

Isophotic Metric を用いた ペーパークラフトモデルの作製

松本明展^{†1} 今井桂子^{†2}

メッシュからペーパークラフトモデルを作製するためのアルゴリズムを提案する。ペーパークラフトモデルを作製するためには、ペーパークラフトの各パーツが二次元平面に展開できる必要がある。これを可能にするために、各パーツを三角形ストリップに近似する手法を採用する。三角形ストリップを用いてペーパークラフト化を行なう際には、元のメッシュの形状をより保持するために特徴線を用いる。本稿では、既存の特徴線検出手法に加えて、新たな特徴線検出手法を提案し、それを使用する。そして、提案した手法に対して計算機実験を行ない、モデルの形状を保持したペーパークラフトモデルが生成されたことを示す。

Making PaperCraft Models from Meshes Using Isophotic Metric

AKINOBU MATSUMOTO^{†1} and KEIKO IMAI^{†2}

In this paper, we propose an algorithm for making papercraft models from meshes using isophotic metric. To make a papercraft model, we subdivide the mesh into parts corresponding to the original features, and segment each part into zonal regions which consist of continuous triangle strips. Feature lines of the original model are important to keep the shape in the construction of its papercraft model. We introduce a new method of making papercraft patterns using feature lines obtained by isophotic metric.

†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

コンピュータグラフィックスの発展に伴い、多くの 3D モデルが生成されている。これらのモデルはコンピュータ内部にデータを保持しているため、手で直接触れることはできない。しかし、近年これらのモデルからペーパークラフトのパーツを作製することで、手軽にモデルの実体化を可能にする研究が行なわれている^{(6),(7),(9)}。モデルを実体化することは工学的に役立ち、またペーパークラフトは多くの人々が楽しむ趣味のひとつであるため、娯楽としての面白味も持っている。

ペーパークラフトモデルを作製するには以下の要求を満たすようにメッシュを分割する必要がある。

- (1) 各パーツが 2 次元平面に展開可能である。
- (2) 隣接したパーツは境界で互いに接着できる。
- (3) 元のモデルとペーパークラフトモデルの間の誤差が小さい。
- (4) パーツの数が組立可能である程度に少ない。

7) の手法は、メッシュの特徴に基づいてメッシュ分割を行ない、三角形ストリップを構築することで上述の要求を満たすペーパークラフトのパーツを生成している。この手法は、曲面を円錐や平面へ近似するメッシュ分割手法では困難であった、丸みを帯びたメッシュに対するペーパークラフト化を直感的なアルゴリズムで可能にした。この手法ではメッシュ上の辺で隣接した 2 つの面の成す角、つまり二面角で求められる特徴線によって、形状がある程度保持されることを保証しており、ほとんどのモデルに対して有効であることが示されている。しかし、中には人間がモデルの特徴だと認識している部分に特徴線を検出することができずに、でき上がるペーパークラフトモデルの形状が崩れてしまう場合がある。

そこで本研究では、予め Isophotic Metric⁽⁸⁾ を用いてモデルの特徴線を検出し、三角形ストリップを構築する際にモデルの形状が大きく崩れないようにする。この特徴線検出手法は Region Growing の考えに基づいて実行され、既存手法では検出困難であった特徴線も検出できるようになる。そして、この手法で検出された特徴線と、既存手法で用いられる特徴線を併せて使用することで、より形状を保持したペーパークラフトモデルの作製が可能になる。

この論文の構成は以下のようになっている。2 節では Isophotic Metric について説明し、Region Growing を用いた特徴線検出手法を提案する。3 節では、本研究で用いたペーパークラフト化手法の詳細を示す。4 節では、提案手法で得られた結果を図を用いて示す。5 節

では、今後の展望について述べる。

2. Isophptic Metric と特徴線検出

ここでは、8)で紹介されている Isophotic Metric の説明を与える。そして Isophotic Metric を用いた Region Growing を使用して、従来の手法では検出困難だった特徴線を検出する手法を提案する。

