

解説



2次元コンピュータグラフィックス技法†

中嶋正之†† 安居院 猛††

1. はじめに

コンピュータグラフィックス (CG) 技法とは、コンピュータを用いて、文字、図形、画像、立体を生成・表示する技法であり、コンピュータの性能が高まり、低廉化し、かつ周辺装置の充実にとともに、その応用分野が急速に広まっている。

コンピュータを用いて画像や図形を扱う関連分野として「画像処理」の分野がある。

画像処理は、主として階調画像、文字、図形などの画像情報を解析・認識・処理する、人工知能を指向した研究分野である。それに対しコンピュータグラフィックスは、主として画像を発生・表示し利用する応用指向の技法であるが、画像処理とコンピュータグラフィックスの境界は必ずしも明確ではない。

これらの二つの分野においては、画像情報のデータベースの構築、符号化、モデリング方法など、各種の処理過程において多くの共通の手法が使用されている。ともに、膨大な画像情報 (数 M から数 10M バイト) を扱い、かつ、人間の視覚情報を扱う計算機応用分野である¹⁾。

コンピュータグラフィックスの技法は、2次元の平面だけに限定された画像を扱う「2次元コンピュータグラフィックス」と、3次元空間を設定し、主として立体形状までを扱う「3次元コンピュータグラフィックス」に大きく分けることができる。

本稿では、2次元のコンピュータグラフィックスの基本的な技法について述べる。

2. 文字、図形のグラフィックス表示

2次元コンピュータグラフィックスは、ある画素点を指定した色に変換する、曲線や直線などの線を描く、また、ある領域を指定した色に変換する作業が、

最も基本的な操作である。その中で、曲線や直線を描画する作業が最も重要な機能である。

2.1 直線の生成と表示

図形データの表現方法として、最も簡単な方法が直線を用いた表現方法である。直線の生成は、Bresenham の方法²⁾を用いることにより高速に生成が可能であり広く利用されている。

ここでは直線の生成と表示に関係する主な技法について紹介する。

直線近似法

描画対象が曲線の場合、それを直線群で表示するため、直線近似法が使用される。直線近似化では、曲線を忠実に近似するような折点を決定する方法が最も重要である。折点を選択する方法としては、大きく分けて曲率を利用する方法と、距離を利用する方法³⁰⁾がある。

曲率を利用する方法はデジタル曲線上をトレースして、小さな曲率点を逐次選択するものである^{31,32)}。たとえば、デジタル曲線の曲率を、曲率を求めたい点を中心として、曲線に沿って前後 N 画素離れた点を結ぶ 2 本のベクトルを考え、そのベクトルの方向の差 θ をパラメータとして定義し、 θ があるいき値以上であれば折点とする方法が利用されている³³⁾。

曲率を計測して折点を求める方式では、図-1(a)に示すように鋭い角がある場合は、正確に折点が求まる。しかし(b)図に示すように、滑らかに変化する曲線上での折点は検出しにくい。そこで、複数のベクトルを曲線上に設定するベクトルトレサによる直線近似化法が提案されている³⁴⁾。

また、1パス目で大まかな直線近似を行い、2パス目で、さらに最良な直線近似を行う方法³⁵⁾、最適な折線近似法³⁶⁾などが提案されている。

ラスタ・ベクトル変換

CAD 図面の自動読み取りや、ベクトル専用型グラフィックス装置 (プロッタ、ベクトル型ディスプレイ、など) へのグラフィックス要素の出力を目的とし

† 2-dimensional Computer Graphics Technologies by Masayuki NAKAJIMA and Takeshi AGUI (Imaging Science and Engineering Lab., Tokyo Institute of Technology).

