7D-5

ポリゴンモデルからの外形線抽出手法

木下　勉†　　今野　晃市‡
†ラティス・テクノロジー（株）　　‡岩手大学工学部

1. はじめに

3 次元モデルの活用方法の一つとして，3 次モデルから 2 次元ベクトルデータを生成し，イラスト化する活用法がある．製造業などで人手によりマニュアル等の挿絵を作成する用途に 3 次元モデルから自動的に生成した 2 次元ベクトルデータを利用することが可能になり，データの再利用や編集ソフトなどでの加工を簡易にすることが可能である．しかし，複雑なデータにおいて，外形線が抽出できなかったり，外形線が一本の閉曲線にならず枝葉ができて，編集ソフト等で線群が閉領域と認識されずにペイントできないといった問題がある．本研究では，3 次元モデルから外形線を枝葉なく一本の閉曲線として取り出す手法を提案する．本手法では，3 次元モデルの位相情報を利用し，容易に閉曲線として外形線を取り出すことが可能である．

2.従来手法の問題点

外形線とは 2 次元空間においての描画領域と非描画領域の境界線と定義する．また，シルエット線とは 3 次元空間において可視な部分と不可視な部分の境界線と定義すると，外形線は必ず物体のシルエット線から生成されることに着目をする．3 次元データから可視な線分をすべて取り出したものを全線分と呼ぶことにすると，全線分のうちシルエット線を 2 次元空間にマッピングした線分の部分集合が外形線ということなる．

一般的に 3 次元モデルから 2 次元ベクトデータを生成する場合に，面の位置関係を考慮した陰線処理手法が重要になる．この有効な手段の一つとして，Z バッファ法 [1] の利用例が挙げられる．Z バッファ内の 2 次元空間は一般に，integer の距離でのみ定義可能な距離空間を用いる．

例えば図1のような場合は要素の特定が困難な場合である．つまり画素の解像度に依存して，微小な物体や，細長い線などに対して，目的となる要素をひとつに特定することはかなり複雑な処理を要することが考えられる．このような問題を解決する方法として，画素の解像度を上げることが考えれるが，メモリの消費量・処理速度とトレードオフになる．

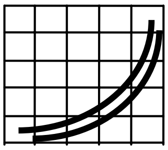

図 1

3.提案手法

本研究では，2 章で述べた問題点を解決した手法を提案する．本手法は，外形線として正しくない要素を取り除き，最終的には閉じたポリラインを生成する手法である．

また本手法を利用する前の段階でポリゴンデータからの Z バッファの構築と，Z バッファの画素に描画をした要素 ID を格納するメモリバッファの生成はすでに済んでいるものとする．

・ ***ステップ1***

シルエット線を正確に求める．シルエット線は 3 次元空間上で定義可能な線ではあるが，本研究の目的である 2 次元ベクトルデータの作成という観点からは 2 次元のスクリーン座標系に変換したものに対してステップ 2 以降の処理を行う．

・ ***ステップ2***

一般に，シルエット線は 3 次元空間で可視な領域と不可視な領域の境界を表す線分なので，シルエット線が 2 次元の描画領域内に描かれることもある．一方外形線は，必ず線分のどちらかの領域が非描画領域となる．この特徴を利用して，線分の両側の面がどちらも可視，またはどちらも不可視な線分は，シルエット線ではないと判断が可能である．また，線分のどちらかの領域が可視であるかの情報も，次のような手順で生成する．

図 2 では可視ポリゴン α の外形線となる線分は，この線の裏側にある不可視ポリゴンとの共有エッジとなっている．このように共有エッジの両側の面の可視・不可視性を利用して外形線に対し，外形線で囲まれた領域の内側方向をエッジに対し設定する．

外形線を示すポリゴンのエッジが共有エッジの場合，すべての頂点が可視となるポリゴン α

「Out Line Extraction by Using Polygon Models」
†「Tsutomu Kinoshita・Lattice Tecnorogy, Japan」
‡「Kouichi Konno・faculty of Enginerring, Iwate University」

の共有エッジの頂点となっていない頂点 A のある向きを，外形線で囲まれた領域の内側方向としてこの共有エッジに設定する．

外形線を示すポリゴンのエッジが非共有エッジの場合，このエッジを有する三角形に対し，上記同様の定義が可能となる．上記の操作を行うことで，すべてのシルエット線に対し，エッジのどちら側に形状が存在するかを得ることができる．

