

物体形状の Voxel 表現に基づく一定義法†

千葉 則 茂^{††} 青木 富 雄^{†††} 斎藤 伸 自^{††††}

物体を空間立方格子点で均一に標本化し、その幾何形状を定義するという方法を Voxel 表現法と呼ぶこととする。これまで、Voxel 表現法に基づき、陰影画像を生成する方法が3次元画像処理やCGの分野でいくつか提案されているが、いずれも標本化不足による人為的な縞模様が物体表面に発生し、高品質な画像は生成されていない。本論文では、高品質な陰影画像の生成可能な、帯域制限補間に基づいた、Voxel 表現法による幾何形状の定義法を提案する。本文では、まず、ここで提案する定義法による計算例と、これまでに提案されている方法の追試例とを比較し、その有効性を示す。次に、その定義法に対する時間と記憶領域計算量において効率的なアルゴリズムを与える。

1. ま え が き

ここでは、物体を空間立方格子点で均一に標本化し、その幾何形状を定義するという方法を Voxel 表現法と呼ぶこととする。これまで、Voxel 表現法に基づき、陰影画像を生成する方法が3次元画像処理やCGの分野でいくつか提案されているが^{1)~7)}、いずれも標本化不足による人為的な縞模様が物体表面に発生し、高品質な画像は生成されていない。本論文では、高品質な陰影画像の生成可能な、帯域制限補間に基づいた、Voxel 表現法による幾何形状の定義法を提案する。

コンピュータグラフィクス(CG)における物体の幾何形状の定義は、通常、標本とそれに対する補間法として定義される。一般には、多角形面を張ることを補間法とする Boundary Representation (B-rep) や、2次関数や分布関数などの基本立体の集合演算を補間法とする Constructive Solid Geometry (CSG) と呼ばれる定義法が用いられている。しかしながら、3次元画像処理や、自然物などのように複雑な形状を取り扱うCGでは Voxel 表現法によるほうが都合がよい場合も多い¹⁴⁾。Voxel 表現法は、CAD では空間格子法と呼ばれ、そのための3次元配列は、種々の立場から、3次元メモリとか立方フレームバッファ(cubic frame buffer)とか、あるいは、単にボクセルとか呼ばれている。ここでは、3次元配列の1つの要素、す

なわち、立方格子点を指すときは voxel と記し、3次元配列全体、もしくは定義法を示すときは Voxel と記すこととする。また、物体の表面に対応する voxel での表面の法線を定めることも、自然に、形状の補間と考えられるので、陰影画像の生成のための法線決定法も、ここでは補間法として取り扱うこととする。これまでに提案されている画像生成法は、

(a) Voxel 表現から、B rep による表現に変換し、それに対する画像生成アルゴリズムを適用する方法^{1),2)}、

(b) Voxel 表現法の上で直接働く画像生成アルゴリズムを適用する方法^{3)~7)}、

の2つに大別される。(a)は、形状がさほど複雑でなく、既存の画像生成パッケージを利用する場合などには向くが、自然物などのように複雑な形状をもつ物体の画像生成を行うためには、単純な変換法では膨大な多角形面が必要となる。また、(a)での B-rep に対して、CSG による表現に変換する方法も考えられるが、基本立体のあてはめという新たな難しい問題が生じてしまうため、画像生成のためだけの補間法としては高価すぎると言えよう。一方、(b)に対するアルゴリズムには単純でインプリメントが容易であり、ハードウェア化に向くものが多い。しかしながら、これまでの近傍 voxel の参照のみで行う補間法では、縞模様などが発生し、高品質な画像が得られにくいという欠点が存在する。

均一な標本化に基づく最も妥当な標本化法と補間法としては、よく知られた標本化定理にしたがった、十分な標本化密度による帯域制限信号の標本化とそれにより得られた標本に対する帯域制限補間があげられよう。しかしながら、多くの物体の境界は、一般には明確であり、物体は帯域制限信号とは考えられない。ま

† A Voxel Based Geometric Modeling by NORISHIGE CHIBA (Department of Computer Science, Faculty of Engineering, Iwate University), TOMIO AOKI (SONY Corporation) and NOBUJI SAITO (Department of Electrical Communications, Faculty of Engineering, Tohoku University).

†† 岩手大学工学部情報工学科

††† ソニー(株)

†††† 東北大学工学部通信工学科

た、Voxel の解像度、すなわち、標本化密度のほうが決まっていると考えるほうが現実的であり、標本化密度を任意のものとするにはできない。したがって、物体を帯域制限（ローパスフィルタリング）し、すなわち、ぼやけた物体としてから、標本化する必要がある。また、それより得られた標本に対する帯域制限補間では、標本化前のぼやけた物体が“正確”に復元されるのであり、もとの境界の明確な物体の（近似的）定義が必要となる。ここで提案する形状定義法では、もとの物体の表面は帯域制限信号（3次元のなめらかな濃度関数）の1つの等濃度面として定義し、その表面のある点での法線ベクトルは濃度関数のその点における濃度勾配として定義する。

本方法の実現上の問題点としては、次のことがあげられる。

- 理想的な標本化を行うための帯域制限を高速離散フーリエ変換（FFT）により実行するためには、前もって標本化が必要となるというジレンマが存在する。したがって、帯域制限を行うための標本化によるエイリアスの影響を実質的に無視できるような標本化密度としなければならず、帯域制限信号の標本を記憶するための Voxel 以上の大きさの3次元配列が必要となる。さらに、それに応じた時間計算量も必要となる。
- さらに、帯域制限補間を FFT により高速に実行するためには、Voxel の補間倍率の3乗倍の容量の3次元配列と、それに応じた時間計算量が必要となる。

本論文では、まず、提案する方式の原理的な有効性を示すために、

- これまでの補間法の追試例と、
- 筆者らの帯域制限補間法による計算例

の比較を行う。次に、実現上の問題点を解決するための、

- 帯域制限と帯域制限補間を効率的に、すなわち、記憶領域計算量と時間計算量の点で、効率的に実行するアルゴリズム

を提案する。

2. これまでに提案されている法線決定法

これまでも Voxel 表現法に対する法線決定法が、いくつか報告されている³⁾⁻⁶⁾。ここでは、その追試と、Sobel のオペレータを用いた方法により得られた陰影付け例を示す。ただし、陰影付けモデルとしては、形