†† 東京工業大学工学部象情報工学研究施設

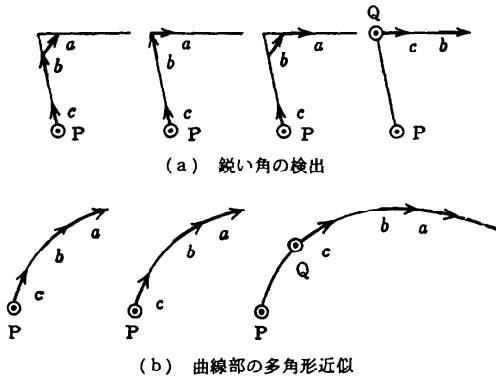


図-1 ベクトルトレサによる多角形近似

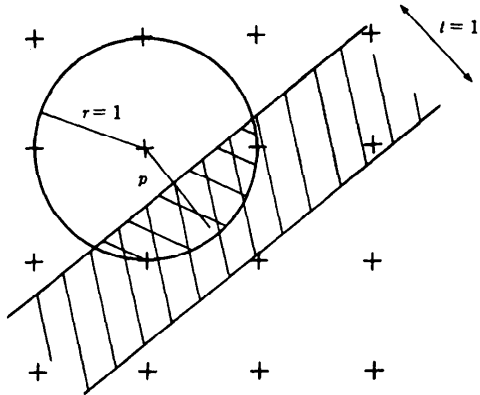


図-2 重なる面積を利用する方法

て、ラスタ・ベクトル変換が利用されている^{9),10)}。

ラスタ・ベクトル変換とは、スキャナや TV カメラから入力されたラスタデータを線分の列すなわちベクトル列のデータに変換する手法をいう。

ラスタ・ベクトル変換は、図形や文字の拡大・縮小、情報圧縮さらにパターン認識にも応用されている。ラスタ・ベクトル変換は、各種の方式が提案されているが、その手順を列挙すると以下ようになる。

- (1) ラスタデータ→ベクトルデータ
- (2) ラスタデータ→細線化→ベクトルデータ
- (3) ラスタデータ→細線化→チェーンコード→ベクトルデータ (逐次処理)
- (4) ラスタデータ→細線化と記号成分除去→チェーンコード→ベクトルデータ
- (5) ラスタデータ→チェーンコード→ベクトルデータ (並列処理)

(1)のラスタデータから直接ベクトルデータを抽出することは、対象データが図面のような比較的細い線から構成されるものに限られ、通常は、ラスタデータを細線化し、チェーンコード列を求め、その直線近似によりベクトルデータを求める(3)の方式が利用されている。

アンチエイリアジング

ラスタ型のディスプレイを用いて図形を表示すると、斜めの線や縁がギザギザになる。これはエイリアジング (aliasing) またはジャギーと呼ばれ、このジャギーを少なくする方法は、アンチエイリアジング¹¹⁾と呼ばれている。アンチエイリアジングには、ディスプレイの解像度を、ジャギーが識別できない程度に高くするハード的な方法と、画素の明るさを変化させるソフト的な方法がある。

アンチエイリアジングのソフト的な手法の一つに、線の周辺の画素をぼかす方法がある。たとえば、黒い線を引く場合、近傍の画素を灰色にすると、線はぼけて見える。すなわち、ジャギーの少ない線を表示することができる。この方法は、古くは織物のタペストリなどにもみられる方法である。線の周辺をぼかすには、次のような方法がある。

線の中心からの距離を用いる方法は、始点と終点の2点を指定して線を引く場合を対象とした方法で、各画素と線の距離を用いる。代表的な方法としては、図-2に示すように、線を太さをもったものと考え、各画素点を中心に、半径 r の円を設定し、その円と重なる面積に比例した階調を、その画素に与えるものがある¹²⁾。また、同様な例として、直線からの距離に反比例した階調を与える方法¹³⁾もある。

その他ローパスフィルタを利用する方法¹³⁾、色系列を利用する方法¹⁴⁾、など多くの方式が提案されている。

2.2 自由形状の描画

われわれの身の回りに見受けられる多くの物体の形状は、簡単な関数で表現することがむずかしい自由形状であることが多い。これら自由形状を計算機に入力する方法としては、グラフ用紙の上に対象形状を描き、その形状を構成する点のデータを入力することで行われる。データを入力し、形状を構成するすべての点を入力しようとする膨大な作業をしいられるので、対象形状を比較的忠実に表す特徴点を定め、それらの座標値だけを入力することにより、ディスプレイ上に形状を再現する方法が広く利用されている。