2.1 Isophotic Metric の定義

任意の連続した曲面を $\Phi \subset \mathbb{R}^3$ と定義する。 $n(p)$ を点 $p \in \Phi$ における単位法線ベクトルとする。 Φ から単位球面 S^2 へのガウス写像を $\gamma: \Phi \rightarrow S^2$ とし、曲面上の点 p を $n(p) \in S^2$ へ写像することを考える。円盤 $c \in S^2$ の逆写像 γ^{-1} は等輝度線と呼ばれる Φ 上の曲線である。等輝度線上の表面法線は c の回転軸と一定の角度を成している。ここで、曲面上の Isophotic Metric は次のように定義される。

定義: 曲面 Φ 上の曲線 c の Isophotic 距離は、そのガウス写像曲線 $\gamma(c) \subset S^2$ の Euclid 距離である。

この Isophotic Metric は次の性質を持つ。

- 曲面の2点間の最短距離はそれらの曲面法線間の角度 $\alpha \in [0, \pi]$ である。
- 点 $m \in \Phi$ を定める。Isophotic Metric 空間における測地軌道、例えば m から一定の距離 r 上に位置している全ての点 $p \in \Phi$ の集合は等輝度線 c_r である。この等輝度線のガウス写像は中心 $\gamma(m)$ と球の半径 r を持つ円である。
- Isophotic Metric における Φ 上の測線円 g が、ガウス写像として大円の一部を持つことを仮定し、また a_g をこの円の回転軸とする。このとき、 g の各点 p において、曲面法線 $n(p)$ は a_g に対して直角である。また、 a_g の方向への平行射影について考えると、曲線 g は Φ のシルエットの一部になる。

2.2 Region Growing を用いた特徴検出

この節では、メッシュ上で Isophotic 距離を算出し、Region Growing を実行することで特徴線の検出を行なう。距離の算出の仕方は以下の通りである。

- (1) メッシュ上の任意の面 Δ_i の近傍 $N_1(\Delta_i)$ を、 Δ_i と少なくとも一点を共有する三角形の集合と定義する。
- (2) Δ_i と $\Delta_j \in N_1(\Delta_i)$ の法線の成す角を2つの面間の Isophotic 距離とする。
- (3) 近傍 $N_k(\Delta_i)$ は反復処理によって定義される。つまり $N_k(\Delta_i)$ には $N_{k-1}(\Delta_i)$ の境界と少なくとも一点を共有する三角形がすべて加えられる。

- (4) Δ_i からレベル k の近傍 $N_k(\Delta_i)$ に含まれる各面への Isophotic 距離を次のように定義する。まず、 $N_{k-1}(\Delta_i)$ の面の中から $\Delta_l \in N_k(\Delta_i)$ が隣接しており、かつ Δ_i との法線の成す角が最小の面を探す。そして得られた面に既に与えられている Δ_i からの Isophotic 距離にこの2面の法線の成す角を加えた値が Δ_l の Isophotic 距離である。次に Region Growing を行ない、メッシュの表面を複数の領域に分割する。これを行なうために以下の操作を行なう。

- (1) メッシュの各面がどれくらい平坦な場所にあるかを示す *flatness* の値を計算する。これは任意の面 Δ_i からその近傍 $N_1(\Delta_i)$ の各面への Isophotic 距離の和を求め、 $|N_1(\Delta_i)|$ で割った値である。
- (2) Region Growing を実行する。
 - (a) 所属している領域が決定していない面の中で *flatness* が最小の面 Δ_s を探す。
 - (b) Δ_s のみの領域 $R(\Delta_s)$ を作る。 Δ_s とその近傍の面 $\Delta_{next} \in N_1(\Delta_s)$ との Isophotic 距離を算出し、領域への追加条件を満たしていれば Δ_{next} を領域 $R(\Delta_s)$ に加える。
 - (c) 領域 $R(\Delta_s)$ に隣接している面が追加条件を満たしているかどうかを判定して領域に加える操作を繰り返す。もし加えられる面がなくなったら領域 $R(\Delta_s)$ の拡張を停止し、(3)の判定に移る。
- (3) 領域に所属している面の数が、メッシュ全体の面数の $R_{num}\%$ 以下であれば(2)に戻り再び Region Growing を行なう。ここで R_{num} は領域の大きさを保証するパラメータである。