状の補間を評価するだけなので、最も単純な Lambert の余弦法則を用い、入力は、一葉双曲面の内部として定義される物体を用いた。筆者らは、自然物などの複雑な形状をもつ物体の形状定義に Voxel 表現法を使うことを考えているが、ここでの目的は、補間法の評価であるので、理想形状の明確な形状を用いた。Voxel 解像度は $64 \times 64 \times 64$ (voxel) であり、画像は1画素を voxel に対応させ、 64×64 (pixel) の解像度で得たものである。

図1～5にそれぞれ、(1) Z 値のみによる陰影付け法³⁾、(2) Phong の法線補間陰影付け法^{4), 5)}、(3) 十字型の法線決定法⁵⁾、(4) 2次元 Sobel のオペレータによる法線決定法⁵⁾、(5) 3次元 Sobel のオペレータによる法線決定法、による計算例を示す。Sobel のオペレータについては文献9)などを参照されたい。(4)の方法は(3)と同様に距離配列（隠面消去計算後のZバッファと考えればよい）上で動作するものであり、(5)は Voxel 上で直接動作するもので



図1 Z値のみによる陰影付け法
Fig. 1 Depth only shading.

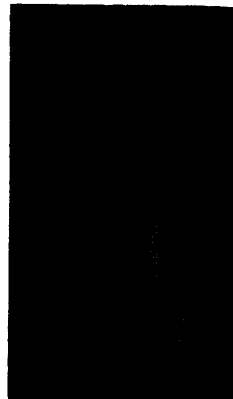


図2 Phong の法線補間法
Fig. 2 Phong's shading.

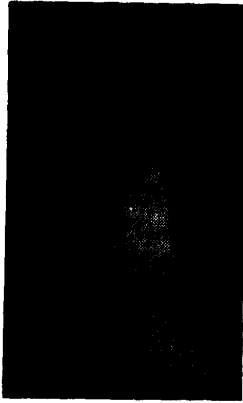


図 3 十字型の法線決定法
Fig. 3 Cross mask shading.



図 4 2次元 Sobel オペレータによる法線決定法
Fig. 4 2-D Sobel operator based normal vector generation.

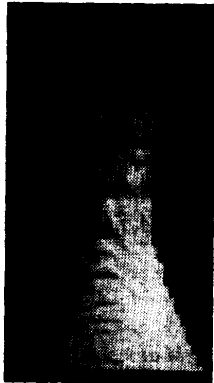


図 5 3次元 Sobel オペレータによる法線決定法
Fig. 5 3-D Sobel operator based normal vector generation.

ある。(1)の方法は法線を用いていないため、品質の高い画像は望めない。(2)-(5)のいずれの方法においても、Voxel 表現のこぼこな表面がなめらかに補間しきれていないことがわかる。さらに、文献 10)

などで、これらの変形も提案されているが、これらと同様に縞模様が発生している。

3. 帯域制限補間法を用いた形状定義法

いま、入力物体が、その外部が“濃度値=0”で、その内部が“濃度値=1”であるような連続空間における濃度関数で与えられたとする(実際は、色や屈折率や透明度をもって定義される)。本論文で提案する Voxel 表現による形状定義法の概要は次のようである。

(1) 形状の標本化は、もとの物体(すなわち、濃度関数)を帯域制限して、ぼやけた物体に変換した後に行う。

(2) 形状の補間は、

(2-1) 標本化定理で示される帯域制限補間法により、標本からぼやけた物体の濃度の補間(復元)を行う、

(2-2) もとの物体の表面は、ぼやけた物体の1つの等濃度面として、その表面における法線ベクトルは濃度勾配として、近似的に定義する。

(1)および(2-1)の詳細を 3.1 節で、(2-2)の詳細を 3.2 節で述べる。帯域制限補間法の説明は付録に示す。

3.1 帯域制限による標本化と補間

もとの物体を表している連続空間における濃度関数を $f(x, y, z)$ とする(図 6 (a)参照)。帯域制限を行うために標本化した離散空間における濃度関数を、解像度 R として (R^3 個の標本点をもつ) $f_R(x, y, z)$ で表す。この f_R の離散フーリエ変換 $F_R(u, v, w)$ にローパスフィルタ $H(u, v, w)$ を掛ける。 $\tilde{F}_R^R = F_R \cdot H$ の離散逆フーリエ変換により、帯域制限された濃度関数 \tilde{f}_R^R を得る。ここで、上付き添え字はローパスフィルタリングのための標本化の解像度を示す。 \tilde{F}_R^R の低域部を集めて得られる解像度 N の配列を \tilde{F}_N^R とし、その離散逆フーリエ変換を \tilde{f}_N^R とする。 \tilde{f}_N^R は \tilde{f}_R^R の標本としても得られる。また、 \tilde{f}_R^R より標本化定理により復元される連続空間における関数を \tilde{f}^R とする。一般に、 $N \leq R$ とし、 N に対して十分な帯域制限により、 \tilde{f}_R^R が得られていれば、 \tilde{f}_N^R から \tilde{f}^R が復元できる。 R が十分に大きければ、実質的に、この \tilde{f}^R は、 f の理想的な帯域制限を行った関数となる。濃度関数 \tilde{f}_N^R を Voxel 表現とする。