図形データの再現の方法として、最も簡単な方法が2.1で述べた直線近似の方法であるが、特徴点の数が

少ない場合は、不自然な形状になるという欠点がある。そのため、良質の形状を再現したい場合は、滑らかな線の生成が行われる。

比較的少ない特徴点から、元の滑らかな形状を復元する方法としてスプライン関数 (spline function) が用いられている¹⁾。

スプライン

スプラインとは、指定された点を通る滑らかな曲線を描くための「自在定規」という意味から名付けられている。

従来、滑らかな曲線近似法としては多項式近似が用いられていたが、スプライン曲線は多項式近似に比べて振動が少なく、美しい近似曲線が得られるという特徴をもっている。また、スプライン関数は、区分的に近似する関数として与えられるため、多項式近似のように、ある区分に対する近似処理が、他の区分に対して大きな影響を及ぼさないという特徴がある。そのため、直線近似に比べて、少ない『節点 (knot)』と呼ばれる特徴点で、比較的忠実に元の形状が再現される。

スプライン関数は、離散的な点を与えられた場合、それらを結ぶ滑らかな曲線を与える近似関数であり、 $N+1$ 個の節点をもつ m 次スプライン曲線は、各区分間ごとに異なる多項式で表される。「 m 次スプライン関数」とは $(m-1)$ 次までの導関数が連続となり、隣り合う区間における二つの多項式は、各節点において滑らかに接続されるという条件を満たす関数である。

スプライン関数にはさまざまな次数のものが考えられるが、1次や2次では近似の精度が不十分であり、4次以上は膨大な計算が必要であるため、コンピュータグラフィックスの応用には、3次スプライン関数が最もよく利用される。

B-スプライン

スプライン関数は、節点を通る滑らかな曲線を発生させる関数であるが、それに対し、必ずしも節点を通らないでもよい方式として、Bスプライン曲線 (Basis-spline)²⁾ やベジエ曲線がある。特にBスプライン曲線は、高速かつ容易に生成が可能のため、レーザープリンタにおける高品質な文字の生成などに使用されている。

フラクタル曲線

ある離散的な点列を滑らかな線で結び元の形状を忠実に再現する方法がスプライン関数である。しかし、滑らかに結ぶことにより、元の形状の特徴を失ってし

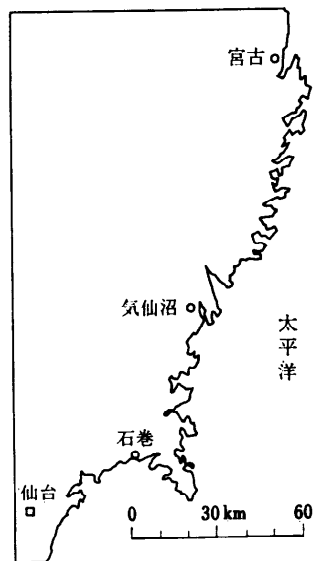


図-3 陸中海岸

まう場合がある。

たとえば、図-3 に示す陸中海岸の形状は、宮古から複雑に曲がりくねって石巻に至っている。この陸中海岸の形状を近似表現する際、スプライン関数を用いて滑らかに結んでしまうと、元来、複雑な形状であるこのリアス式海岸の特徴を失ってしまう。そこで、複雑な形状は、極力その複雑性に合わせて再現しようとするのが、フラクタルの考え方を応用した自由曲線の描画法³⁾である。

2.3 文字パターンの生成



図-4 Bスプラインによるフォント作成例

文字のパターンのデータ形式としては、従来キャラクタージェネレータを用いたラスタ形が主であったが、高品質な文字が生成可能なベクトル形がデスクトップパブリッシングの分野において使用され始めている。ベクトル形フォントは、任意の拡大、縮小、回転が行えることや書体の変換が比較の容易なため、各種のベクトル文字生成に関する研究が活発に行われている。

たとえば、漢字のストロークの解析による楷書毛筆体の生成¹⁶⁾、フーリエ記述子を用いた毛筆体の一生成法¹⁷⁾、ラスタ文字のベクトル変換方法¹⁸⁾、多項式近似を用いた輪郭線フォントの生成法¹⁹⁾が提案されており、その他各種の文字パターンの生成アルゴリズム²⁰⁾が提案されている。図-4に、Bスプライン曲線で高品質な文字フォントを作成した例を示す²⁹⁾。

