

Open Surface メッシュに対する PolyCube Map の構築

佐藤 晋^{†1} 今井 桂子^{†2}

テクスチャマッピングに有用な手法として polycube map がある。polycube map はテクスチャマッピングする 3D モデルと同じトポロジーを持つ立方体の複合体をパラメータ化領域とすることで歪みの少ないマッピングを可能とする。He et al. はこれを自動的に構築するだけでなく、全単射であることを保証した手法を提案した。しかし、He et al. の手法では open surface メッシュに対しては polycube map を構築することができない。本稿では、He et al. の手法において適用外であった open surface メッシュに対して polycube を構築する手法を提案する。

PolyCube Map Construction for Open Surface Mesh

SHIN SATOH^{†1} and KEIKO IMAI^{†2}

Polycube map is a novel cross-surface parameterization technique for texture mapping, where the parametric domain is a polycube (a.k.a. cubical complex). The obtained polycube resembles the geometry of the input shape and shares the same topology. He et al. have proposed a divide-and conquer approach for automatic polycube map construction. However, their approach does not work for constructing a polycube for an open surface mesh. In this paper, we improve their polycube construction technique so that it is applicable open surface meshes.

†1 中央大学大学院 理工学研究科 情報工学専攻
Information and System Engineering Course, Graduate School of Science and Engineering, Chuo University

†2 中央大学 理工学部 情報工学科
Department of Information System and Engineering, Chuo University

1. はじめに

画像の色情報、あるいはその他の 2D の領域を持ったいくつかの情報を用いて 3D の物体の表面にテクスチャをマッピングすることは、コンピュータ・グラフィクスを扱うの多くのアプリケーションで有用な方法である。その品質はパラメータ化の質に極度に依存する。理想的なパラメータ化とは、等角であり面積を保存するものである。しかし、そのような理想的なパラメータ化は、円錐や円柱のような展開可能な曲面にのみ存在することが知られている。したがって、三角形メッシュのための標準的なアプローチは、ひずみが少なくなるように表面をいくつかの円盤状のパッチに切り、それぞれパラメータ化することである。このアプローチは必然的に継ぎ目、すなわち 3 次元的に同じ位置座標を持っているが、2 次元的に異なるテクスチャ座標値を持つような頂点をパッチの境界で複製する必要がある。この継ぎ目は、メッシュ依存性や不適当なフィルタリングを引き起こすことが知られている。継ぎ目を回避する方法は、与えられたメッシュと同じトポロジーを持ち似たような形状を持つパラメータ領域を選ぶことである。

Tarini et al. は、与えられたメッシュと同じトポロジーを持つように設計した polycube と呼ばれる立方体の複合体をパラメータ領域として選ぶことでこの問題を解決した³⁾。しかし 3) において、polycube は手動で構築されているため、アプリケーションへの適用が難しいという問題があった。そこで、He et al. は自動的に polycube map を構築する手法を提案した¹⁾。この手法は自動的に polycube map を構築するだけでなく、3D 表面メッシュとパラメータ領域間の全単射を保証している。しかしながら、1) の手法では、メッシュの表面が欠けていると用いることができないという問題点が存在する。本研究では、1) の手法を、表面が欠けているメッシュに対しても用いることができるように改良することを目的とする。

2. PolyCube Map の自動構築 1)

ここでは、本研究のベースとなる He et al. の polycube map を自動構築するアルゴリズムについて触れる。polycube map とは、与えられたメッシュの形状に似せて作った立方体の複合体の表面をパラメータ領域として使用するマップである。アルゴリズムは以下の 5 つの Step からなる。また、2.2 では特に関わる箇所について詳細に述べる。

