

3次元自由曲面モデルの高品質ベクトルレンダリング法 High Quality Vector Rendering Method of 3D Free-Form Surface Model

原 武徳*

山本 強*

HARA Takenori YAMAMOTO Tsuyoshi

1 はじめに

ベクトル形式のデータは任意の解像度のラスタ形式のデータに変換できる。そのため、ラスタ形式のデータのように解像度を変更しても絵が粗くならず、必要な解像度に最適な画像として出力することが可能である。つまり、解像度の低い画面で表示する時も、高解像度で印刷する時も同じデータを使用することができる。また、ベクトル画像データではベジエ曲線を利用することができる。ベジエ曲線を用いることで、直感的な操作によって図形を作成したり操作することが可能である。以上のような特徴をもつことから、ベクトル形式のデータはイラストレーションの作成や、高解像度が必要な印刷用のデータとして広く用いられている。

一方、3次元CGは、普通ラスタ画像としてレンダリングされる。しかし前述のように、印刷目的に用いる場合などではベクトル形式の画像のほうがより適しているといえる。3次元モデルをベクトル形式の画像としてレンダリングする、ベクトルレンダリング法を確立することによって、レンダリングする画像の解像度を意識する必要がなくなり、出来上がった画像に必要な大きさに応じて自由に拡大縮小して利用できるようになる。さらに出力された画像データを、ドローツールを用いて修正をしたり、イラストレーションの素材として用いたりすることが簡単にできるようになる。

ベクトルレンダリングの従来法としてはElectric Rain社のベクトルレンダリングエンジンRAViX IIがある[1]。しかし、扱うモデルデータがポリゴンベースであるため、曲面モデルを扱う場合もポリゴンに一度分割してからレンダリングされることになる。曲面モデルの輪郭は実際には曲線になるが、ポリゴン分割したために輪郭は折れ線で表される。しかし、輪郭線を折れ線で表すと線の折れる部分が拡大した時に目立ってしまう、輪郭線を修正する時にベジエ曲線のように大域的な操作ができない、などの欠点があるため、あまり良い方法ではない。輪郭線をベジエ曲線を用いて表すことができれば、これらの問題が解決できると期待される。

そこで本稿では、3次元自由曲面モデルを入力し、ポリゴンモデルを介さずに輪郭線をベジエ曲線を用いたベクトル形式画像として出力する方法について提案する。

2 提案手法

提案手法では、まず自由曲面モデルの法線ベクトルと視線ベクトルとの内積が0になる点列を求め、その点列をベジエ曲線で結ぶことによって輪郭線を得たのち、そして輪郭線を用いて曲面を分割した後、優先順位法を用いて隠面消去を行う。

2.1 輪郭線の決定

ベジエ曲面、B-Spline曲面などの自由曲面は次の様に表される。2つの実数 u, v に対し、点 S を u, v の関数として $S(u, v)$ のように定める。 u, v の値をそれぞれの変域の間で変化させた軌跡として曲面が得られる。この曲面の法線ベクトルは $N(u, v) = \frac{\partial S}{\partial u} \times \frac{\partial S}{\partial v}$ の様に表される。

ある視点 V から見て $S(u, v)$ が輪郭線上にある条件は、視線と面の法線が直交することである。このとき、視線ベクトル $(S - V)$ と法線ベクトル N の内積 $F(u, v)$ は0になる。ゆえに

$$F(u, v) = (S(u, v) - V) \cdot N(u, v) = 0 \quad (1)$$

が成り立つ。

2.1.1 輪郭点列の決定

u, v について一定間隔でサンプルし、各サンプル点での視線ベクトルと法線ベクトルの内積 $F(u, v)$ を求める。 F の値の正負は、面の表側が見えているか、裏側が見えているかに対応している。次に $F(u, v) = 0$ となる u, v 空間上での等値線上の点列をマーチングスクエアズ法によって求める(図1)。まず、 $F < 0$ の格子点(図1の●で表された点)と、 $F \geq 0$ の格子点(図1の○の点)が隣り合っている組を見つける。それらの格子点の間には $F = 0$ となる点、つまり輪郭線上の点に対応する点が存在するので、その点の u, v 座標を線形補間によって求める。全ての格子点を調べていくことで、格子線上にいくつかの $F = 0$ の等値線に対応する点列が得られる。求めた点列を $S(u, v)$ に代入することによって、輪郭線上の点列の3次元座標が得られる。それらの点列を透視変換によってスクリーン座標へ変換する。以上のようにして、曲面の輪郭線上の点列を2次元のスクリーン座標として得ることができる。