3. カラー画像のグラフィックス表示

本章では、主にコンピュータを用いたカラー階調画像の表示技法について述べる。

3.1 彩色技法

画像入力装置などから得られた2値の線図形に対し、顔とか体とか同じ意味をもつ各連結した白領域ごとに、同一の色を付与する作業を“彩色処理またはフ

表-1 フィルアルゴリズムの種類

ベーシックフィル アルゴリズム	2値図形を対象とした 基本的なフィルアルゴリズム
バウンダリフィル アルゴリズム	2値またはカラー階調を対象とした、 指定した閉曲線内のフィルアルゴリズム
チントフィル アルゴリズム	白黒階調画像に対するカラーによる フィルアルゴリズム
テキストチャフィル アルゴリズム	各種テキストチャによる フィルアルゴリズム

ィルアルゴリズム”と呼ぶ。フィルアルゴリズムは、“コンピュータペインティング”または“彩色アルゴリズム”とも呼ばれ、多くの方式が提案されている。主なフィルアルゴリズムとその特徴を表-1に示す。

この中で最も簡単なフィルアルゴリズムは、図-5に示すように2値線画を対象としたもので、彩色を施すべき連結している白画素領域を、指定した色に変換する手法である。このフィルアルゴリズムは別名“ベーシックフィル”とも呼ばれ、最も基本的なフィルアルゴリズムである¹¹⁾。

以下に代表的なフィルアルゴリズムを示す。

チントフィルアルゴリズム²¹⁾

“チントフィル (Tint fill) 作業”とは、白黒階調つ

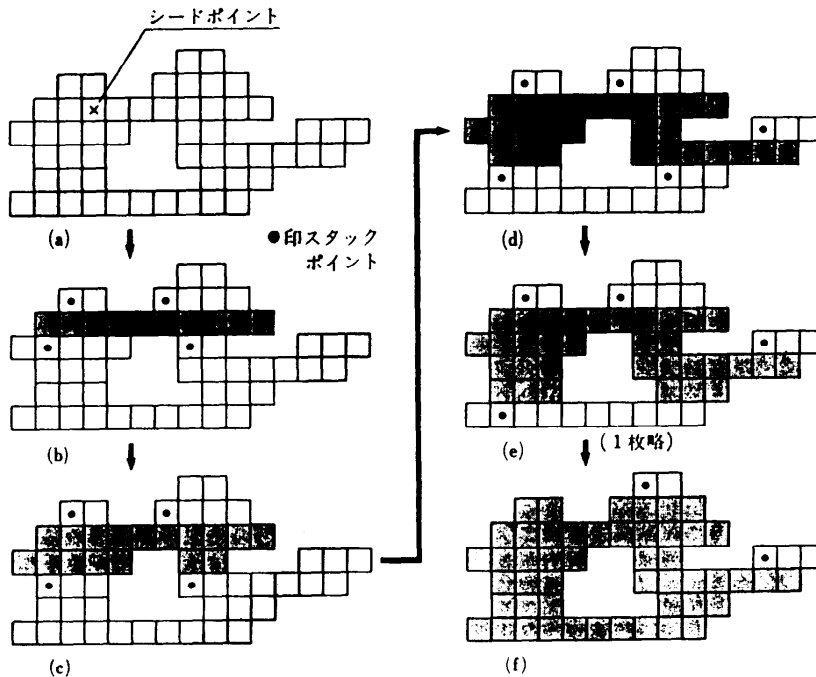


図-5 ベーシックフィルアルゴリズム

き (shades of gray) 境界線内を新しい色で塗り込む作業である。ベーシックフィルの場合は、2値画像を扱っているため、“8連結の黒面素領域内の4連結白領域をすべて同じ色とする”というきわめて明確に定義される作業である。しかし、チントフィルアルゴリズムの場合、色塗りをすべき領域の範囲が明確には定義されず、領域の決定と階調部分に対する色の付与方法が問題となっている。

バウンダリフィル

階調画像からシェーディングエッジを抽出し、エッジ内の領域を指定した色Cに塗る作業を“バウンダリフィル (Boundary fill)”と呼ぶ。バウンダリフィルは、2値図形におけるベーシックフィルを直接利用する方式で、最も簡単に実行できる。バウンダリフィルでは、その中心線の内部の領域を無条件に指定色へ変換するので、カラー変換処理に利用される。