\tilde{f}^R を得るには、

• \tilde{f}_N^R から空間領域での帯域制限補間により得る

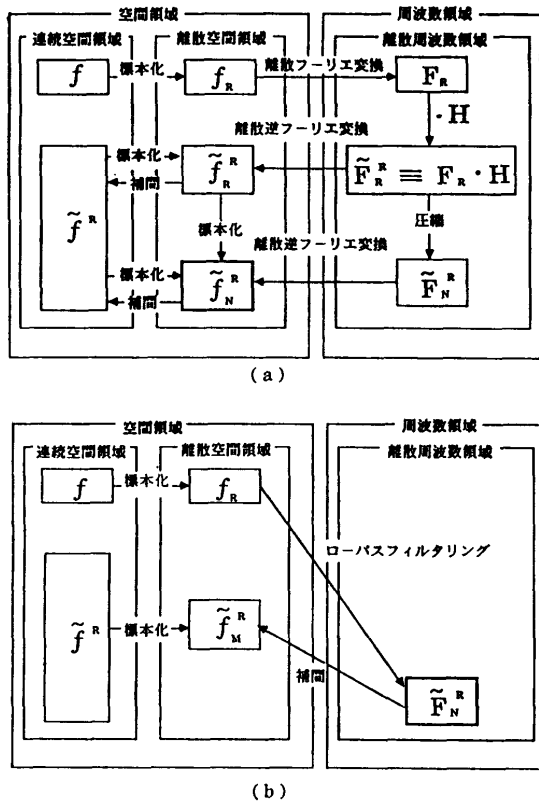


図6 帯域制限補間による形状定義法
Fig. 6 The geometric modeling based on band limited interpolation.

方法と、

- \tilde{F}_N^R から離散逆フーリエ変換による帯域制限補間により求める方法

があるが、後で述べるようにここでは後者による。また、実際は \tilde{f}^R を求めずに、スクリーン解像度 M ($M=LN$) によって定まる補間倍率 L により、 \tilde{f}_M^R を求めている。本論文で提案する形状定義法を図6 (b)にまとめる。以降では、図に示すように \tilde{F}_N^R を Voxel 表現として説明を行うが、もちろん、 \tilde{f}_M^R を Voxel 表現としてもよく、以降でも、混乱のない限り、断りなく両者を Voxel 表現と呼ぶ。 $R=N \leq M$ の場合は、帯域制限されていない物体の標本から、直接、補間倍率 L で帯域制限補間を行うことになり、 $R=M > N$ の場合は、単に、物体の標本そのものをローパスフィルタにより平滑化することと一致する。

R も N も十分に大きければ問題はないが、限られた N に対して良質で倍率の高い補間を達成するためには、 R ができるだけ大きくとれなければならない。しかしながら、現実的には、少ない記憶領域量で大規模な3次元配列 (R^3) が仮想的に実現できるほうが望

ましい。

もし、もとの物体が一度に標本化せずとも、部分的な区画ごとに標本化可能であれば (ほとんどの場合は可能であるが)、5章で述べるように N^3 の R/N 倍程度の記憶領域量で、さらに、3次元フーリエ変換を単に1次元FFTで実行することにより高速に実行できる。実用上は Voxel の定数倍の記憶領域量で実行可能であると考えられる。

\tilde{F}_N^R と \tilde{f}_M^R を求める効率的アルゴリズムは5章で示す。

3.2 物体表面および法線の決定法

物体表面は \tilde{f}^R の等濃度面として定義し、視線との交点は図7に示すように決定する。この交点における法線ベクトルは、交点における濃度勾配として定義するが、これは以下で述べるように、離散逆フーリエ変換により定めることができる。このほか、交点近傍の \tilde{f}_M^R を補間により求め、3次元 Sobel のオペレータを適用し、法線ベクトルを定めるという方法もあるが、紙面の都合上、以降では前者についてのみ記述する。

補間された $\tilde{f}_M^R(x, y, z)$ は

$$\begin{aligned} \tilde{f}_M^R(x, y, z) &= (1/M^3) \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} \sum_{n=0}^{M-1} \tilde{F}_M^R(l, m, n) \\ &\quad \cdot \exp(j2\pi(xl + ym + zn)/M) \end{aligned}$$

で表される。これより、 $\tilde{f}_M^R(x, y, z)$ の濃度勾配の x 成分は、

$$\begin{aligned} \Delta_x \tilde{f}_M^R(x, y, z) &= (1/M^3) \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} \sum_{n=0}^{M-1} \tilde{F}_M^R(l, m, n) \\ &\quad \cdot (j2\pi l/M) \cdot \exp(j2\pi(xl + ym + zn)/M) \end{aligned}$$

で定義でき、他の成分も同様に定義できる。法線ベクトルは $(-\Delta_x \tilde{f}_M^R, -\Delta_y \tilde{f}_M^R, -\Delta_z \tilde{f}_M^R)$ で求めることができる。

begin

{視線を解像度 M のデジタル直線とする。等濃度面を定める等濃度値を d とする。}
視線上の $\tilde{f}_M^R(x, y, z)$ を補間により求める；
視線上を探索し、 d より大きな濃度をもつ点が存在するかを調べる；
もし、存在すれば、その点と1つ前の点との濃度を比較し、 d に近い点を交点とする

end;

図7 交点の求め方

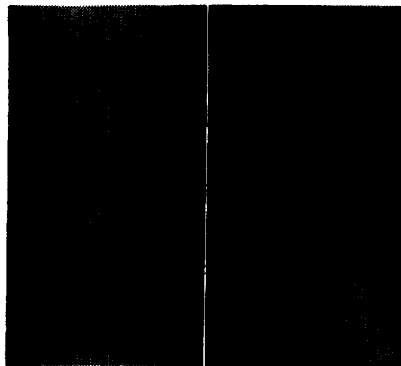
Fig. 7 The procedure for finding an intersection point.

5章で、法線ベクトルを求めるための効率的な帯域制限補間法について述べる。

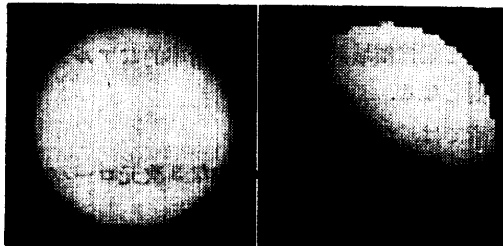
4. 形状定義例

ここでの計算例においては、スクリーンは $Z=0$ の平面とし、各視線ベクトルは $(0, 0, 1)$ とし、物体は拡散反射物体として、等濃度面を定める閾値 d は形状定義における最大濃度の $1/2$ とした。また、帯域制限のためのローパスフィルタとしては、よく知られている指数型フィルタ、

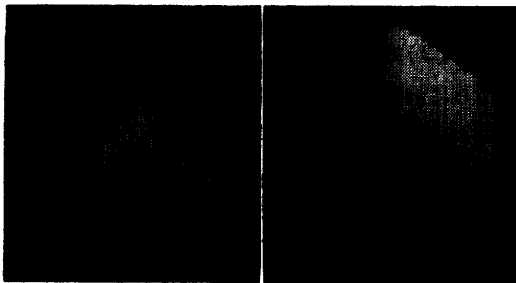
$$H_E(u, v, w) = \exp(-(D(u, v, w)/D_0)^n)$$



(a)



(b)



(c)

図 8 帯域制限補間による形状定義例 (I)

(a) 一葉双曲面, (b) 球, (c) 立方体.