図1のモデルに対して、Region Growing を実際に行なった結果が図2である。各領域はメッシュの特徴的な部分の手前で拡張を停止していることが見て取れる。

この状態から特徴線を求めるために、今度はすべての領域を Isophotic 距離でパラメータ *Dist* だけ一斉に拡張する操作を行なう。これを行なうことでいくつかの領域同士が接するようになる。この領域同士が接した部分の境界を特徴線として扱うことにする。この処理を実行し、特徴線を検出したものが図3である。

この操作により、モデルが滑らかに湾曲して、従来の手法では検出することが困難であった特徴線を検出することが可能になった。

3. ペーパークラフトモデルの作製

この章では、前章で検出した特徴線と、7)で紹介されている手法を用いてメッシュをペー

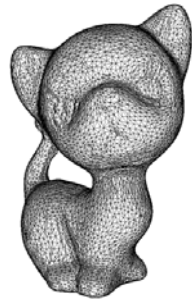


図 1 kitten モデル
Fig.1 kitten model

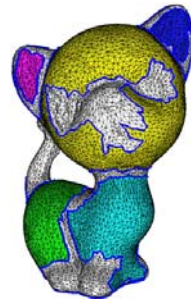


図 2 Region Growing の結果
Fig.2 Result of region growing

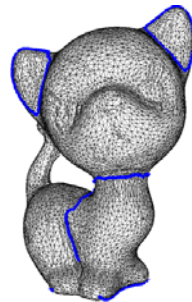


図 3 Isophotic 特徴線
Fig.3 Isophotic feature lines

パークラフト化する手法を紹介する。3.1 では、5) で紹介されている二面角に基づいた特徴線検出手法と特徴線に基づいた領域分割手法の紹介をする。3.2 では、3.1 節で作られた領域をさらに帯状の領域に分割する。これは三角形ストリップをより簡単に、美しく生成するために行なう操作である。3.3 では、元のメッシュの形状がペーパークラフト化したときにも保持されるように、いくつかの領域にカットラインを付加する。3.4 では、帯状領域やカットラインを滑らかにする操作を行ない、最後に 3.5 で各帯状領域を三角形ストリップに近似する。

3.1 二面角に基づく特徴線検出とメッシュ分割

この説ではモデルの特徴に基づいてメッシュを複数の領域に分解する。メッシュ分割に関する手法は数多く研究されているが、ここでは Levy らの手法⁵⁾を採用する。この手法はテ

クスチャマッピングに対して提案されているが、ペーパークラフトモデルの作製にも使用できる。

この手法では最初に、二面角の大きさに基づいて特徴線を検出する。ここで求められる特徴線は、既に検出された特徴線から一定以上離れた位置で検出を行なう操作であり、他の特徴線と交差したり、既存の特徴線に過度に近い位置では検出されない。前章の操作も併せて、特徴線検出操作で得られるすべての特徴線を図 5(a) に示す。

次にこれまでで求められた特徴線に基づいてメッシュを複数の領域に分割する。詳細は 5) で示されている通りであるが、概要としては特徴線上で領域が互いに接するように、複数の領域を一斉に拡張させるアルゴリズムである。

このメッシュ分割手法で得られた領域が十分な大きさになることを保証するために、次の領域を結合する操作を行なう。

- (1) 三角形の数が元も少ない領域 R_{min} を選択する。もし R_{min} が事前に定義された三角形の数 C_{num} よりも多くの三角形を持っていたらこの操作を終了する。そうでない場合は (2) の操作に移る。
- (2) R_{min} に隣接する各領域 R_{adj} に対して、 R_{adj} と R_{min} の間の境界上にある辺の数を数える。
- (3) (2) で数えた数が最大であった領域と R_{min} を結合し、(1) に戻る。

