

方向性を考慮した3次元テクスチャ法

村岡一信[†]千葉則茂^{††}[†]岩手県立大学盛岡短期大学部^{††}岩手大学工学部情報工学科

あらまし

微細な表面構造を持つ物体表現にはテクセルに代表される3次元テクスチャによる方法が、エイリアシングの発生も少なく有効である。このための3次元テクスチャの各要素にはレンダリング時に必要な情報（反射分布、密度など）が記憶される。これらは方向性を有するが、従来は簡単化のため、反射分布の他は方向性を考慮しないことも多い。本報告では、ポリゴンによる幾何モデルから、方向性を考慮した3次元テクスチャを生成する手法と、そのレンダリング法について提案する。

Anisotropic Three-dimensional Textures

Kazunobu Muraoka[†]Norishige Chiba^{††}[†]Morioka Junior College, Iwate Prefectural University^{††}Department of Computer Science, Faculty of Engineering, Iwate University

Summary

Three-dimensional textures are suitable for generating aliasing-free images of objects having fine surface structure. The well-known "texel" is a typical three-dimensional texture. Each element of the three-dimensional texture includes rendering-parameters, e.g., reflection distribution and object density. Although these parameters are anisotropic, in the previous work, most of them except reflection distribution are often assumed as scalars for the sake of simplicity. In this paper, we present the modeling method for transforming a given polygon-based geometric model into the anisotropic three-dimensional texture, and the rendering method for generating aliasing-free images from the texture.

1. はじめに

コンピュータグラフィックスにおいて、降雪や積雪、布やニット、動物の毛、こけ、遠くから見た場合の草原、田畑、森林などの微細な表現構造を持つ物体のリアルな表現は重要なテーマの1つである。これらを単純に幾何形状でモデリングするとエイリアシングの問題が発生し、柔らかい質感を表現することが困難である。これを解決する方法として、表面構造と反射特性を3次元テクスチャによって表現し、これをボリュームレンダリングと同様な手法でレンダリングする方法が提案されており、エイリアシングの少ない良好な結果が報告されている。

このための3次元テクスチャの各要素にはレンダリング時に必要な情報(光の反射分布、密度、色等)が記憶される。これらは方向性を有するが、従来は簡単化のため、光の反射分布の他は方向性を考慮しないか [Kaji89] [木元92] [木元93] [細川95]、大幅な簡略化が行われている例が多い [村岡95] [Ney95] [Ney98] [Noma95]。3次元テクスチャの1ボクセルに含まれるような微小な物体が、見る方向によって色が変化したり、見かけの大きさ(投影面積)が変化する場合に方向性を考慮しなければうまく表現することはできない。

本報告では、ポリゴンによる幾何モデルから、密度、色の方向性を考慮した3次元テクスチャを生成する手法について、基本型とそれの改良型の2種類を提案する。

2. 従来の3次元テクスチャ法

ここでは、Kajiya らのテクセル(texel) [Kaji89] を例に従来の3次元テクスチャ法について説明し、方向性に関するこれまでの研究について述べる。

2.1 テクセルの要素

テクセル法では、物体は微小面から構成されていると考え、これを3次元の部分領域(ボクセル)に分解して記述する。ボクセルには微小面の集まりそのものではなく、その特徴を表す次の3要素を格納する。

第1要素:微小面の集まりの投影面積あたりの密度

第2要素:微小面の集まりの構造

第3要素:微小面の集まりのシェーディング関数

第1要素は、定義すべき物体を構成する微小面の集まりの投影面積あたりの密度であり、ボクセルの不透明度と微小面の集まりの光の反射強度を表している。投影面積あたりの密度とは、単位空間をある平面に平行投影した場合の、単位空間の投影面積に対する単位空間に含まれる微小面の集まりの投影面積の比率を意味する。