Fig. 8 Examples of the geometric modeling based on band limited interpolation (I).

(a) hyperboloid, (b) sphere, and (c) cube.

ただし、

$D(u, v, w)$: 周波数領域の立方格子の中心から (u, v, w) までの距離,

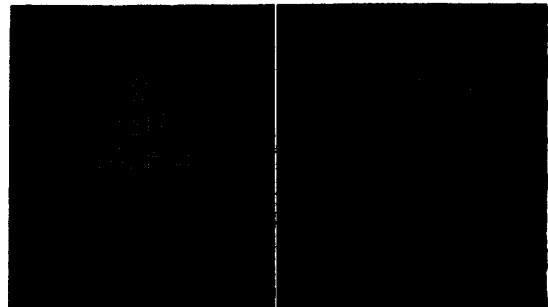
D_0 : カットオフ周波数,

n : 定数,

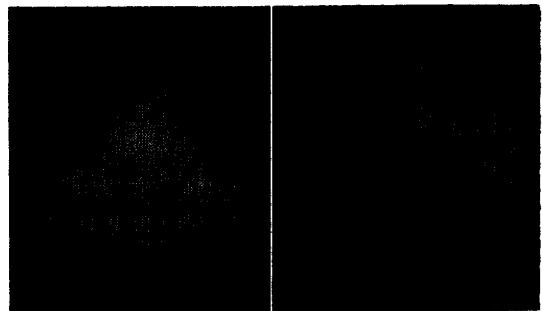
を用いた。法線ベクトルは、離散逆フーリエ変換により計算した。図 8 (a), (b), (c) に、 $R=64, N=8, M=64$ とした場合の、一葉双曲面、球、立方体の例を示す。各画像とも、左のものは光源を正面手前側、右側のもは右上方としたものである。これらの Voxel は 8^3 のサイズであり、これまでの追試のもの



(a) $R=16, N=16, M=64$



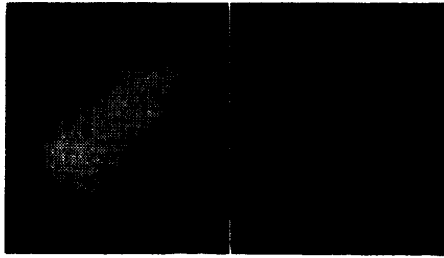
(b) $R=32, N=16, M=64$



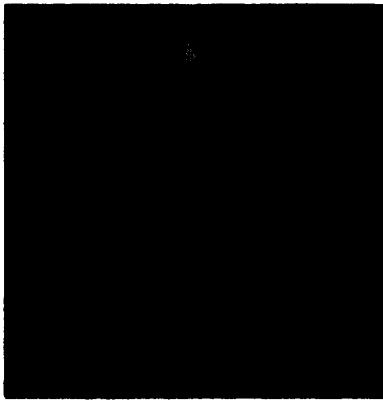
(c) $R=64, N=16, M=64$

図 9 帯域制限補間による形状定義法 (II)

Fig. 9 Examples of the geometric modeling based on band limited interpolation (II).



(a)



(b)

図 10 分布関数法への応用

Fig. 10 Application to modeling of electron density maps.

は 64^3 や 16^3 であることに注意されたい。図 9 には、ローパスフィルタの解像度 R の違いによる物体形状への影響を示すために、 $N=16$, $M=64$ とし、 $R=16, 32, 64$ とした場合の立方体の例を示す。

最後に、ローパスフィルタのカットオフ周波数と等濃度面の決定の仕方により、分布関数法(メタボール)による形状定義法^{11), 12)}と似た効果が得られることを示すための画像例を図 10 に示す。(a)は2つの球の融合を示し、(b)は樹木における枝どうしのなめらかな接合、すなわち、フィレット (fillet) 効果を示している。

以上の計算例より、次のことがまとめとして言えよう。

(1) 2章で示したこれまでのものと比べ、良好な陰影付けが得られており、本論文で示した形状定義法は原理的に有効である。

(2) $R \geq 4N$ とするとローパスフィルタのための標本化によるエイリアスが除去されようである。この点については、さらに、実験を重ねることと、理論面からの検討が必要である。

(3) Voxel の解像度 N に応じてカットオフ周波

数が定まるため、 N が小さい場合は、角が丸みを帯びるなど形状が単調化する。しかし、これは、なめらかな形状を得るための補間法、もしくは、アンタイエイリアシング一般についての問題であり、本方法のみがもつ欠点とは言いがたい。また、今回の計算例の立方体の場合、 $N=16$ であることに注意すると、筆者らは、Voxel の解像度に対して十分な形状が得られていると考えているが、この点については、さらに Voxel の解像度を高めた実験により実証されることが必要である。

5. アルゴリズムの効率化

\tilde{F}_{N^R} を単純に求めるとすると、 R 点の1次元 FFT を縦、横、奥行き方向にそれぞれ R^2 回実行する必要がある。したがって、時間計算量を乗算数で評価し、 R 点 FFT の時間計算量を $R \log R$ とすると(厳密に言うと $(R \log R)/2$ で済むが、評価は相対的な比較であるので、ここでは、 $R \log R$ と評価しておく)、全体で時間計算量は $3R^3 \log R$ となり、 f_R のための3次元配列のために、記憶領域計算量は R^3 となる。さらに、 \tilde{F}_{N^R} から f_{M^R} への帯域制限補間には、 $3M^3 \log M$ の時間計算量と M^3 の記憶領域計算量が必要となり、一般には、 $N \leq M, R$ であるため、単純には、ここで提案した方式の実行は困難であり、時間計算量と記憶領域計算量の面で効率的なアルゴリズムが必要となる。