以上の操作を行なうことで、メッシュを複数の領域に分割できる。また、前節とここで検出された特徴線のほとんどが領域の境界上に置かれるようになる。しかし、すべての特徴線が境界上に置かれるわけではない。そのため、これらの特徴線をカットラインとして加えることで、ペーパークラフトモデルの形状を保持する。このとき、このカットラインを特徴カットラインと呼ぶ。

3.2 帯状領域の生成

3.1 で生成された各領域を三角形ストリップの集合に近似しやすくするために、領域を帯状の領域に分割する。各領域から帯状領域を作製するために以下の操作を行なう。

- (1) 各面に対して、最も近い境界、もしくは特徴線からの位相距離の値を割り当てる。
- (2) 三角形に割り当てられた値に従って領域を分解する。各領域の境界は nw と $nw + 1$ ($n = 1$ と w は正整数) の値を持つ三角形の間の辺に沿って与えられる。

この操作によって、最も内部の領域以外は w の幅を持つ帯状領域に分割される。帯状領域の幅である w は近似の正確さに影響を与える。たとえば、より小さな値を持つ w を用いると、その分正確な領域を生成してくれるが、その分最終的にペーパークラフトを生成する

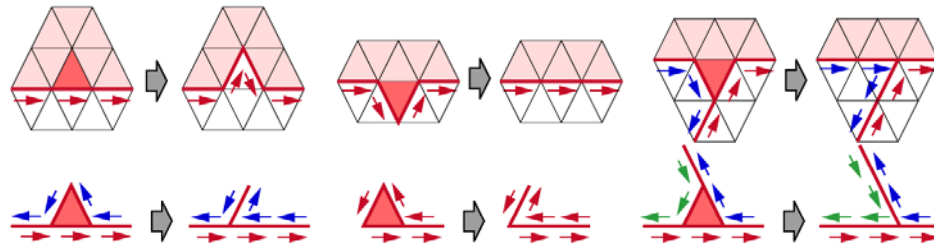


図4 外側ループの更新規則
 Fig.4 Rules for updating outer loops.

のが困難になる。もし内部領域の大きさが Z_{num} に満たない場合、3.1 で小さい領域に適用した領域結合アルゴリズムを実行する。

帯状領域のボーダーは最終的にカットラインとなる。このとき、これらをボーダーカットラインと呼ぶ。

3.3 カットラインの追加

この手法では、3.4 でカットラインを保持したまま各帯状領域を三角形ストリップに近似する。しかし、特徴カットライン、ボーダーカットラインだけでは、近似をする際に領域の内部の特徴が消えてしまう可能性がある。これを避けるために、特徴を保持するための追加のカットラインを領域に追加する。

この特徴保持のための操作は、ディスクと位相同型である領域に対して実行する。次のように領域の外側を一周するループを規則に従って縮約していくことで、領域の中心部にカットラインを生成する。

- (1) 領域内のすべての三角形をリスト T に加えて、さらに領域の外側を一周する辺のリスト L を作る。
- (2) T から三角形を削除することにより L を更新していく。削除される三角形は領域の境界に最も近いものであり、 L を更新する方法は三角形がどのように L に接しているかによって異なる(図4を参照)。赤い太線は外側ループ L の辺で、矢印はループの方向を示している。おなじ色の矢印は L 内で隣接した辺を表している。 T 内の三角形はピンクで示され、削除される三角形は赤で示されている。
- (3) T が空になるまでステップ2を繰り返す。

この操作によって、点と辺で作られる木が領域内に作られる。しかし、この木はそのままカットラインにするには複雑で大きすぎる。そのため、以下の操作を行なうことで、適当な