2.1 アルゴリズム概要

Step1 位相的な構造の抽出

He et al. のアルゴリズムでは、与えられたメッシュを種数が 0 の形状の集合に分割

する必要がある．そのため，本 Step では与えられたメッシュ M の位相的な構造を調和関数²⁾を用いて抽出する．具体的に抽出する要素は鞍点と呼ばれる極点群であり，メッシュを構成する頂点の部分集合である．また，調和関数値を算出するために極大点と極小点と呼ばれる制御点を定める必要があるが，ここではメッシュ M の z 座標値が最大の点および最小の点をそれぞれ 1 つずつ用いる．

Step2 polycube の構築

scan-line アルゴリズムに類似したアルゴリズムを用いて徐々に polycube を構築する．図 1 に polycube の例を示す．本 Step は 2.2 において詳述する．

Step3 multi-hole ディスクへの埋め込み

与えられたメッシュ M および対応する polycube P を $M = \bigcup M_i, P = \bigcup P_i$ となるような非連結要素 M_i および P_i に分割する．さらに， M_i および P_i に対する uniform flat metric を計算し，これらを multi-hole ディスクと呼ばれる内部に穴が開いた円盤へ埋め込む．

Step4 区分的な写像の計算

multi-hole ディスク間の調和写像を解くことにより， M_i, P_i 間の 1 対 1 のマップを計算する．このとき，2 つの隣接した要素は共通の境界 C_0 を持っており，この境界をまたいで連続であるように写像を計算する．

Step5 polycube map の平滑化

個々の面，辺，角のチャートに対して調和写像を解くことによって，Step4 までに構築した polycube map を平滑化する．

2.2 polycube の構築

ここではパラメータ領域となる Step2 における polycube の構築手法について述べる．polycube の構築アルゴリズムはさらに以下の 3 つの Step に分けることができる．

Step2-1 Contour の検出

Step2-2 Axis-Aligned Boundary Box の作成

Step2-3 polycube の各側面および天面，底面の作成

Contour の検出

Step1 において求めた極点群を z 座標値において昇順でソートし，極小点を v_0 ，鞍点を c_1, c_2, \dots, c_{2g} ，極大点を v_1 とする．但し， g は与えられたメッシュの種数である．この極点群から

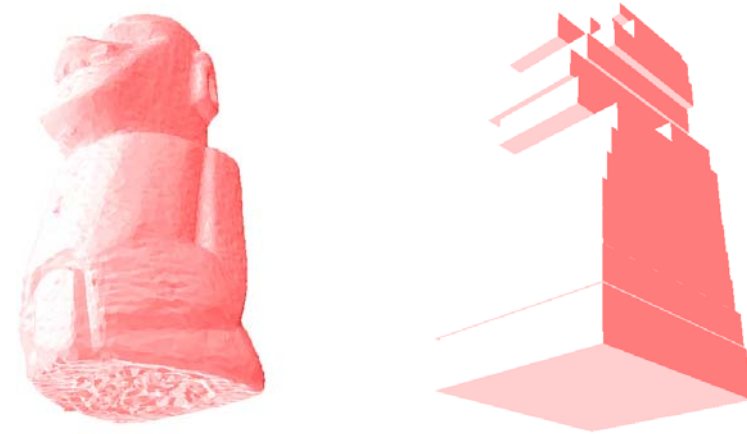


図 1 入力メッシュとその polycube (moai model) .
 Fig.1 Input mesh and its polycube(moai model).

$$\begin{aligned} z_0 &= \frac{z(v_0) + z(c_1)}{2}, \\ z_1 &= \frac{z(c_1) + z(c_2)}{2}, \\ &\dots \\ z_{2g} &= \frac{z(c_{2g}) + z(v_1)}{2} \end{aligned}$$

となるような $2g + 1$ 個の水平切断平面を構築する．但し， $z(p)$ は極点 p の z 座標値である．また，与えられたメッシュの形状が左右対称である場合，2 つ以上の頂点が同じか近い z 座標値を持つことがある．そのようなときはその中のうち 1 つの極点を代表点として選ぶ必要がある．