第2要素は、定義すべき物体の表面構造を表現している法線、接線などであり、これらは第3要素を適用するために利用される。

第3要素は、ボクセル中の微小面の集まりの反射特性を表す関数(シェーディング関数)である。

2.2 テクセルのレンダリング

テクセルのレンダリングはボリュームレンダリングとほぼ同じである。すなわち、図1に示すように視線上の各点sの微小面が視線方向へ反射する光を累積する。なお、光源光が各微小面に到達するまで、また、各微小面の反射光が視点に到達するまでの間に、それらの光量はセル中の投影面積当たりの密度(第1要素)により減衰を受ける。よって、視点における光の強さBは次式で表される。

$$B = \sum_{t=t_{\text{near}}}^{t_{\text{far}}} \left\{ \exp(-\gamma \sum_{S=t_{\text{near}}}^t \rho(x_s, y_s, z_s)) I_s(x_t, y_t, z_t) \psi(x_t, y_t, z_t) \rho(x_t, y_t, z_t) \right\}$$

ここで、 t_{near} 、 t_{far} はそれぞれ視線がテクセルへ突入した点と脱出した点である。また、 t, s は視線上の点、 ρ は微小面の投影面積あたりの密度、 λ は微小面の投影面積あたりの密度を光の減衰係数に変換する係数である。

ここで、 I_s は光源からの光の強さであり、 ψ はシェーディング関数(第3要素)である。

I_s は次式で求められる。

$$I_s = I \exp(-\lambda \sum_{r=S}^{L_{in}} \rho(x_r, y_r, z_r))$$

ここで I は、光源の光の強さであり、 L_{in} は光源の光線がテクセルに突入する点である。

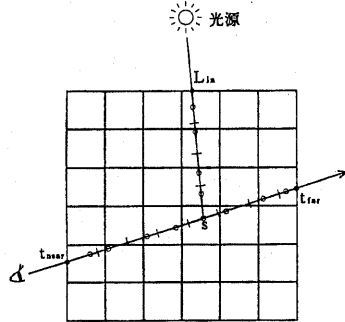


図1 レンダリング法

2. 3 方向性に関するこれまでの研究

幾何モデルで表現された微細な構造を正確に3次元テクスチャで再現するためには、

- (1) 投影面積当たりの密度の方向性、
- (2) 色の方向性、
- (3) 微小面に対する一般的なシェーディング法

が必要である。これまでの報告では、(1)については、Kajiyaら[Kaji89]は密度に方向性が必要であることを指摘したが、方法については示していない。村岡ら[村岡95]は密度を関数で定義する方法を提案したが、幾何モデルから密度を求める方法は示していない。Noma[Noma95]は、幾何モデルによる樹木の葉の集まりについて、固定された複数の方向の密度を求め、任意の方向の密度は、これらの補間により求める方法を提案したが、補間の詳細は不明である。Neyret[Ney95][Ney98]は密度の方向性を楕円体で近似したが、楕円体で表現しうる単純な方向性に限定されている。

(2)についての報告はみあたらない。

(3)については、Noma[Noma95]は樹木の葉の集まりのためのシェーディング関数を示したが、枝やその他の形状に対しても適用可能かどうかは明らかではない。Neyret[Ney95][Ney98]は反射分布を楕円体で近似する方法を提案したが微細構造が複雑化すると誤差が増大すると思われる。

3. 方向性を考慮した3次元テクスチャ法

ここでは、2. 3の(1)~(3)を考慮した3次元テクスチャ法について、基本型とその改良型について述べる。

3. 1 基本型3次元テクスチャ法

基本型は、各ボクセルに仮想的な球体を置き、この球体の表面上の点にレンダリングに必要な情報を記憶させるというものである。

3. 1. 1 方向性のための仮想球体

密度の方向性については、便宜的に密度球と呼ぶ投影面積当たりの密度を記憶するための仮想的な球体を考え、密度球の表面の点 p から密度球の中心(つまりボクセルの中心)方向に見た場合の、すべてのポリゴンの投影面積当たりの密度を加算登録する。

色の方向性と反射特性を記憶するために陰影球と呼ぶ仮想の球体を考え、ポリゴンと同じ法線を持つ陰影球の表面の1点に、ポリゴンの面積の重み付け加算で拡散反射係数と鏡面反射係数およびPhongのモデルの係数を登録する(ポリゴンの反射モデルはPhongのモデルとする)。