大きさのカットラインを生成する。

- (1) 前述のアルゴリズムを実行して木を作成する。
 - (a) 木の葉の点のリスト L' を作成する。
 - (b) L' 内の点を木から削除していく。木の中の点の数が事前定義された数 T_{num} まで減らされたらステップ2に進む。もし L' が空になったら、残りの木から L' を再構築し(a)から繰り返す。
- (2) もし領域の境界に接している点が存在したら、木からこれらの点を削除する。この結果で得られる木をカットラインとして加え、センターカットラインと呼ぶ。

3.4 カットラインの平滑化

紙からパーツを切り離して、組み立てる作業を容易にするために、これまでに生成されたすべてのカットラインを平滑化する、これを行うために、次に説明する接続性の平滑化と幾何学的な平滑化の両方を適用する。

3.4.1 接続性の平滑化

接続性の平滑化は同一カットライン上に2つの辺をもつ三角形が存在したら、三角形の3本目の辺にカットラインを置き換える操作である。この置換操作は、カットラインが分岐している部分に隣接している三角形には適用しない。

3.4.2 幾何学的な平滑化

カットラインを滑らかにするために、接続性の平滑化を適用した後、次に示す一次元ラプラス操作を用いることで同一カットライン上に二つの隣接点を持つ点を動かす。

$$p[i] = \frac{1}{4}p[i-1] + \frac{1}{2}p[i] + \frac{1}{4}p[i+1] \quad (1)$$

$p[i]$ は動かされる点の座標である。そして $p[i-1]$, $p[i+1]$ はカットライン上で $p[i]$ に接続している隣接点の座標である。この操作を繰り返すと、カットラインはより滑らかになるが縮小してしまう。そのため、この操作を2回だけ適用し、ラインを過度に縮小させることなく滑らかにする。

3.5 三角形ストリップの生成

ここでは、前節までに得られたカットライン上にある点以外のすべての点を削除するというメッシュ単純化手法を用いて三角形ストリップを生成する。Cohen は2) でメッシュ単純化の様々な手法をまとめているが、ここではその中でも Garland と Heckbert によって提案された辺崩壊と呼ばれるメッシュ単純化手法³⁾を用いる。この辺崩壊は適用できる限り繰り返し実行される。そのうち、任意のカットライン上にない点が残っていたら、点削除操

作¹⁰⁾を用いる。この単純化の結果として、カットライン上にのみ点を持つメッシュが得られる。単純化ののち、三角形ストリップをより滑らかなものにするため、対角変形操作⁴⁾を用いる。この対角変形操作は、カットライン上にない辺で、さらに辺に隣接した2つの面の法線の成す角が小さくなる時にのみ適用する。

以上の操作により、三角形ストリップをパーツとして持つ単純化されたペーパークラフトモデルを作成することができた。これらのパーツを二次元平面にパッキングすることで、ペーパークラフトを組み立てることが可能になる。三角形ストリップのパッキングは、平面にストリップ内の三角形の一つを置き、既に平面内にある三角形に接続している三角形を再帰的に平面に加えていくことで可能になる。もしひとつのストリップが自己交差したときは、展開するストリップを二つに分割する。オブジェクトを平面にパッキングする問題は様々な研究が行なわれているが、ここではこの問題については議論せず、パーツを手動でパッキングしている。

4. 実行結果

本稿では、既存手法⁷⁾では特徴検出のしにくかった kitten モデル (図1) に対して提案手法を適用した結果を示していく。kitten モデルは11039個の頂点と22078個の三角形の面を持つ。

