

3次元モデルからのあみぐるみ生成

Knitting a 3D Model

五十嵐 悠紀[†] 五十嵐 健夫[‡] 鈴木 宏正^{*}

Yuki IGARASHI[†] Takeo IGARASHI[‡] and Hiromasa SUZUKI^{*}

[†]*東京大学大学院工学系研究科 [†]*Graduate School of Engineering, The University of Tokyo

[‡]東京大学大学院情報理工学系研究科 / JST ERATO

[‡] Graduate School of Information Science and Technology, The University of Tokyo / JST ERATO

E-mail: [†] yukim@acm.org, [‡] takeo@acm.org, ^{*} suzuki@den.rcast.u-tokyo.ac.jp



(a) 3次元サーフェスモデル (b) 3次元あみぐるみモデル (c)自動生成された編み図 (d) 実際に編んだあみぐるみ
図1:本システムは3次元モデルをあみぐるみにするための編み図を生成する.自動生成した編み図を元にシミュレーションを適用することで,あみぐるみ完成予想モデルをユーザに提示する.ユーザは生成された編み図を編むことで実世界にあみぐるみを作成できる.

概要

あみぐるみ(図1(d))は閉じた面の内部に綿を詰めることで作られている.本論文では,3次元サーフェスモデルからあみぐるみを作成するための編み図を生成するシステムを紹介する.編み図はどのように編んでいくかを記号化したものであるが,編み図をデザインすることは素人には難しく,スキルをもった専門家がデザインしている.本システムではその過程を自動化し,誰でも簡単に3次元モデルから編み図を生成することができる.本システムは3次元モデルへ等幅のストリップを巻きつけ,得られたストリップを等幅でサンプリングすることで,編み図へと変換する.編み図は通常用いられているフォーマットで出力され,誰でも簡単に編み図からあみぐるみを作成できる.本システムを利用して作成したあみぐるみを紹介する.

1. はじめに

コンピュータグラフィックス(CG)は物理的なオブジェクトをデザインするためのツールとしても発展してきた.伝統的なCAD(Computer Aided Design)システムは主に専門化のユーザに使われているが,非専門化のユーザにとって使いやすいCGや,インタラクション技術を盛り込んだシステムの研究が近年盛んであり,ペーパークラフトをデザインするツール[1, 2]や,ぬいぐるみをデザインするツール[3, 4]などが挙げられる.本論文では,同様のアプローチをあみぐるみのデザイ

ンに適用したシステムを紹介する.

毛糸とかぎ針を使ってぬいぐるみを作る“あみぐるみ”は日本に存在する重要な文化である.あみぐるみは毛糸を筒状に編んでいき,綿を詰めることでできあがる(図1(c), (d)).あみぐるみは図2のように胴体,頭,腕,脚など,いくつかの筒状のものや円と同相の部品から成り立っている.それぞれのパーツは,1本の毛糸で編まれており,渦巻きの形状は,複数段の編み図で表現される(図1(c)).それぞれの段は前後の段と編み目によってつながれており,基本的に四角形のセルの形状になっている.部分的に1つの編み目が次段の2つの編み目とつながっている「増やし目」や,2つの編み目が次段の1つの編み目とつながっている「減らし目」がある(図3, 4).



図2: あみぐるみの部品.筒状のものや,円と同相のものから成る.