テクスチャフィル

フィルアルゴリズムを“ある範囲内を同一の色に変換する作業”と定義したが、色だけでなく、ある模様を境界線内部に一樣に配置する作業も含めることができる。この操作を一般に“テクスチャフィル (Texture Fill) アルゴリズム”と呼び、他のフィルアルゴリズムと区別している。図-6はテクスチャフィルを施した一例である。

さらに、フィルアルゴリズムとして、他の作業で求まっているパターンを閉領域内に配置することもある。この場合は、一般に“パターンフィル”と呼んでいる。

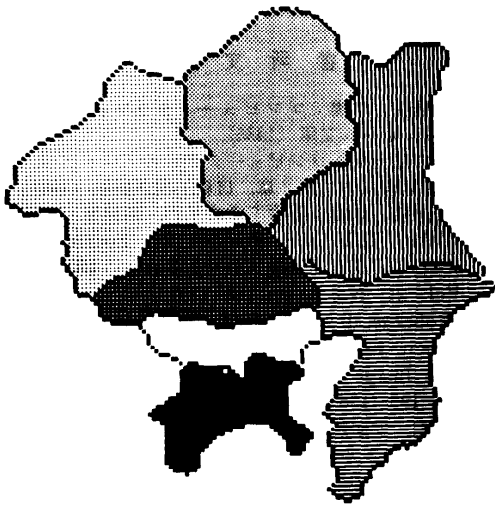


図-6 テクスチャフィルの例

3.2 疑似表示技法

計算機を利用したカラー画像の表示技法として、少ない色数を用いて極力、美しい表示を得るための画像表示技法が開発されている。その代表的な手法について示す。

ディザ法

階調画像を少ないレベル数で表示すると偽の輪郭線と呼ばれる原図形にはない明確な輪郭線を生じ、著しく画質の劣化を招く。

そこで、偽の輪郭線を目立たなくするように雑音を加える方法があり、ディザ法 (dither method) と呼ばれている。

このディザ法は、CRT ディスプレイ、プラズマディスプレイなどの表示装置だけでなく、レーザプリンタやインクジェットプリンタなどの印字装置を用いて階調画像を描画する方式としても使用されている。

ディザ法には、各種の方式が提案されているが最も広く利用されているのが組織的ディザ法である。この方法は、ディザ行列を導入し、各行列の値をいき値とし、対応する座標点の画素の濃度と比較して、白か黒かを決定する。すなわち、入力画像の濃度が対応するディザマトリクスの値よりも大きいか同じときは黒の1、それ以外は0を付与するというきわめて簡単な方法である。

濃度パターン法

ディザ法は、与えられた画像の各画素1点に対して、表示画素も1点に対応させる方式である。それに対して、原画像の1画素点に対して、表示画像に $n \times n$ の配列の画素を割り当てる方式を濃度パターン法という。最も簡単な濃度パターン法は、 2×2 の配列を用いて5段階の濃度を表現するものである。すなわち単位面積当たりの黒画素の密度を変化させて階調を表現することになる。この際 2×2 の領域を1画素に対応させているため、本来は4bit (16段階) の情報量があるがそれを5段階の階調情報として使用している。この余分な情報を他の目的に使うという研究がなされており、文献²⁸⁾では、文字データの画像への埋め込みに使用されている。

誤差分散法²⁹⁾

誤差分散法は、ある画素点 (x, y) の赤成分の濃度を $R_{x,y}$ とし、この $R_{x,y}$ に対応する最も誤差の少ない表現可能な多値階調レベル値を P_i としたとき (1) 式で生じる誤差を注目している画素点の右隣 $(x+1, y)$ 、右下 $(x+1, y+1)$ および下 $(x, y+1)$ の各

点に(2)式に従って配分し、この処理を繰、青の成分 $G_{x,y}$, $B_{x,y}$ について行い、さらにすべての画素について施すものである。

$$E_{x,y} = R_{x,y} - P; \quad (1)$$

$$\left. \begin{aligned} \bar{R}_{x+1,y} &= R_{x+1,y} + E_{x,y} \times 3/8 \\ \bar{R}_{x+1,y+1} &= R_{x+1,y+1} + E_{x,y} \times 1/4 \\ \bar{R}_{x,y+1} &= \bar{R}_{x,y+1} + E_{x,y} \times 3/8 \end{aligned} \right\} \quad (2)$$