3. 1. 2 仮想球体への登録アルゴリズム

ポリゴンで表現した微細な形状を用意し、これから各ボクセルの濃度球、陰影球の表面上に必要な情報を登録する。設定アルゴリズムを次に示す。

濃度球・陰影球の登録アルゴリズム

ステップ0: 微細な形状をポリゴンで作成する。

ステップ1: ポリゴンをボクセルの範囲でクリッピングし、クリッピングされたポリゴン(サブポリゴンと呼ぶことにする)をそのボクセルに登録する。

ステップ2: 各ボクセルごとに密度球、陰影球を用意する。

ステップ3: 各ボクセルごとに、サブポリゴンの法線、面積、表面反射率などから、密度球、陰影球の表面上にそれらの情報を登録する。登録後、サブポリゴンは破棄する。

ステップ3の詳細を次に示す。

- (1) 密度、陰影球の表面上の各点 p をクリアする。

(2) ボクセルに含まれるサブポリゴンのすべてについて、次の(2-1)～(2-3)を行う。

(2-1) ボクセルに含まれる投影面積を1とした相対面積で、各ポリゴンの面積を正規化する。

(2-2) サブポリゴンの法線をN, 相対面積をS, 密度球の表面の点pの法線をNpとすると、密度球上のすべての点pについて、次式で与えられる値Spを加える。

$$S_p = S(N \cdot N_p)$$

ここで、演算・は内積である。ただし、(N・Np)が負の場合はSp=0とする。

(2-3) ポリゴンの法線Nと同じ法線を持つ陰影球上の点pに、そのポリゴンの相対面積と表面反射係数(拡散および鏡面)およびPhongの係数を加える。なお、表面反射係数、Phongの係数はポリゴンの相対面積による重み付け加算とする。

3. 1. 3 レンダリング法

レンダリングは各ボクセルに登録された密度球および陰影球を用いて行う。レンダリングのために必要な計算は次の(1)～(3)である。

(1) 視線上の微小面への光源光が到達するまでの光量の減衰。

(2) 視線上の微小面の視線方向への反射光の強度

(3) 視点方向への反射光が視点に到達するまでの光の減衰。

(1)および(3)には密度球を使用する。すなわちボクセルに登録されている密度球の中心から光源ベクトルの方向、または視線ベクトルと反対の方向へ半直線をのばし、その半直線と密度球の表面との交点pを求める。表面の点pに登録されている密度をそのセルの投影面積あたりの密度とし、光の減衰の計算を行う。

(2)には陰影球を使用する。陰影球の視点から可視の半球の、光源光が当たっている表面すべてをスキャンし、各点pの視点方向への反射光Ipを求め、その総和をボクセル中の微小面の視点方向への反射光の強度とする。各点pの反射光Ipは次式で求める。

$$I_p = I_s(K_d(N_p \cdot (-E)) + K_s(R \cdot (-E)))$$

ただし、Isはボクセルに到達する光源の光の

強さ、Lは光源ベクトル(光源の方向を向くものとする)Npは点pの法線ベクトル、Eは視線ベクトル、Sは点pに登録された微小面の相対面積、Rは光源ベクトルの正反射ベクトル、Ka、Ks、nはそれぞれ点pに登録された微小面の拡散反射、鏡面反射係数およびハイライトの拡がりを表すPhongのモデルの係数である。

3. 1. 4 密度球・陰影球の実現方法

密度球、陰影球は、球体の表面をサンプリングした点の集合として表す。球体の表面上の1点は中心からその点へ向かう方向ベクトル(すなわち法線ベクトル)によって表せる。

3. 1. 5 任意の方向の密度の補間法

任意方向の密度ρは次式によって補間する。

$$\rho = \sum \rho_i (V \cdot N_{pi})^m$$

ここで、Vは光線ベクトル(視線あるいは光源ベクトル)、ρi、Npiは密度球のi番目の点の密度と法線ベクトルである。ただし、(V・Npi)が負の場合は0とみなす。mは点のサンプリングの粗密に関係する定数であり、サンプリングが粗の場合は小、密であれば大とする。