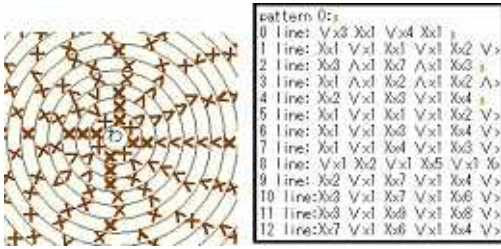


図 3：編み図の例．左図は円状の編み図で，右図はテキスト出力．X は普通目，V は増やし目， は減らし目を表す．

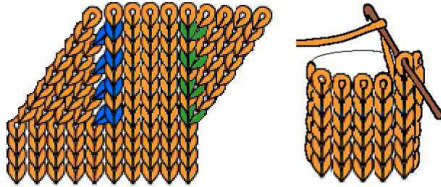


図 4：編み目の構造．緑色が増やし目，青色が減らし目．その他が普通目．

編み図は，欲しい形状をどのように編んだら良いかを示すものである．図 3 のように円状，もしくはテキスト出力で編み目の記号を用いてデザインされる．円状の編み図は内側から外側に向けて，1 段目，2 段目，...となっており，らせん状に記号に沿って編んでいく．できあがりの形状を想像しながらこの編み図をデザインすることは素人には難しく，編み物に長けた人が試行錯誤によって作成した編み図を利用していることがほとんどである．本システムはこの工程を助けるものである．

本システムは 3 次元サーフェスモデルを入力として，あみぐるみを作成するための編み図を生成する．システムはまず，いくつかの閉じた部品に領域分割し，それぞれの領域に対して等幅のストリップで包むために，等間隔で複数の等値線を計算する．次に，等間隔で等値線をリサンプリングして，メッシングすることで編み目を生成する．最後に，生成された編み目を渦巻き状に並べることで標準フォーマットの編み図を生成する．ユーザはこの編み図をもとに実際に編むことであみぐるみを作成することができる．我々のシステムは入力モデルに物理シミュレーションを適用することであみぐるみの完成予想モデルもユーザに提示する．

2. 関連研究

近年，3 次元 CG をコンピュータの中で扱うだけでなく，手にとって楽しめる作品にするための研究が盛んである．ペーパークラフトのための展開図を作成する研究[1][2]や，ぬいぐるみのための型紙を作成する研究[3][5]が行われている．Plushie [4]は 3 次元モデリン

グと物理シミュレーションを組み合わせることでオリジナルなぬいぐるみをデザインするシステムである．これらのシステムは 3 次元サーフェスモデルを 2 次元へと展開するものである．本論文では，あみぐるみ作成のために，3 次元サーフェスモデルを 1 次元の小さなセル列として表現することを行った．Knitty [6] は対話的にオリジナルなあみぐるみをデザインするツールである．ユーザからのスケッチ入力を元に，システムは 3 次元あみぐるみモデルと編み図を構築する．しかし，既存の 3 次元モデルから編み図を生成することはできない．

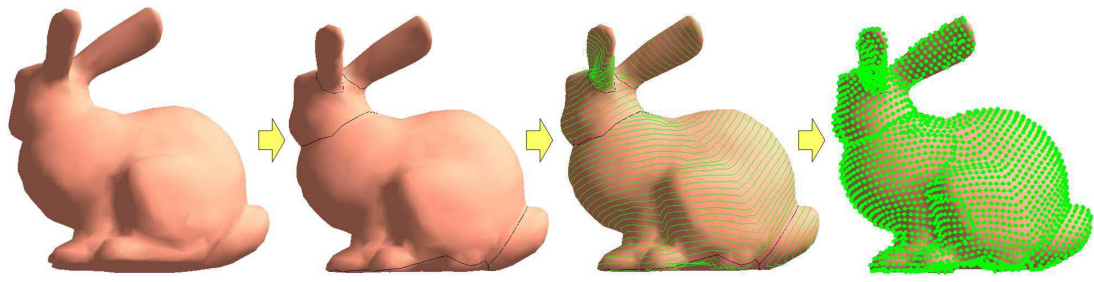
3 次元モデルから 1 次元セル列への変換というコンセプトは，既存アルゴリズムでは我々のニーズにあうものは存在しない．3 次元サーフェスモデルから三角形ストリップを生成するものはいくつか提案されている[7, 8]が，これらは等幅のストリップを生成することはできない．本システムは四角形メッシュと少量の三角形メッシュを生成する．我々は 1 本のストリップが必要であるが，既存の 3 次元サーフェスモデルから四角形メッシュを生成する手法[9, 10, 11] は 1 本の等幅のストリップを生成することはできない．Sweep-based モデル[12, 13] や，generated cylinder [14] は我々のアプローチに似たものであるが，これらから等幅のストリップは生成できない．