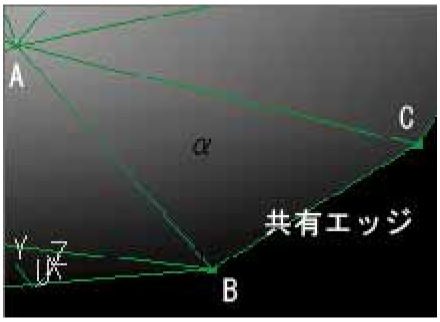

図 2

- *ステップ3*

シルエット線から外形線を特定する．Z バッファに対し，同一のサイズの 2 値(0 or 1)の Z バッファを生成する．形状が描画されている画素を 1，非描画の画素を 0 とする．2 値で表わされた Z バッファから画素に基づいて外形線を生成する．注目する画素とその 8 近傍の画素とを半時計回りで比較し，値が 1 の画素を見つけた場合は，その画素を次の注目する画素として逐次追跡を行い，最初に着目した画素を検出するまで検索を続けることで閉じた領域を生成できる．

またこの画素に基づいたの外形線の 8 近傍に相当するメモリバッファを逐次追跡することにより，外形線となる線分の候補の特定をする．

以上の操作によりシルエット線から外形線となり得るポリライン群が得られる．

- *ステップ4*

外形線同士を連結する頂点の接続についての位置関係を表す構造を構築する．まず 2 次元化したポリラインに対しすべて交線計算を行い，必ず，すべての線分は交点で切断されている状態を構築する．その後以下の情報をそれぞれの頂点，線分に対し構築する．

■ 頂点情報
  - 頂点座標
  - 頂点に接続する線分のインデックス

■ 線分情報
  - 線分の視点側の頂点インデックス
  - 頂点の終点側の頂点インデックス
  - 線分に対して領域の内側を表す方向

- *ステップ5*

外形線の候補となる線が，ある頂点から複数本伸びている場合に，描画領域と非描画領域の境界線となる線を特定する．ステップ 2 で着目した線分の片側が描画領域，もう片側が非描画領域という特徴により意図する線分を特定する．

図 3 のように，ある頂点から伸びている複数本の線分のうち、となりあう線分で定義される領域が，非描画領域の場合に，その線分が外形線を示す 2 本の線分である．このような線分のみを処理の対象として保存する．

例えば図 3 では，a と e が外形線を示す 2 本の線分である．

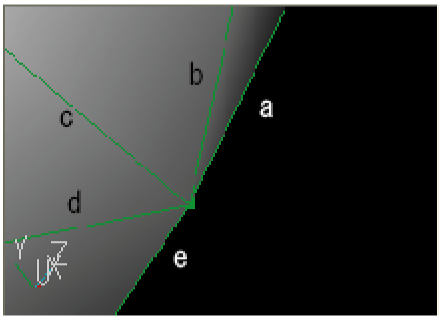

図 3

- *ステップ6*

ステップ 4 で構築した頂点と線分の接続情報と，ステップ 5 で保存した線分の情報から外形線の追跡を行い閉領域を生成する．

4.結果

この処理をある CAD データに適用した結果を図 4 に示す．この図からわかるように外形線が抽出されている．

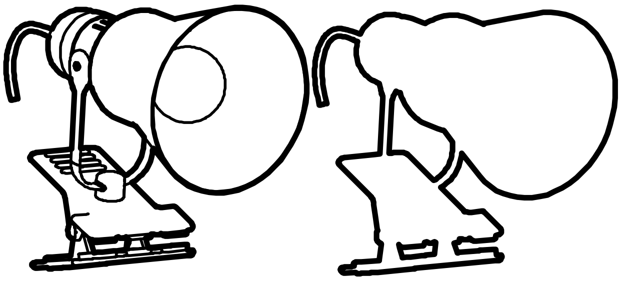

図 4

5.まとめ

本稿ではポリゴンデータを利用し，ポリゴンデータのシルエット線群と画像情報を利用して，外形線を表すポリラインを生成する方法を提案した．今後は生成したポリラインの曲線化，またより複雑な形状に対する処理を実装し，精度を高めていく完全にループを形成していない場合の対処などを考案していく．

[参考文献]

[1] D. Shreiner, M. Woo, J. Neider, and T. Davis, "Opengl Programming Guide: The Official Guide to Learning Opengl”.