への要望はあまり高くない。計算時間の最適化に関してはこの点を考慮する必要がある。

4. 有意点数制御方式 –Feedback algorithm–

セル画アニメーションの 1 フレーム内には様々な形状の輪郭線が存在するため、単一の閾値により全ての輪郭線の近似誤差最適値を得ることは困難である。形状近似の正確さに重きを置き、誤差閾値を小さく設定すると、ある輪郭線において有意点数が発散するか、もしくは一部のセグメントに集中的に有意点が抽出されることが上記の実験より判明した。一方で小さな点数削減率を主眼において誤差閾値を大きく設定すると、複雑な形状の近似誤差が増え、近似曲線の歪みが主観評価でも明らかになってしまう。従来から誤差閾値のみで有意点数を正確に制御することは困難であったが、提案する Feedback algorithm を併用することで、有意点数制御が可能となり有意点数発散問題を解決できる。さらに本手法により、輪郭線単位もしくはフレーム単位で点数削減率の正確な制御が可能となる。

4.1 Feedback algorithm

有意点抽出のために用いた Dynamic Programming を逆方向 (抽出した有意点を削除する方向) に用いるという意味で、Feedback algorithm と定義する。本アルゴリズムでは、抽出した全有意点の中から 1 つずつ仮に削除し、近似誤差を順に算出していく。最も小さい誤差を与える有意点を実際に削除する。この処理は削除目標の点数削減率に達するまで繰り返し実行される。本アルゴリズムはセグメント単位で処理しているのでは無いため、輪郭線の中で最も冗長な有意点から順番に削除することが可能である。なお本アルゴリズムの繰り返し演算数は $\sum_{k=0}^{N_r-1} (N_s - k)$ と表される。ただし、 N_r は削除する有意点数、 N_s は本アルゴリズム実行前の有意点数である。

4.2 評価結果

サンプル閉曲線 (b) に対して実験を行う。この場合の最適誤差閾値は 1 である。そこで、誤差閾値を 0.5 に設定し、有意点が局所的に集中発生する状況から Feedback algorithm を実行する。この場合、目標点数削減率をパラメータとする。結果画像を図 10 に示す。

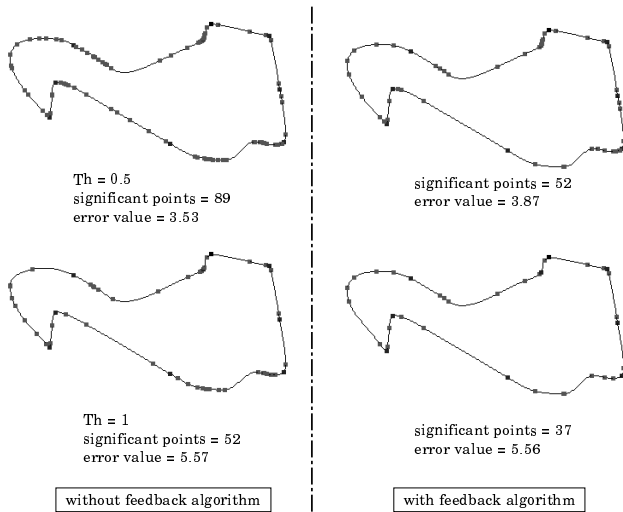


図 10: Feedback algorithm 近似結果

図 10 では Feedback algorithm を用いない場合と用いた場合の比較をしている。Feedback algorithm を用いない場合、データ量が局所的に発散する時 (閾値 0.5) の近似結果と最適閾値 1 の近似結果を示した。また、Feedback algorithm を用いた場合、目標のサンプリングレートを実現した近似結果 (有意点数 52 点) と本アルゴリズムを用いずに誤差閾値を 1 とした時の近似誤差にほぼ等しい点数削減率の結果 (有意点数 37 点) を示す。結果より、有意点数 52 点のとき、Feedback algorithm を用いた方が近似誤差が 1.70 小さいことが分かる。また、Feedback algorithm を用いて最適誤差値まで有意点数を削除すると有意点数が従来法と比較して 15 点減少したことが分かる。Feedback algorithm の性能を評価するために、有意点数と近似誤差値の関係を図 11 に示す。

図 11 に示した結果より次の 2 点に分かる。1) Feedback algorithm を実行することで、従来法と比較して大幅に誤差値を削減することが可能である。さらに、2) 局所的に発散した冗長な有意点を削除可能となった。以上の 2 点は次のように説明できる。有意点を抽出するためには、輪郭線をセグメント化しなければ、細かな形状を近似できない。しかしながら、セグメント化及び一定値以下の閾値を用いることで、本来必要な有意点と共に冗長な有意点まで抽出されてしまう。一方、Feedback algorithm は輪郭線全体の中で不要な抽出点から優先的に削除するので、冗長な抽出点を除去可能となる。

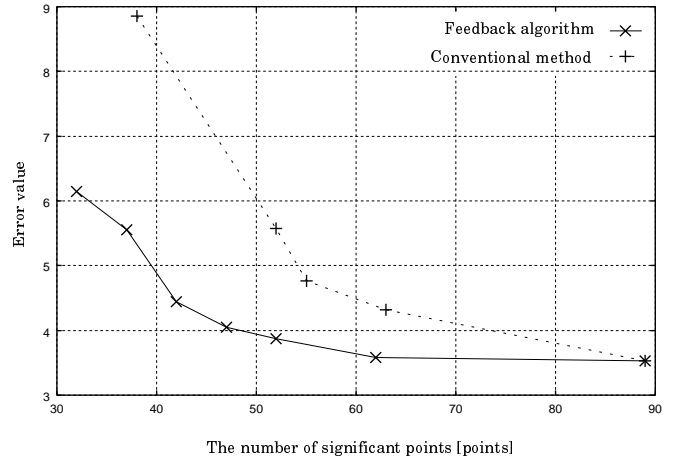


図 11: Feedback algorithm による近似誤差

5. まとめ

本稿では、輪郭線の形状を符号化するために、Dynamic Programming により補間処理に最適な有意点 (標本化点) を抽出するアルゴリズムを提案した。実行時間評価によって、点数に計算時間が大きく依存し、またその他として、セグメント近似不適度、スプライン近似限界が計算時間増加の要因となることが判明した。セグメント化に基づく Dynamic programming による近似法では閾値に依存して、データ量が局所的に発散する問題が生じる。この問題解決のために、冗長な有意点を削除及び点数削減率制御を実現する Feedback algorithm を提案した。本アルゴリズムは従来法の点削減率にまで冗長点を削除した際、誤差値が従来法の値よりも小さくなることが判明した。なお、我々は直線・スプラインハイブリッド符号化方式を開発しており、本提案手法を基に W3C 勧告の SVG (Scalable Vector Graphics) に出力可能である。

参考文献

- (1) 古角, 渡辺, 小林, "アニメーション映像におけるスプライン/DCTハイブリッド符号化の一検討", 情処学研報, AVM24-3, 1999
- (2) 宮澤, 亀山, 渡辺, 阪谷, 富永, "アニメーション画像符号化における輪郭線抽出と近似方式の検討", 情処学研報, 2001-AVM32-12, 2001
- (3) U.E.Ramer: "An iterative procedure for the polygonal approximation of plane curve", CGIP 1, pp.244-256, 1972.
- (4) F. Lu, E.E.Milios, "Optimal Spline Fitting to Planar Shape", Signal Processing, vol.37, pp.129-140, 1994.