3. アルゴリズム

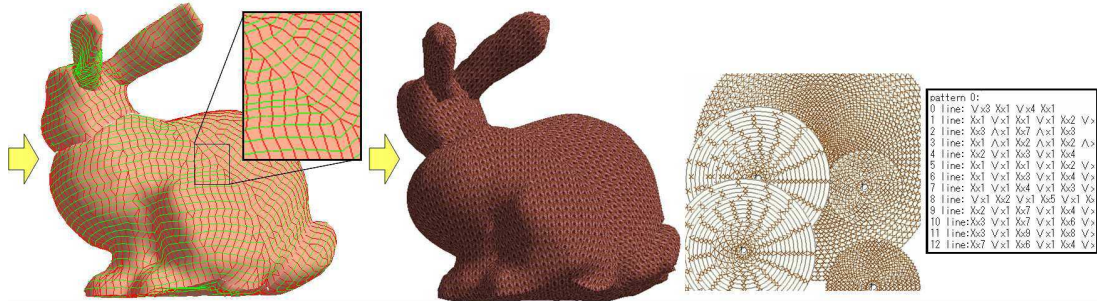
本システムは閉じた 3 次元サーフェスモデルを入力とする．我々は通常の PC 上でインタラクティブなパフォーマンスを出すために荒いメッシュ(頂点数 1000-3000)を利用している．

3 次元モデルからあみぐるみモデルにするためのアルゴリズムは図 5 に示すように，以下の 4 つのステップから成る：

- 1) **領域分割**：まずいくつかの閉じた部品に領域分割する．本システムでは手動で領域分割を行う(図 5(b))．
- 2) **Wrapping**：領域分割されたそれぞれの領域に対して等幅のストリップで包むために，等間隔で複数の等値線を計算する(図 5 (c),(d))．
- 3) **リサンプリングとメッシュ生成**：それぞれの領域に計算された等値線を等間隔にリサンプリングすることで，四角形メッシュと少量の三角形メッシュに分割する(図 5 (e))．四角形メッシュは普通目，三角形メッシュは増やし目と減らし目を表す．
- 4) **シミュレーションと編み図生成**：システムは求まった編み目を並べることで編み図を生成する(図 5 (f))．また，シミュレーションを適用し，あみぐるみ完成予想モデルを提示する(図 5 (f))．



(a) 入力した 3 次元サーフェスモデル (b) 領域分割 (c) 等値線の計算 (d) 等値線のリサンプリング



(e) メッシング (f) シミュレーションと編み図生成

図 5 : システムの概要 .

3.1 領域分割

領域分割を自動で行う手法はいくつも提案されている [15, 16, 17] が、いずれも初心者が簡単に編むための領域分割には不向きである。また、初心者が編みぐるみを編む際には領域分割数は少ないほうが良く、少量の領域分割を行う際には手動で領域分割するほうが簡単で早く行える。さらに、どのような領域分割線にするかということがあみぐるみのデザインに大きく影響するため、本システムでは領域分割の過程は手動で行うこととし、初心者のユーザが簡単に領域分割できるように、3次元モデル上に分割線を描くだけで領域分割ができるユーザインタフェースを備えた。分割されたそれぞれの領域は円と同相もしくは、円に穴が開いた形状と同相であることを仮定する。球やドーナツ形状と同相など、他の位相になっている領域があった際にはシステムは警告を提示する。我々はこの手動領域分割の過程を自動で行えるようなアルゴリズムを今後考えたい。

3.2 ラッピング

本節の目的はそれぞれの領域を複数の等値線で等間隔に包むことである。我々のアプローチはまずシードとなる初期境界線を選び、距離場を計算することを繰り返すことで複数の等値線を等間隔に抽出する。

1つの領域に対して、境界線が1つの場合、その境界線を初期境界線とする(図7上段)。境界線が複数ある場合は、最長の境界線を初期境界線として、そのほかの境界線はホールフィリングを適用する(図7下段)。ホールフィリングには穴の中央に頂点を生成し、穴の境界上の頂点と結んで面を貼ることで穴埋めを行うという簡単な手法を適用している。