本章では、

- (1) \tilde{F}_{N^R} を求める効率的な3次元ローパスフィルタリングアルゴリズムと、
- (2) \tilde{F}_{N^R} から f_{M^R} への3次元帯域制限補間および法線ベクトルの決定を行う効率的な基本的アルゴリズムと、
- (3) (2)での法線ベクトルを求めるアルゴリズムよりさらに効率的な改良型アルゴリズムの概要を、各々、5.1, 5.2 および 5.3 節で示す。

文献 13), 14) で与えられている以下のような効率的なアルゴリズム(a), (b)および(c)を用いると、上記アルゴリズム(1), (2)および(3)が構成できる。

(a) 1次元離散フーリエ変換の数値列の部分区間だけを高速に求めるアルゴリズム。

(b) 1次元帯域制限補間を効率的に実行するアルゴリズム。

(c) 部分区間 W の1次元帯域制限補間を効率的

に実行するアルゴリズム.

以下では、アルゴリズム(1)、(2)および(3)の説明に必要となることを記しておく.

アルゴリズム(a)の時間計算量 T_N は、

$$T_N = R(1 + \log N)$$

であることが示されている¹⁴⁾.

N 点の離散信号を F_0 , その離散逆フーリエ変換を f_0 とし, F_0 から付録で示されるように定義される $M(=LN)$ 点の零値付加離散信号を F とすると, その離散逆フーリエ変換 f を求めることが, f_0 の補間を行うことであった. ここで, 後のアルゴリズムの記述のために,

$$\text{IFFT}_N(s, F_0(*))$$

で, 各々の $s(=0, 1, \dots, L-1)$ に対する N 点の離散逆フーリエ変換 $f(s+rL)$, $r=0, 1, \dots, N-1$, を求めるアルゴリズムを表す. $\text{IFFT}_N(s, F_0(*))$ をすべての s に対して実行することにより, M 点の補間が完了するが, これを

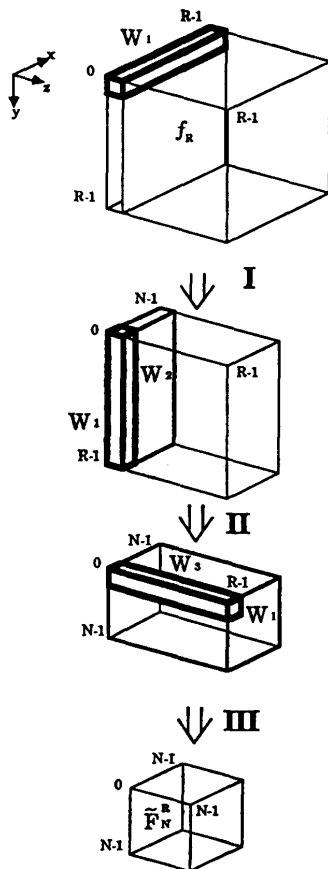


図 11 効率的 3次元ローパスフィルタリング法
Fig. 11 An efficient 3-D lowpass filtering method.

$$\text{IFFT}_M(F_0(*)) (\equiv \text{IFFT}_N(*, F_0(*)))$$

で表す. これがアルゴリズム(b)である, このアルゴリズムの時間計算量 T_A は、

$$T_A = M(1 + \log N)$$

であり (もちろん, 1回の $\text{IFFT}_N(s, F_0(*))$ の時間計算量は, $T_A/L = N(1 + \log N)$ である), さらに, 作業領域での記憶領域計算量 S_A は、

$$S_A = N$$

で済むことが示されている^{13), 14)}.

アルゴリズム(c)の時間計算量 T_w , (作業)記憶領域計算量 S_w は、

$$T_w = M(1 + \log(W/L)) + M - W,$$

$$S_w = N$$

であることが示されている^{13), 14)}. ただし, 部分区間 W は必ずしも連続した点列でなくともよい.

5.1 \tilde{F}_N^R を求める効率的ローパスフィルタリング法

もとの物体が一度に標本化せずとも部分的な区画ごとに標本化可能であれば, アルゴリズム(a)を用いて図 11 に示すように, \tilde{F}_N^R を効率よく求めることができる. そのアルゴリズムを図 12 に示す. 計算 I, II, III の時間計算量は, 各々, $R^2 T_N, R N T_N, N^2 T_N$ であるので, このアルゴリズム全体の時間計算量 T_L は、

```

begin
for z:=0 to R-1 do begin
for y:=0 to R-1 do begin (計算 I)
サイズRの作業用1次元配列  $W_1(*)$  に  $f_x(*, y, z)$  を標本化
によりもとめる;
方法(a)により, 低周波数側のN点の周波数信号を求め,
サイズ  $N \times R$  の作業用2次元配列  $W_z$  の  $W_z(*, y)$  に格納する
end;
for x:=0 to N-1 do begin (計算 II)
 $W_1(*)$  を  $W_z(x, *)$  とし, 方法(a)により, 低周波数側のN点の
周波数信号を求め, サイズ  $N \times N \times R$  の作業用3次元配列  $W_s(x, *, z)$  に格納する
end
end;
for すべての対(x,y)について do begin (計算 III)
 $W_1(*)$  を  $W_s(x, y, *)$  とし, 方法(a)により, 低周波数側のN点の
周波数信号を求め, サイズ  $N \times N \times N$  の Voxel  $\tilde{F}_N^R(x, y, *)$ 
に格納する
end
end.
    
```

図 12 アルゴリズム I: 効率的 3次元ローパスフィルタリング法
Fig. 12 Algorithm I: An efficient 3-D lowpass filtering method.