3. 2 改良型3次元テクスチャ法

基本型では、その基本となる仮想球体を、表面をサンプリングした有限な点の集合で表している。この点は多いほど精度は向上するが、それに比例して全体のデータ量も増大する。また、密度球から密度を求める際に補間を行っているが、点pの数が少ないほど、補間結果の信頼性は低くなる。さらに、陰影球にサブポリゴンの情報を登録する場合も、サブポリゴンの法線に一致する点pがなければ、最も近い法線を持つ点pに登録するため、法線の差が大きいほどシェーディングの精度は低下する。

このため、密度計算に補間を必要とせず、シェーディングの精度も低下しない、改良型の3次元テクスチャ法をここで示す。これは、仮想球体の表面上の点を、基本型のように固定的に決定しておくのではなく、個々のボクセルごとに、それに含まれるサブポリゴンの法線と同じ法線を持つ球体表面の点を、仮想球体の点とし

て採用するものである。

3. 2. 1 改良型の仮想球体

改良型で使用する仮想球体は1種類のみである。球体は表面の点の集合で表し、各点には、その点の法線ベクトル、正規化されたサブポリゴンの面積の和、重み付け加算された拡散・鏡面反射、Phongのモデルの係数が記憶される。

3. 2. 2 仮想球体への登録法

仮想球体への登録は、次のように基本型とは同じように行う。

仮想球体への登録アルゴリズム

ステップ0：微細な形状をポリゴンで作成する。

ステップ1：各ボクセルにポリゴンをクリッピングし登録する。

ステップ2：各ボクセルごとに仮想球体を用意し、表面の点pの数を0とする。

ステップ3：各ボクセルに登録されたサブポリゴンについて、その法線に一致する仮想球体の表面の点pが存在すれば、そこにサブポリゴンの情報を登録し、存在しなければ、サブポリゴンの法線と同じ法線を持つ点pを仮想球体に追加し情報を登録する。登録後、サブポリゴンは破棄する。

ステップ3の詳細を次に示す。

(1)ボクセルに含まれる投影面積を1とした相対面積で、各サブポリゴンの面積を正規化する。

(2)サブポリゴンの法線Nと同じ法線を持つ仮想球体上の点pに、サブポリゴンの相対面積を加算し、拡散・鏡面反射係数、Phongの係数を、相対面積で重み付け加算する。

3. 2. 3 密度と輝度の計算

光の減衰を計算するための密度 ρ は、次式で求める。

$$\rho = \sum S_i |V \cdot N_p|$$

ここでVは光線ベクトル、 S_i, N_p はi番目の点pの相対面積と法線ベクトルである。

どのようなVに対しても密度が求められるため、補間は必要としない。

輝度の計算は、陰影球における計算と同様に行うが、シェーディングに必要なサブポリゴン

の正確な法線を使用できるため、基本型より正確な輝度計算が行える。

3. 2. 4 データ量削減の工夫

与えられた幾何形状によっては、特定のボクセルに膨大な点pが必要となる場合も考えられる。このため、方向性に大きく寄与する、相対面積の大きなサブポリゴンのみを登録し、残りのサブポリゴンの情報はスカラ量として1つにまとめてしまうことによりデータを削減する。

輝度計算や密度計算は、このスカラ量も考慮するように変更する。

4. 3次元テクスチャ生成例

ポリゴンにより表現された樹木をレンダリングした結果を図2に、それを $50 \times 50 \times 50$ のサイズの3次元テクスチャに変換し、拡大表示したものを基本型は図3に、改良型は図4に示す。基本型、改良型とも仮想球表面の点数は24である（改良型は相対面積の大きい24のサブポリゴンまでを考慮し、その他はスカラ量とした）。

ポリゴンによる樹木はポリゴン数が約10万枚であり、テキストファイルデータで約10MBである。各ファイルのおおよそのデータサイズは表1のとおりである。

表1 データサイズ

データファイル	データサイズ (MB)
ポリゴン	10
基本型	18
改良型	9

密度計算の違いによる影響と思われるが、ポリゴンによる結果よりも基本型がやや暗く、改良型がやや明るい画像となる傾向が見られた。

データサイズは改良型が基本型の約1/2であり、改良型の方が優れている。また、画像は省略するが、仮想球の点を6まで減少させたところ、改良型では画像に大きな変化は見られなかったが、基本型では葉の陰影が全体的に平坦になる傾向が見られた。改良型の方がデータ削減による画像への影響が小さいと思われる。なお、6点とした場合のデータサイズは両者とも約5MBであった。