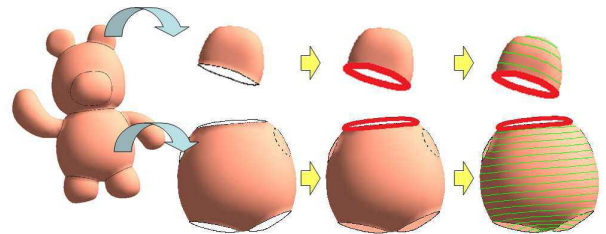


図 6 : 初期境界線の見つけ方。領域が1つの領域分割線を持っていたら、その線を初期境界線とする(上段)。領域が複数の領域分割線を持っていたら、一番長い領域分割線を初期境界線とし、他の領域分割線をホールフィリングする(下段)。

初期境界線を決定したら、それを最初の等値線として、そこから順に等値線を次々に計算していく。システムはまず、等値線からの3次元モデル上のユークリッド距離場を計算し(図7(b))、求めたい幅(r とする)の値をトレースする(図7(c))ことで、次の等値線を得る(図7(d))。トレースには単純な marching cubes (marching triangles)法 [15] を用いている。システムはこの過程を領域内に新たな等値線が定義できなくなるまで繰り返す。

等値線が枝分かれをし、複数の閉じたループに分割された場合がある(図8(a))。この場合には、システムは最長のループを次の等値線として選択し、等値線の抽出を続ける。他のループに関しては新たな領域分割線として別の領域として等値線を抽出する。実際に編

むときには別々の領域として編み、つなげる。図 8(b)のような際には、最後の等値線の内部に中心軸を求め、最後に求めた中心軸が短い場合(最後の等値線が円に近似できる場合)は、システムは中心に線ではなく、頂点を作る(図 8(c))。

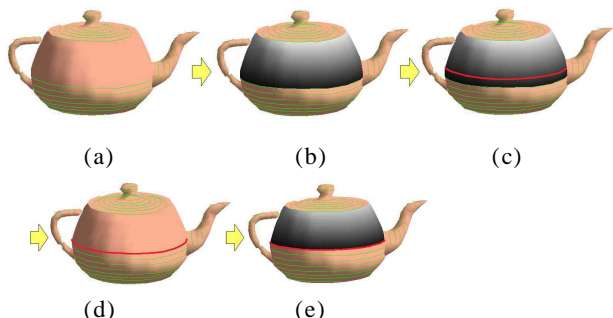


図 7: ラッピングアルゴリズム。(a) 前等値線；(b) 距離場を計算；(c) 次の等値線をトレースする；(d) 求めた次等値線；(e) 次の距離場を計算。

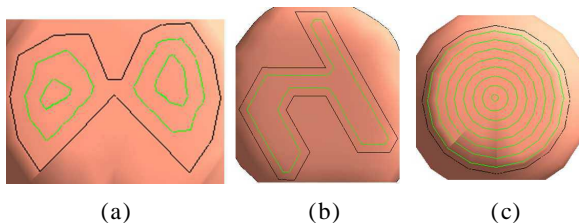


図 8: 等値線の枝分かれする例。

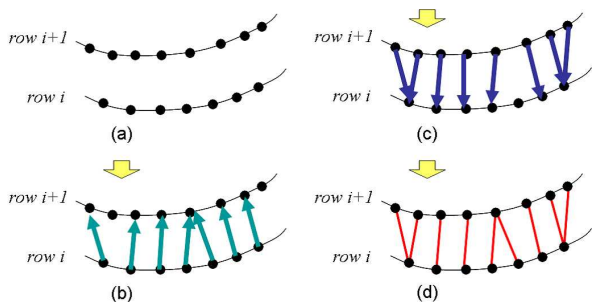


図 9: 等値線と等値線とのメッシュ構築。

3.3 リサンプリングとメッシング

まず図 9(a)のようにそれぞれの等値線をサンプリングする。サンプリングの間隔には等値線の間隔(r)を用いる。次に隣同士の頂点をつないでメッシュを生成する。等値線同士のメッシュのつなぎ方は以下に述べる。