$T_L = (R^2 + RN + N^2)T_N$
 となる。記憶領域計算量 S_L は、明らかに、
 $S_L = R + RN + RN^2$
 であり、 RN^2 が支配的であり、サイズ N^3 の Voxel の R/N 倍の記憶領域計算量で実行できると言える。計算例から見ると、実用的には R/N は 4 ないし 8 程度の定数と考えられるので、単純に行う R^3 の $1/16$ ないし $1/64$ で済むことになる。

5.2 \tilde{F}_N^R から \tilde{f}_M^R を求める効率的 3 次元帯域制限補間法

以下の説明については図 13 を参照されたい。 N^3 の Voxel \tilde{F}_N^R より補間されて得られる M^3 の空間領域 \tilde{f}_M^R の座標系を (x, y, z) とし、スクリーンは $x-y$ 面に平行にあり、その解像度は M であるとして説明する。さらに、

$$x = q + pL, \quad q = 0, 1, \dots, L-1, \quad p = 0, 1, \dots, N-1$$

$$y = s + rL, \quad s = 0, 1, \dots, L-1, \quad r = 0, 1, \dots, N-1$$

とおく。

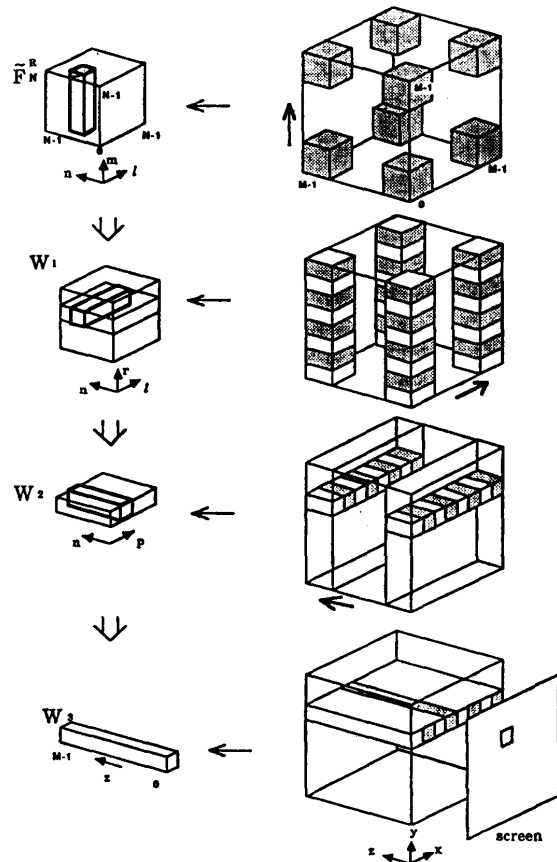


図 13 効率的 3 次元帯域制限補間法
 Fig. 13 An efficient 3-D band limited interpolation method.

図 14 に示すように、アルゴリズムでは、間隔 L の正方形格子点上の N^2 個のスクリーンの画素を単位として、それらに対応する視線との交点および法線ベクトルを求めることを、 M^3 の 3 次元配列を用いることなく効率よく実行している。法線ベクトルの z 、 x 成分の計算を行う部分は、 \tilde{f}_M^R を求める部分と同じ構造となっているので、理解は容易である。アルゴリズムでは、作業記憶領域として、 N^3 の 3 次元配列 W_1 、 N^2 の 2 次元配列 W_2 、 $NL (= M)$ の 1 次元配列 W_3 、求められた交点を格納する $I(x, y)$ 、および、法線ベクトルの各成分を格納する $NX(x, y)$ 、 $NY(x, y)$ 、 $NZ(x, y)$ を用いる。また、

$$\text{IDFT}_N(F(*), z) \equiv (1/N) \left(\sum_{n=0}^{N/2-1} F(n) W_M^{-nz} + \sum_{n=N/2}^{N-1} F(n) W_M^{-(M-N+n)z} \right)$$

で 1 次元離散逆フーリエ変換を表す。ただし、 $W_M = \exp(-jz\pi/M)$ であり $z = \infty$ のときは、 $\text{IDFT}_N(F(*), z) = \infty$ とする。

以下で、このアルゴリズムの計算量を評価する。ただし、ここでの時間計算量は、IFFT と IDFT で実行される乗算数とする。このアルゴリズムは、 \tilde{f}_M^R を求めるための補間に、

$$(1 + L + L^2)N^2 T_A,$$

法線ベクトルの z 成分を求めるのに、

$$L^2 N^3,$$

x 成分を求めるのに

$$LN^2 T_A + L^2 N^3,$$

y 成分を求めるのに、

$$(L + 1)N^2 T_A + L^2 N^3$$

の時間計算量を必要とする。いずれも、 M^3 の 3 次元配列を用いて、 M 点の高速逆フーリエ変換により単純に計算するのと比べると、より高速となっている。記憶領域計算量は明らかに、

$$N^3 + N^2 + LN$$

である。

5.3 法線ベクトルを求める高速アルゴリズム

物体と視線との交点は必ずしもスクリーン全体に存在しないことも多い。このような場合には、5.2 節で示した法線ベクトルの計算法では、 W_1 と W_2 の全体について IFFT と IDFT を実行しているため無駄が多い。本節では、紙面の都合上詳細は割愛するが、アルゴリズム (c) として述べた部分区間帯域制限補間法を用いれば、極力必要な計算のみを行う高速アルゴ


```

begin (交点および法線ベクトルの決定)
  for s:=0 to L-1 do begin
    for 全ての(ℓ,n)について do
      W1(ℓ,* ,n) := IFFTN(s, FN*(ℓ,* ,n)) (計算I);
    for r:=0 to N-1 do begin
      y := s + rL;
      for q:=0 to L-1 do begin
        for n:=0 to N-1 do
          W2(*,n) := IFFTN(q, W1(*,r,n)) (計算II);
        for p:=0 to N-1 do begin
          W3(*) := IFFTN(W2(p,*)) (計算III);
          x := q + pL;
          W3(*)から前述の方法により交点(x,y,z)を求める (交
            点が存在しないときは, z=∞とする.);
          I(x,y) := z;
          (法線ベクトルのz成分の計算)
        end
      end
    end
  end
  (x成分の計算)
end
end
(y成分の計算)
end.

begin (z成分の計算)
  for n:=0 to N-1 do
    if n < N/2 then W2(p,n) := (j 2 π n/NL) W2(p,n)
    else W2(p,n) := (j 2 π (NL-N+n)/NL) W2(p,n);
  NZ(x,y) := IDFTN(W2(p,*), I(x,y))
end.

begin (x成分の計算)
  for q:=0 to L-1 do begin
    for n:=0 to N-1 do begin
      for ℓ:=0 to N-1 do
        if ℓ < N/2 then W1(ℓ,r,n) := (j 2 π ℓ/NL) W1(ℓ,r,n)
        else W1(ℓ,r,n) := (j 2 π (NL-N+ℓ)/NL) W1(ℓ,r,n);
      W2(*,n) := IFFTN(q, W1(*,r,n))
    end
  end
  for r:=0 to N-1 do
    y := s + rL;
    for q:=0 to L-1 do begin
      for n:=0 to N-1 do
        W2(*,n) := IFFTN(q, W1(*,r,n));
      for p:=0 to N-1 do begin
        x := q + pL;
        NY(x,y) := IDFTN(W2(p,*), I(x,y))
      end
    end
  end
end
end
end.
end.