2節で新たな特徴検出手法を提案したが、この特徴検出を行なうに当たって用いたパラメータや、領域に面を追加する条件は次の通りである。

まず、領域に面を追加する条件は次の通りである。判定を行なう面 Δ_i が Δ_s のレベル k の近傍である、つまり $N_k(\Delta_s)$ に所属する面であると仮定する。また、メッシュ全体の隣接する2つの面の法線の成す角の平均値を α_{mean} とする。このとき、もし $N_{k-3}(\Delta_s)$ の境界から Δ_i に到達するのに必要な Isophotic 距離が、 $2.5 \times \alpha_{mean}$ よりも小さかったら Δ_i を領域 $R(\Delta_{start})$ に追加する。次に各パラメータは次のように設定した。Region Growing を引き続き行なうか判定するためのパラメータ R_{num} は80とする。また、複数の領域を一斉に拡張するときの Isophotic 距離 $Dist$ には $\pi/4$ を用いた。

次に既存手法で用いているパラメータ C_{num} , w , Z_{num} , T_{num} はそれぞれ次の値を用いた。5) の手法を用いて用いられる領域の大きさを決めるパラメータ C_{num} はメッシュ全体の面の数の3%とした。帯状領域の太さを決定するパラメータ w は7とした。帯状領域の最も内部にある領域の大きさを保証するパラメータ Z_{num} はメッシュ全体の面の数の0.5%とした。また、帯状領域に加えらるセンターカットラインの大きさを決めるパラメータ

T_{num} は、最初に求められた木に含まれる点の数の30%とした。図5は各ステップにおける実行結果を示したものである。

また、図6は既存手法を用いてペーパークラフトモデルを作成する際に、各ステップで得られる結果である。

図5と図6を比較してみると、提案手法では特徴検出においてモデルの耳や首の部分によりしっかりと特徴線が得られていることが分かる。また、最終的に得られる結果も検出できた部分においてしっかりと形状が保持できていることが分かる。ここで、定量的に2つの結果を比較するために、Metro¹⁾を用いて元モデルとペーパークラフトモデルの形状の比較を行なった。まず既存手法で得られたペーパークラフトモデルを元モデルと比較したところ、RMSの値は0.005771であった。これに対し、提案手法で得られたペーパークラフトモデルと元モデルを比較したところ、RMSの0.004673であった。RMSの値は小さい方が元のモデルに近いと言えるため、数値的に見ても提案した手法がより形状を保持できていることが分かる。

5. 結論と今後の課題

本研究では、Isophotic Metric を用いて特徴線の検出を行なうアルゴリズムを提案し、そして7)で提案されているメッシュのペーパークラフト化手法を用いて、より形状を保持したペーパークラフトモデル作製手法の提案を行なった。特徴検出においては、二面角の大きさからだけでは検出することが困難であった特徴線を検出することに成功した。また、これらの特徴線はペーパークラフト化する際の形状保持の観点から、有効であることを示した。

今後の課題としては、ペーパークラフトの各パーツの形状が複雑で、さらに直感的な組立が困難であるため、モデルの組立に時間がかかってしまうことが挙げられる。また、数多くのパラメータを使用していることから、これらのパラメータの適切な値を自動で求めるアルゴリズムを構築することも課題として挙げられるだろう。

参考文献

- 1) Cignoni, P., Rocchini, C. and Scopigno, R.: Metro: measuring error on simplified surfaces, Technical report (1996).
- 2) Cohen, J.D.: Concepts and Algorithms for Polygonal Simplification, ACM SIGGRAPH Course Tutorial (1999).
- 3) Garland, M. and Heckbert, P.S.: Surface simplification using quadric error metrics, *Proceedings of the 24th annual conference on Computer graphics and interactive*