システムはまずターゲットとなる等値線上の頂点から次の等値線上の頂点列の中で一番近い頂点とエッジを結ぶ(図 9(b))。次に、次の等値線上の頂点から前の等値線上の頂点列の中で一番近い頂点とエッジを結ぶ(図 9(c))。これらの和をとることで隣り合う頂点同士のエッジを生成し、メッシュが構築できる。単純な手法であるが、本論文の目的にあった理想的なメッシュを構築することができる。等値線上の 1 つの頂点が次の等値線上の複数の頂点とつながることもありうる

が、最初にリサンプリングしておくことで、我々の試した例の中ではそのようなケースは現れなかった。

3.4 編み図生成とシミュレーション

出来上がったメッシュは四角形メッシュと少量の三角形メッシュから成る。四角形メッシュは普通目を表し、三角形メッシュは増やし目、減らし目を表す。システムは図 10 のようにそれぞれの目にテクスチャを貼ることであみぐるみモデルを表示している。

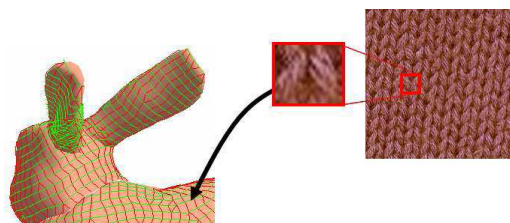


図 10: それぞれのセルにテクスチャを貼ることであみぐるみを表現する。

それぞれの段の編み目は閉じたループでつながっているため、2 次元の編み図に展開するために、始点を定めなければならない。システムは最初の段の始点をランダムに選択し、次段の始点は前段の終点が一番近い頂点を選択する。

円状の編み図やテキストスタイルの編み図では説明するための編み目の記号はたくさん存在するが、我々は初心者にもわかりやすくするために、図 3 のように普通目、増やし目、減らし目の 3 つだけを用いた。編む際には中央から外側へ円状に編んでいく。我々のアルゴリズムは外側の境界線から始め、内側へ進んでいくものなので、編み図を生成する段階で頂点の向きを逆にする。

入力した 3 次元モデルはあみぐるみになることを考慮していない形状であるため、あみぐるみの出来上がり形状は入力した 3 次元モデルとは必ず違いが生じる。このため、表面だけで構築できたあみぐるみに綿を詰めたシミュレーションを適用することであみぐるみの完成予想モデルを提示する(図 5(f))。汎用的な PC でリアルタイムな対話シミュレーションを実現するため、シミュレーションには Plushie[4]の手法を用いた。

4. 結果

図 2(c)と図 11 に実際に本システムで作成したあみぐるみを示す。ユーザは 3 次元モデル上をペイントツールを用いて色塗りをすることができ、システムは自動的に編み図上の対応する編み目の色を変更する。実際に編む際には、編み図上の編み目の色を参照しながら、糸の色を変えることで、図 12 のような途中で色の違うデザインを編むことが可能である。本システムはユーザがアップダウンキーを押

すことで、現在編んでいる段を3次元モデル上と編み図上にマーク表示することができる。また、編み図の編み目の上で右クリックをすることで、その編み目の編み方をイラストレーションで提示する。あみぐるみ製作時間と必要な毛糸の量も提示することができる。表1にそれぞれのモデルでの編み図生成時間を示す。



図 11：本システムを使用して編んだあみぐるみ。



図 12：色塗りした例。モデル上をペイントツールで塗ると、編み図上にも色が反映する。実際に編む際に毛糸の色を変えて編む。

6. まとめと今後の課題

我々は3次元モデルから自動的に編み図を生成するアルゴリズムを提案した。本システムはまずモデルのサーフェスに等幅のストリップを巻きつけ、等間隔で等値線を抽出していく。システムは求めたストリップを等幅でサンプリングした後、編み目とする。最後にスタンダードフォーマットの編み図を生成し、ユーザはそれを使うことで実際にあみぐるみを作成することができる。