```

図 14 アルゴリズム II : 効率的 3 次元帯域制限補間法
 Fig. 14 Algorithm II : An efficient 3-D band limited interpolation method.

リズムが得られる。概要は以下のようである。

W_1 において、交点の存在する $x-z$ 平面に対応する $n-\ell$ 面の r のリストを $r(i), i=0, 1, \dots, Y_s-1$, また W_2 において、交点の存在する x に対応する p のリストを $p(i), i=0, 1, \dots, X_q-1$, で表す (図 15 参照)。 $Y_s, X_q, r(i)$, および $p(i)$ を定める手続きは前述のアルゴリズムに容易に組み込める。アルゴリズムでは、これらのリスト r, p で示される部分区間についてのみ計算を行う。ここでは、アルゴリズムを明示していないので、証明なしに計算量だけを示しておく。 Y_{\max}, X_{\max} を次のように定める。

$$Y_{\max} = \max_s Y_s, \quad X_{\max} = \max_{s,r,q} X_q.$$

また、 $W=x$ のときのアルゴリズム (c) の時間計算量 T_w を T_x と記すこととする。すると、このアルゴリズムの時間計算量は、 x 成分については、

$$L^2 N^2 T_{X_{\max}} + L^2 N^2 X_{\max},$$

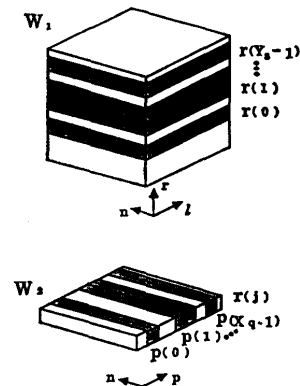


図 15 改良型効率的 3 次元帯域制限補間法
 Fig. 15 A revised efficient 3-D band limited interpolation method.

y 成分については、

$$L^2 N Y_{\max} T_{X_{\max}} + L N^2 T_{Y_{\max}} + L^2 N X_{\max} Y_{\max}$$

となる。ここで、

$$X_{\max}, Y_{\max} \leq N,$$

$$LT_{X_{\max}}, LT_{Y_{\max}} \leq T_A$$

であるので、より高速化が達成されているのがわかる。

ここでは、部分区間帯域制限補間法を用いて、 (r, p) 座標系での部分区間を効率的に求める方法の概要を示したが、あきらかに、 (x, y) 座標系での部分区間を直接求めることも可能であることを断っておく。

6. む す び

本論文では、高品質な陰影画像の生成可能な、帯域制限補間に基づいた、Voxel 表現法による幾何形状の定義法を提案した。これまでに提案されている補間法の追試例と比較し、その有効性を示し、さらに、それに対する効率的なアルゴリズムを与えた。

ここでは、さらに、今後の課題を示しておく。今回は、Voxel の配列の大きさから、分布関数法のシミュレーションを除き、1個の物体のみを取り扱ってきたが、複数の物体を記憶した場合について実験を拡張する必要がある。帯域制限による近接物体どうしの形のゆがみなどの影響について調べる必要がある。

今回は、Voxel には物体の存在の有無の情報だけを格納したが、色や、反射率や屈折率などの物体の属性をもつものに拡張し、高品質なカラー画像を得る方法へと検討を進める必要がある。

CT 画像による医用3次元画像処理への応用の検討も期待される。

さらに、霧や雲のように物体がもともと濃度で表現されるようなぼやけた物体の場合への画像生成法の拡張も重要であると考えられる。

謝辞 昭和62年4月の初投稿以来、長期にわたり数々の貴重なご意見を頂きました査読者の方々に感謝いたします。さらに、論文原稿を丁寧に読んで頂き御討論頂いた東北大学西関隆夫教授に感謝いたします。本研究の初期において多大な協力を頂いた新保賀津雄氏（現在、グラフィカ）に感謝いたします。なお、本研究の一部は、文部省科費費(奨励研究(A)61750313)の補助による。

参 考 文 献

- 1) 安生, 宇佐美, 川島: 積層断面データに基づく3次元物体表示アルゴリズム, グラフィクスとCADシンポジウム論文集, pp. 93-99 (1985).
- 2) Lorensen, W. E. and Cline, H. E.: Marching Cubes: 3D Surface Construction Algorithm, *ACM Comput. Gr.*, Vol. 21, No. 4, pp. 163-

169 (1987).