ここで d_z を 2 つの隣接する切断平面間の最大距離とする． d_z はユーザが指定するパラメータである．この大きさが大きいほど polycube の形状が粗くなり，小さいほど与えられたメッシュの形状に近づく．ある 2 つの連続した切断平面 z_i, z_{i+1} の間の距離が d_z より大きかった場合，その間に均一になるように $\lfloor (z_{i+1} - z_i) / d_z \rfloor$ 個の切断平面を挿入する．このようにすることで 2 つの隣接した極点間に少なくとも 1 つの切断平面があるので，与えら

れたメッシュ M はそれぞれが種数 0 の閉じていない表面メッシュである多くの非連結要素に分割される。このとき挿入した切断平面と極点群から構築した水平切断面とメッシュ M との交差から平面曲線 Contour を検出する。Contour とは、切断平面によってメッシュ M を輪切りにした際にできる M の輪郭である。

Axis-Aligned Boundary Box の作成

Contour を検出できたら、これを囲う Axis-Aligned Boundary Box(AABB) を作成する。AABB とは、Contour を囲うような内角が全て 90 度の多角形である。AABB は四分木を用いて作成する。すなわち Contour を丁度囲うような長方形からスタートし、与えられた閾値 d_a を満足するか、最大分割レベルに達するまで領域分割を行なうことで AABB を作成する。ここで閾値 d_a は、ユーザが入力する Contour c に対する AABB p の近似精度であり、 $d_a > (\text{area}(p) - \text{area}(c))/\text{area}(c)$ であるとき分割を終了する。 d_a の値が大きいほど polycube は粗くなり、また小さいほどメッシュ M に近い形状となる。

polycube の各側面および天面、底面の作成

AABB を得たら、polycube を成す面を作成する。AABB を z 座標値において昇順にソートし、 z 座標値 z_i における AABB の集合を i 番目の集合 Q_i とする。但し、 $i = 1, 2, \dots, n$ である。 z 座標値が小さい順に以下の 2 つの工程を繰り返すことにより最終的な polycube が得られる。

側面の作成

AABB の集合 Q_i における側面は、 Q_i を上下に拡張することで得られる。すなわち、 z 軸に沿って $(z_i + z_{i-1})/2$ ($z(v_0)$ が最初のスキャン平面) から $(z_i + z_{i+1})/2$ ($z(v_1)$ が最後のスキャン平面) へ拡張する。

天面、底面の作成

各側面の間を埋める天面、底面は隣接する 2 つの AABB の集合に対し、必要な CSG 操作を行なうことで作成する。すなわち、 z 座標値が $(z_i - z_{i-1})/2$ の面は $Q_{i-1} - (Q_i \cap Q_{i-1})$ と $Q_i - (Q_i \cap Q_{i-1})$ の和集合で計算される。

2.3 問題点

これまでに述べた 1) の手法は、閉じた表面メッシュにのみ polycube を構築する。そのため、3D スキャナを用いて取り込む際にできる表面の欠損や、元から穴が開いているようなメッシュ (open surface メッシュ) に対して polycube を構築することができない。考えられる原因として Step2 における Contour の検出がある。ここでは Contour を検出するために切断平面を挿入しているが、このとき open surface メッシュにおけるふちの部分

切断平面が通っていた場合、Contour は閉曲線ではなくなってしまい、意図しない AABB, polycube を構築してしまう。本稿では、このような状況においても Contour を正しく検出し、望ましい polycube を構築することを目標とする。

3. 提案手法

open surface メッシュに対して、polycube を構築する手法を 1) をベースに提案する。アイデアは検出される開曲線を補間し、閉曲線である Contour にすることである。ここで検出される開曲線を chain と呼ぶこととする。また、閉曲線に補間する際に継ぎ足した曲線を覚えておくことで穴のあいた polycube を構築することができる。これは元から穴が開いているメッシュに対してテクスチャマッピングを行なう際に有用であると考えられる。