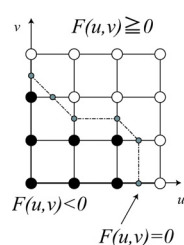


図1 マーチングスクエアズ法

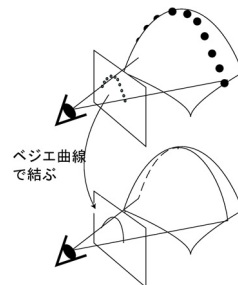


図2 輪郭線の作成

*北海道大学大学院工学研究科

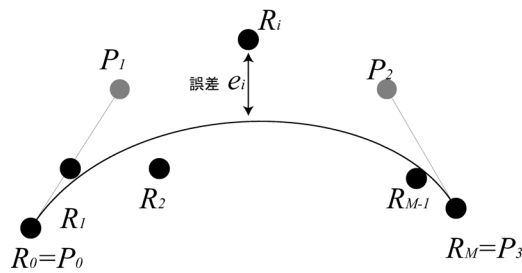


図3 ベジエ曲線による点列の近似

2.1.2 ベジエ曲線による輪郭点列の補間

3次ベジエ曲線 $C(t)$ は4つの制御点 $P_i (i=0,1,2,3)$ とパラメータ $t \in [0,1]$ を用いて次式の様に表示される。

$$C(t) = \sum_{i=0}^3 B_i^3(t) P_i \quad (2)$$

$$\text{ただし, } B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i},$$

$$\binom{n}{i} \text{ は二項係数}$$

この式に $t=0, t=1$ を代入すると、ベジエ曲線は $t=0$ で最初の制御点 P_0 を、 $t=1$ で最後の制御点 P_3 を通過する事がわかる。しかし、制御点 P_1, P_2 は一般に通過しない。

スクリーン座標で表された輪郭点列を通過するベジエ曲線を求めるために、内堀らの手法を用いる [2]。通過すべき輪郭点列を $\{R_i\} (i=0,1,\dots,M)$ とする。輪郭点列の最初の点と最後の点を通るように、 $P_0 = R_0, P_3 = R_M$ とする。そして、残りの制御点 P_1, P_2 を最小2乗法を用いて求める (図3)。輪郭上の各点 R_i に対応するベジエ曲線のパラメータの値 t_i を $t_0 = 0, t_M = 1$ であり、かつ単調増加列となるように、 $t_i = i/M$ と仮定する。点列とベジエ曲線との誤差 E を $C(t_i)$ と R_i の誤差 e_i の2乗の総和と考えて、次式のように定義する。

$$E = \sum_{i=1}^{M-1} e_i^2 = \sum_{i=1}^{M-1} |C(i/M) - R_i|^2 \quad (3)$$

E が最小になる必要条件は、 E を P_1, P_2 の座標値 x_1, y_1, x_2, y_2 で偏微分した値が全て0になることである。このときの x_1, y_1, x_2, y_2 の値を求め、それを制御点の座標として用いる。このときの誤差の値が、指定した許容誤差と比較してより大きい場合は、点列を十分近似できていないと判断し、点列を2つに分割して同じ方法を用いてそれぞれの点列をベジエ曲線で結ぶ。これを誤差が許容誤差以下になるまで繰り返すことにより、輪郭線を近似したベジエ曲線が得られる。

2.2 隠面消去

隠面消去は優先順位法を用いて行う。その前処理として、まず曲面を前節の輪郭線によって切り取ることで、いくつかの曲面に分割する。例えば図4では曲面1と曲面2の2つの曲面に分割されている。これにより、全ての曲面は表の面、または裏の面どちらかのみが見えるようになる。そのため曲面の表の面

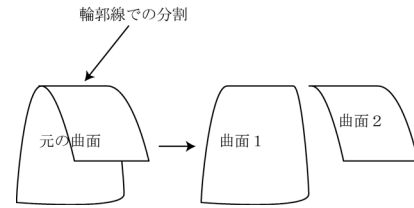


図4 優先順位法

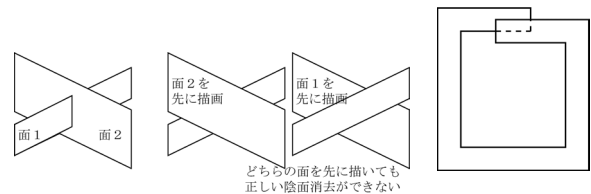


図5 相貫している場合

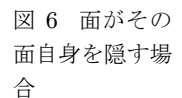


図6 面がその面自身を隠す場合

が自分自身の裏の面を隠してしまうということが無くなる。前方の曲面が後方の曲面によって隠されることが無ければ、視点から見て後方にある曲面から順に描画することで、後から描画される前方の曲面によって陰面は消去されていく。図4では曲面1が曲面2の後方にあるので、曲面1、曲面2の順で描画することになる。

優先順位法の問題点として、描画する物体間の前後関係が決定しないと正しく隠面消去することができないことが挙げられる。そのため、2つの曲面が相貫している場合 (図5)、曲面が自分自身を隠してしまっている場合 (図6) は隠面消去することはできない。これらの場合は曲面をあらかじめ適切に分割する必要がある。

3 まとめ

本稿では、3次元自由曲面モデルをベクトル画像レンダリングする手法を示した。提案手法では、輪郭を自由曲面のパラメータ u と v に対する法線ベクトルと視線ベクトルの内積が0になる等値線ととらえて処理することで、自由曲面をポリゴンに分割するよりも正確な輪郭線を得ることができる。描画する際は輪郭によって曲面を分割することにより、優先順位法を用いて陰面消去ができることを示した。

また問題点として、優先順位法では正しく隠面消去できない場合があり、隠面消去が可能かどうかのチェックする方法を考慮していない。そのため、隠面消去が可能かどうかチェックし、隠面消去が不可能である場合は、隠面消去可能なように曲面を分割するアルゴリズムを考える必要がある。この点は今後の検討課題である。今後は提案手法に基づいた、曲面をベクトルレンダリングするシステムを完成させ、有効性を検証したい。

参考文献

- [1] Electric Rain Inc. <http://www.erain.com>
- [2] 内堀 克則, 苗村 健, 金子 正秀, 原島 博「3次ベジエ曲面チューブを用いた時空間領域画像の記述」テレビジョン学会誌 Vol51. No.10, pp1668-1696, 1997.