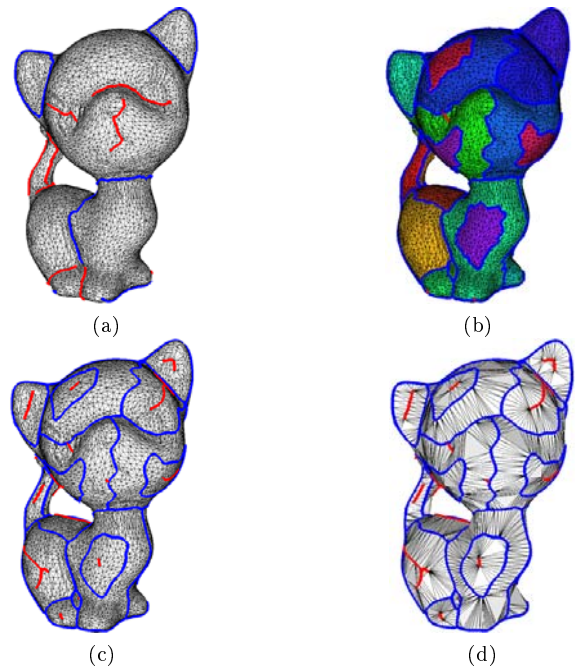


図 5 提案手法の出力結果
 (a) 特徴線 (b) 帯状領域 (c) カットライン (d) 三角形ストリップ
 Fig.5 Result of proposed method.
 (a) Feature lines (b) Zonal regions (c) Cutlines (d) Triangle strips

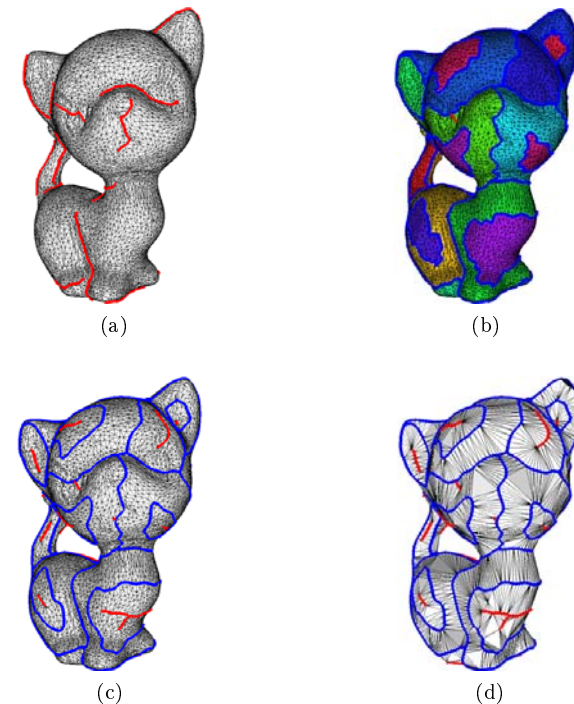


図 6 既存手法 7) の出力結果
 (a) 特徴線 (b) 帯状領域 (c) カットライン (d) 三角形ストリップ
 Fig.6 Result of existing method 7).
 (a) Feature lines (b) Zonal regions (c) Cutlines (d) Triangle strips

techniques, ACM SIGGRAPH, pp.209–216 (1997).

- 4) Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W.: Mesh optimization, *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, ACM SIGGRAPH, pp.19–26 (1993).
- 5) Levy, B., Petitjean, S., Ray, N. and Maillot, J.: Least squares conformal maps for automatic texture atlas generation, *ACM SIGGRAPH*, pp.362–371 (2002).
- 6) Massarwi, F., Gotsman, C. and Elber, G.: Papercraft Models using Generalized Cylinders, *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pp.148–157 (2007).
- 7) Mitani, J. and Suzuki, H.: Making papercraft toys from meshes using strip-based approximate unfolding, *ACM SIGGRAPH* (2004).

- 8) Pottmann, H., Steiner, T., Hofer, M., Haider, C. and Hanbury, A.: The isophotic metric and its application to feature sensitive morphology on surfaces, *Computer Vision — ECCV 2004, Part IV* (Pajdla, T. and Matas, J., eds.), Lecture Notes in Computer Science, Vol.3024, pp.560–572 (2004).
- 9) Shatz, I., Tal, A. and Leifman, G.: Paper craft models from meshes, *Vis. Comput.*, Vol.22, pp.825–834 (2006).
- 10) Turk, G.: Re-tiling polygonal surfaces, *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, ACM SIGGRAPH, pp.55–64 (1992).