ここに $\bar{R}_{x+1,y}$ などは誤差を配分して新しく設定された画素点 $(x+1, y)$ における赤成分の濃度である。

限定色表示

階調画像のディスプレイ表示において、3原色である R, G, B に各 8 ビットの階調を与えると約 1,600 万色を直接表示することが可能となる。しかし、実際に扱う画像内に同時に現れる色数は一般にその一部であり、また画像によっては、きわめて似ている色を 1 色にまとめても自然さを失わない場合がある。

そこで、画像のメモリの節減、作画作業の効率化、ルックアップテーブル方式のディスプレイの利用などを目的として、色数の節減、すなわち限定色表示が行われている。

限定色表示の方式としては、多くの方式が提案されている^{23)~25)}。最も簡単な限定色表示のアルゴリズムとしては、R, G, B の 8 ビット階調値を直交した 3 軸として与えられる RGB 空間内に、1 枚の CG 画像に使われた色を分布させ、色が存在した部分空間のみを色数 (たとえば 256 色) で均等に分割する方法である²³⁾。

α アルゴリズム

計算機を用いたモンタージュ写真の作成における画像合成処理において、ある物体を切り出して、他の物体上に配置した場合、その切りはしが目立ち、劣化が感じられる場合がある。その合成処理において視覚

上、劣化を目立たなくする方法として、 α アルゴリズムがある²⁶⁾。すなわち、図-7 に示すように、画素を表す正方形領域の中に二つの画像、すなわち、背景画像とその上に重ねる物体画像が存在する場合に、それぞれの画像要素の占める面積比に応じた割合で、それぞれの画像要素の色を混合した結果を、その画素の色とする。たとえば、図-7 の画像要素の境界線よりも右下の領域に画像要素 A、左上の領域に画像要素 B が存在する場合、図の斜線で示した画素、すなわち両画像要素の境界線によって分断されている画素に対して、画像要素 A の色と画像要素 B の色とを、それぞれの画像要素の占める面積比に応じた割合で混合した色を与える。

このような混合比は、 α 値と呼ばれている。

4. おわりに

本稿では、2次元コンピュータ・グラフィックス技法として、線や曲線の生成技法およびカラー画像の表示技法の主な技法について述べた。

2次元の CG 技法としては、このほか紙面の都合上割愛したが多くの技法が開発されている。

たとえば、CG 画像に対する圧縮符号化の研究²⁷⁾は、特にコンピュータアニメーションなどにおいて大量の CG 画像を効率良くファイル化したり電送するために重要となっている。また、より自然な各種のテクスチャ画像の生成アルゴリズム、効果的なクリッピングアルゴリズムなど多くの 2次元コンピュータ・グラフィックス技法が提案されている。

参考文献

- 1) 安居院, 中嶋: コンピュータグラフィックスの技法, オーム文庫 (1986)。
- 2) 山口富士夫: コンピュータディスプレイによる図形処理工学, 第 2 章, 日刊工業新聞社 (1981)。
- 3) 安居院, 中嶋: 画像工学の基礎, 昭晃堂 (1986)。
- 4) Zahn, C.T. and Roskies Z.: Fourier Descriptors for Plane Closed Curves, IEEE-C, Vol. 21, No. 3, pp. 296 (1972)。
- 5) Roberge, J.: Data Reduction Algorithm for Planer Curves, Computer Vision, Graphics, and Image Processing, Vol. 29, No. 2, pp. 168-195 (1985)。
- 6) 中嶋, 安居院, 飯塚: 市街化地図に対するパラレルベクトルトレサを用いたグラフ構造解析, 信学論, Vol. 67-D, No. 12, pp. 1419-1426 (1984)。
- 7) 安居院, 飯塚, 中嶋: 原図形に忠実な直線近似