- 3) 石井: 三次元画像処理的手法を用いたソリッドモデル・システム, グラフィクスとCADシンポジウム論文集, pp. 161-166 (1985).
- 4) Chen L., Herman, G. T., Reynolds, R. A. and Udupa, J. K.: Surface Shading in the Cuberille Environment, *IEEE CG & A*, pp. 33-43 (1985).
- 5) 内木, 所: 3次元メモリを用いた立体図形表示機構—SCOPE—, 電子通信学会論文誌, Vol. J68-D, No. 4, pp. 741-748 (1985).
- 6) Lenz, R., Gudmundsson, B., Lindskog, B. and Danielsson, P. E.: Display of Density Volumes, *IEEE CG & A*, pp. 20-29 (1986).
- 7) Kaufman, A.: Efficient Algorithms for 3D Scan-Conversion of Parametric Curves, Surfaces, and Volumes, *ACM Comput. Gr.*, Vol. 21, No. 4, pp. 171-179 (1987).
- 8) 例えば, Foley, J. D. and Van Dam, A. V.: *Fundamentals of Interactive Computer Graphics*, pp. 580-584, Addison-Wesley (1982) など.
- 9) 高木, 鳥脇, 田村: 画像処理アルゴリズムの最新動向, pp. 84-106, 株式会社新技術コミュニケーションズ (1986).
- 10) Gordon, D. and Reynolds, R. A.: Image Space Shading of 3-Dimensional Objects, *Computer Vision, Graphics, and Image Processing*, Vol. 29, pp. 361-376 (1985).
- 11) Blinn, J. F.: Generalization of Algebraic Surface Drawing, *ACM Trans. Gr.*, Vol. 1, No. 3, pp. 235-256 (1982).
- 12) 西村, 平井, 河合, 河田, 白川, 大村: 分布関数による物体モデリングと画像生成の一手法, 電子通信学会論文誌, Vol. J68-D, No. 4, pp. 718-725 (1985).
- 13) 千葉, 海野, 三浦: FFTによる高速補間法, 電子情報通信学会論文誌, Vol. J71-A, No. 2, pp. 207-212 (1988).
- 14) 青木, 千葉, 斎藤: Voxel 表現による画像生成法について, NICOGRAPH '87 論文集, pp. 200-209 (1987).

付 録 帯 域 制 限 補 間 法

ここでは、Voxel 表現されている離散フーリエ変換 $\tilde{F}_N^R(u, v, w)$ から、帯域制限された物体 $f_M^R(x, y, z)$ の復元、すなわち帯域制限補間法について示す。ここでは説明の簡単さから、1次元信号について述べるが、3次元信号についても同様である。まず、よく知られている標本化定理を示す。

〔標本化定理〕 連続空間における関数 $f(t)$ の周波数帯域が 0 [Hz] から W [Hz] の間に制限されているなら、

$$f(t) = \sum_{K=-\infty}^{\infty} f(k/2W) \frac{\sin \pi(2Wt-k)}{\pi(2Wt-k)}$$

が成立する。

(定理・終)

この式からわかるとおり、空間領域における復元(補間)は、無限回の加算を必要とする。しかしながら、与えられた有限個の標本点が、周期的な帯域制限関数の1周期に対応するものと仮定でき、画像生成で補間値が要求される点 t が有限の精度で表現されるとすれば、次のように、補間が離散逆フーリエ変換により実行できる。ここで、離散フーリエ変換 F 、逆変換 f は次の定義にしたがう。

$$F(n) = \sum_{k=0}^{N-1} f(k) \cdot \exp(-j2\pi nk/N),$$

$$n = 0, 1, \dots, N-1.$$

$$f(k) = (1/N) \sum_{n=0}^{N-1} F(n) \cdot \exp(j2\pi kn/N),$$

$$k = 0, 1, \dots, N-1.$$

標本化間隔 Δt 、標本数 N に対して標本化定理を満足するように帯域制限された連続空間における信号を $\tilde{f}(t)$ とし、標本化間隔 Δt をもつ離散空間における信号を、 $\tilde{f}_N(k) = \tilde{f}(k \cdot \Delta t)$ 、 $k = 0, 1, \dots, N-1$ とする。 $\tilde{f}_N(k)$ の離散フーリエ変換を $\tilde{F}_N(i)$ 、 $i = 0, 1, \dots, N-1$ とし、 $\tilde{F}_M(i)$ を下記に定義される $M (= NL)$ 点の零値付加離散信号とする。

$$\tilde{F}_M(i)$$

$$= \begin{cases} L \cdot \tilde{F}_N(i) & , i = 0, 1, \dots, N/2-1, \\ 0 & , i = N/2, \dots, M-N/2-1, \\ L \cdot \tilde{F}_N(i-M+N) & , i = M-N/2, \dots, M-1. \end{cases}$$

このとき、

$$\tilde{f}_M(k) = \tilde{f}(k \cdot \Delta t/L), \quad k = 0, 1, \dots, M-1$$

は、 \tilde{F}_M の離散逆フーリエ変換で与えられる。これより、標本数 N の離散空間における信号 \tilde{f}_N の離散フーリエ変換 \tilde{F}_N より標本数 M の信号 \tilde{f}_M が補間される。

(昭和63年5月6日受付)
(平成元年12月12日採録)



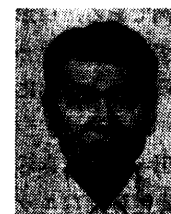
千葉 則茂 (正会員)

昭和26年生。昭和50年岩手大学工学部電気工学科卒業。同年より昭和53年まで(株)日本ビジネスコンサルタントに勤務。昭和59年東北大学大学院博士後期課程修了。工学博士。同年同大学工学部助手。昭和61年仙台電波工業高等専門学校助教授。昭和62年岩手大学工学部助教授。現在に至る。アルゴリズム、コンピュータグラフィクス、形の科学に興味を持つ。共著書「Planar Graphs: Theory and Algorithms」(North-Holland)、「離散数学」(朝倉書店)。電子情報通信学会、ACM、形の科学会各会員。



青木 富雄

昭和38年生。昭和61年東北大学工学部通信工学科卒業。昭和63年同大学院修士課程電気および通信工学専攻修了。同年ソニー(株)に入社。現在に至る。在学中、コンピュータグラフィクスに興味を持つ。



斎藤 伸自 (正会員)

昭和3年生。昭和26年東北大学工学部電気工学科卒業。同大通信工学科教授。工学博士。超短波帯雑音、分布定数回路、帯域制限パルス波形、グラフ理論とその応用、コンピュータグラフィクスなどの研究に従事。著書「回路論」、訳書「グラフ理論入門」など。電子情報通信学会、IEEE 各会員。