Contour は水平切断面とメッシュ M との交差から検出されるため、ある 1 つの平面上に存在することは明らかである。したがって、chain を補間するに際し、2 次元上の補間アルゴリズムを適用することが可能である。本研究では主にベジェ曲線を用いて補間することを考える。しかし、chain の形状によってベジェ曲線を用いることができない場合が存在する。この場合は特別な処理をする必要がある。

3.1 ベジェ曲線を用いて補間する

chain の始点と終点の頂点を r_0, r_1 としたとき、それぞれを端点に持つ辺 e_0, e_1 を r_0, r_1 の方向に伸長することでできる交点 m_{inter} と r_0, r_1 それぞれとの中点 m_0, m_1 を考える。このとき、 r_0, m_0, m_1, r_1 をベジェ曲線の制御点として用いることで chain を補間する。

3.2 ベジェ曲線を用いることができない場合

辺 e_0, e_1 を伸長した半直線が chain と交差してしまう場合、ベジェ曲線を用いることができない。強引に用いることはできるが、望ましい補間はされないだろう。したがって、別のアルゴリズムを用意する必要がある。

r_0, r_1 を結ぶ線分 e_r を考える。 e_r の中点を m_r とおき、 m_r, r_0 間、 m_r, r_1 間をそれぞれ k 個に分割する。このとき k は 1 以上のユーザ入力のパラメータであり、この値が大きいほど詳細な補間となる。各線分を分割した頂点から端点が r_0 なら e_0 の半直線、 r_1 なら e_1 の半直線へ垂線を下ろす。この垂線を端点 $r_i, i = 0, 1$ に近い方から $l_{ij}, j = 1, 2, \dots, k$ とする。この l_{ij} を $j/(k+1)$ で内分する点を制御点として線分で結ぶことでベジェ曲線を用いることができない場合の chain の補間を行なう。図 2 では、 $k = 2$ のときの形状補間の例を示す。

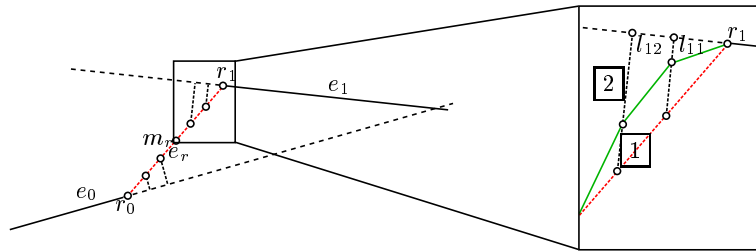


図 2 ベジエ曲線を用いない場合．赤い破線が e_r を，緑の線が補間した線を示す．

Fig.2 Not using Bézier curve case. Segment e_r (red dashedline). Interpolating segments(green line).

4. おわりに

本研究では，1)において polycube を構築できなかった open surface メッシュに対して polycube を構築することができた．しかし，ベジエ曲線を用いることができない chain の形状のうち，chain の両端点が巻きこんでいるような形状では提案したアルゴリズムも用いることができない．このような場合に対しても polycube を構築できるようにすることが今後の課題である．

また，表面の欠損という形で穴が存在する場合，これが塞がるのが望ましい形状であると考えられる．そのため，写像を構築する際に polycube 側で頂点を追加し，穴を塞ぐことで元のメッシュを望ましい形状に変形することが必要であると考えられる．

謝辞 本研究の一部は科学研究費補助金によるものである．

参 考 文 献

- 1) He, Y., Wang, H., Fu, C.-W. and Qin, H.: A divide-and-conquer approach for automatic polycube map construction, *Comput. Graph.*, Vol.33, pp.369-380 (2009).
- 2) Ni, X., Garland, M. and Hart, J.C.: Fair morse functions for extracting the topological structure of a surface mesh, *ACM Trans. Graph.*, Vol.23, pp.613-622 (2004).
- 3) Tarini, M., Hormann, K., Cignoni, P. and Montani, C.: PolyCube-Maps, *ACM Trans. Graph.*, Vol.23, pp.853-860 (2004).