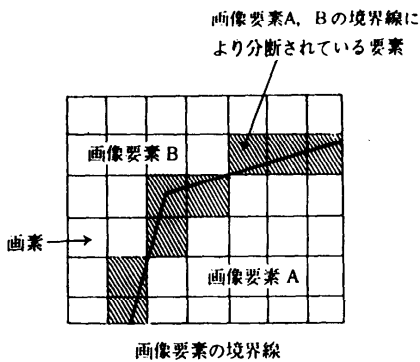


図-7 α アルゴリズム

- 法, 信学論, Vol. 68-D, No. 8, pp. 1539-1540 (1985).
- 8) 佐藤幸男: 平面曲線の最適折線近似, 信学論, Vol. 65-D, No. 9, pp. 1145-1150 (1982).
- 9) 大沢, 坂内: 多次元データ構造を用いた図面処理, 信学論, Vol. 68-D, No. 4, pp. 845-852 (1985).
- 10) 岩田 清: 図面自動読み取りシステム, 信学論, Vol. 68-D, No. 5, pp. 522-528 (1985).
- 11) 安居院, 中嶋, 大江: コンピュータアニメーション, pp. 72-73, 産報出版 (1983).
- 12) Gupta, S.: Filtering Edges for Gray-Scale Display, ACM Computer Graphics, Vol. 15, No. 3, pp. 1-5 (1981).
- 13) Kajiya, J. and Ullner, M.: Filtering High Quality Text for Display on Raster Scan Devices, ACM Computer Graphics, Vol. 15, No. 3, pp. 7-15 (1981).
- 14) 安居院, 中嶋, 北村: 色系列を用いた階調画像に対する彩色処理, 信学論, Vol. 67-D, No. 3, pp. 321-326 (1984).
- 15) Newman, W. M. and Sproull, R.: Principles of Interactive Computer Graphics, McGraw-Hill (1979).
- 16) 張, 真田, 手塚: 漢字楷書毛筆体の計算機による生成, 信学論, Vol. 67-D, No. 5, pp. 599-606 (1984).
- 17) 松本, 山根, 他: フーリエ記述子を用いた毛筆字体の生成法, 信学総全大, 1556 (1986).
- 18) 上原, 安居院, 中嶋, 田原: 漢字 ROM データのベクトル化とその応用, TV 学会技術報告, 10, 9, 1-6 (1986).
- 19) 鳥島, 山崎: 文字輪郭線の区分的多項式による近似, 信学総全大, 1552 (1986).
- 20) 西川, 長田: 輪郭表現による文字パターンの拡大, 縮小方式, 信学総全大, 1558 (1986).
- 21) Smith, A. R.: Tint fill, SIGGRAPH '78, pp. 276-282 (1979).
- 22) 小野: ディザ法, 画像電子学会誌, Vol. 10, No. 5, pp. 388-397 (1981).
- 23) Agui, T., Arai, Y. and Nakajima, M.: A Color Compression Algorithm for Natural Scenes and Animation Pictures, IEEE Workshop on Visual Languages, pp. 2-5 (Dec. 1984).
- 24) 中嶋, 清水, 安居院: 適応形部分空間法を用いたカラー階調画像の表示, 信学技報, IE 8514 (May 1985).
- 25) 吉良, 井上, 福井: 限定された数の代表色による適応型自然色画像表示, テレビ学技報, IPD 84-3 (1984).
- 26) Porter, T.: Compositing Digital Images, Computer Graphics, Vol. 18, No. 3, pp. 253-259 (July 1984).
- 27) 中嶋, 安居院: カラー印刷用製版画像の圧縮符号化法 [1], 印刷学会誌, Vol. 25, No. 1, pp. 2-12 (1987).
- 28) 中村, 松井: 濃度パターンを用いた濃淡画像とテキストデータの合成符号化, 信学論, Vol. J70-B, No. 12, pp. 1475-1481 (1987).
- 29) 大野義夫: デジタル曲線と多面体の平滑化アルゴリズム, コンピュータソフトウェア, Vol. 5, No. 1, pp. 23-41 (1988.1).
- 30) Roberge, J.: Data Reduction Algorithm for Planer Curves, Computer Vision, Graphics, and Image Processing, Vol. 29, No. 2, pp. 168-195 (1985).
- 31) 安居院, 宮田, 中嶋: フラクタル次元に基づいたデジタル図形の擬似符号化法, テレビ誌, Vol. 39, No. 10, pp. 979-987 (1985).

(昭和63年6月6日受付)