5. むすび

ポリゴンによる幾何モデルから、方向性を考慮した3次元テクスチャを生成する手法と、そのレンダリング法を基本型と改良型について提案した。今回は樹木のみ限定された結果ではあるが、改良型の方がデータ数がより少なく、データ数削減による画像への影響も少ない。

今後の課題としては、降雪、積雪などさまざまな微細構造を持つ物体への適用や、さらにデータ量を削減する改良などがあげられる。

なお、本研究の一部は、文部省科研費基盤研究 (C)09680406の補助による。

参考文献

[木元92]木元, 大野, 拡張3次元テクスチャを用いた布地物体の質感表示, 信学論, J75-D-II (4), pp. 706-713, 1992.

[木元93]木元, 大野, テクスチャマッピングによる流水の質感表現, 情報処理学会「グラフィクスとCAD」研究会報告, 61-6, pp. 41-47, 1993.

[細川95]細川, 村岡, 千葉, 3次元テクスチャによる森林景観のレンダリング法, 情報処理学会「グラフィクスとCAD」研究会報告, 74-6, pp. 29-34, 1995.

[村岡95]村岡, 菊池, 千葉, 太田原, テクセルの効率的生成法, 画像電子学会誌, 第24巻, 第3号, 1995.

[Kaji89] J,T,Kajiya, T.L.Key, Rendering Fur with Three Dimensional Textures, Computer Graphics,23,2,pp.271-280,1989.

[Ney95] F.Neyret, A General and Multiscale Model for Volumetric Textures, Graphics Interface'95, pp.83-91,1995.

[Ney98]F.Neyret,Modeling,Animating, and Rendering Complex Scenes Using Volumetric Textures, IEEE Visualization and Computer Graphics,Vol.4,No.1,1998.

[Noma95]T.Noma,Bridging Between Surface Rendering and Volume Rendering for Multi-Reslution Display, Proc. Sixth Eurographics Workshop Rendering, pp.31-40,1995.

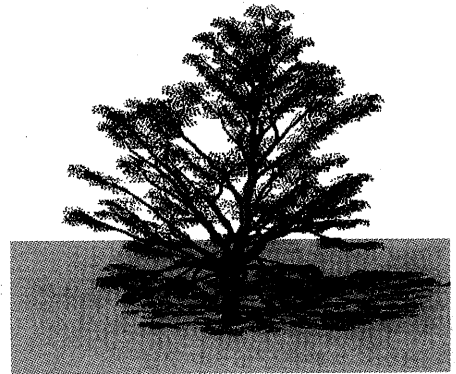


図2 幾何モデルによるレンダリング

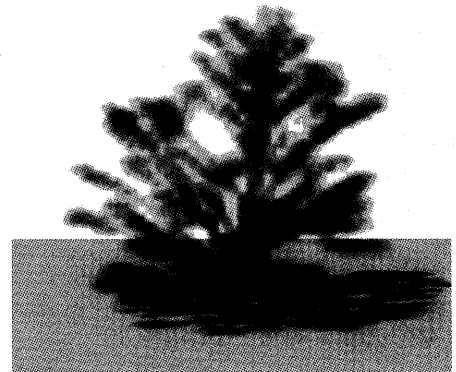


図3 基本型3次元テクスチャ

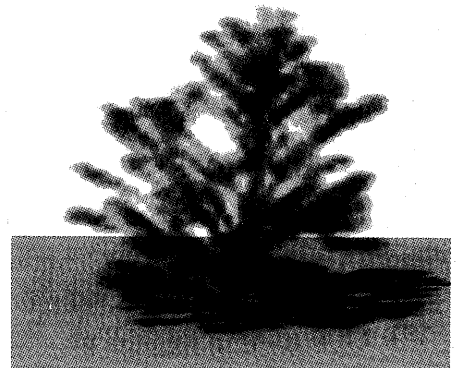


図4 改良型3次元テクスチャ