我々のアルゴリズムでは、編む段に垂直方向のサーフェスの凸凹形状に関しては図13のように増やし目と減らし目を用いて表現することができる。しかし、編む段に水平方向のサーフェスの凸凹形状は、それぞれの段の位相が円と同相なため、表現することができない。実際にあみぐるみを作る際に専門家のユーザは凹形状はボディの内側へ糸ひっばることで再現している。これらを解決するアルゴリズムを今後考えたい。

本手法は物理オブジェクトの物質の特性を生かした効率的なデザイン手法を示したものである。今後は本手法を皮細工や籐の家具作りなどに適用したい。



図 13：詳細な形状の例。左図はバニーの耳のへこみが表現できている。右図は牛の前足の間接が綿をいれなくて表現できている。

文 献

- [1] J. Mitani and H. Suzuki: Making Papercraft Toys from Meshes using Strip-based Approximate Unfolding, ACM Trans. Graphics, vol.23, No.3, pp.259-263 (2004)
- [2] I. Shatz, A. Tal, and G. Leifman. 2006. Paper craft models from meshes. The Visual Computer: International Journal of Computer Graphics (Proceedings of Pacific Graphics 2006) vol.22, No.9, pp.825-834.
- [3] D. Julius, V.Kraevoy, and A. Sheffer. 2005. D-charts:quasi developable mesh segmentation. Computer Graphics Forum (Proceedings of Eurographics 2005), Vo.24, No.3, pp.981-990.
- [4] Y. Mori and T. Igarashi. Plushie: An interactive pattern design for plush toys. ACM Transactions on Computer Graphics (ACM SIGGRAPH 2007), vol.23, No.3, Article No.45, 2007.
- [5] Y. Mori, T. Igarashi: Pillow: Interactive pattern design for stuffed animals. In DVD publication at SIGGRAPH 2006 Sketches (2006).
- [6] Y. Igarashi, T. Igarashi, H. Suzuki: Knitty:3d modeling of knitted animals with a production assistant interface. In Eurographics 2008 Annex the Conference Proceedings, pp.187-190, 2008.
- [7] M. Gopi, D. Eppstein: Single-strip triangulation of manifolds with arbitrary topology. Computer Graphics Forum (EUROGRAPHICS), vol.23, No.3, 2004.
- [8] J. Mitani, H. Suzuki: Making papercraft toys from meshes using strip-based approximate unfolding. ACM Transactions on Graphics (Proceedings of SIGGRAPH 2004) vol.23, no.3, pp.259-263, 2004.
- [9] P. Alliez, D. Cohen-Steiner, O. Devillers, B. Levy, M. Desbrun: Anisotropic polygonal remeshing. ACM Transactions on Graphics (In Proceedings of SIGGRAPH 2003) vol.22, no.3, pp.485-493, 2003.
- [10] M. Marinov, L. Kobbelt: Direct anisotropic quad-dominant remeshing. In Proceedings of Computer Graphics and Applications (PG '04), pp.207-216, 2004.
- [11] S. Dong, S. Kircher, M. Garland: Harmonic functions for quadrilateral remeshing of arbitrary manifolds. Computer Aided Geometry Design, Special Issue on Geometry Processing vol.22, no.5, pp.392-423, 2005.

表 1: 計算時間(RAM1.5GB, 1.1 GHz Pentium M.)

3次元モデル	頂点数	領域分割数	Segmentation	Wrapping	Meshing	Simulation
House	527	9	0.10 sec	5.22 sec	0.42 sec	15.43 sec
Bunny	1039	6	0.12 sec	13.91 sec	0.60 sec	22.95 sec
teddy	1990	9	0.23 sec	7.76 sec	0.44 sec	6.71 sec
teapot	527	4	0.09 sec	2.39 sec	0.37 sec	6.42 sec
cow	787	12	0.06 sec	6.11 sec	0.45 sec	17.01 sec

- [12]D. E. Hyun, S. H. Yoon, M. S. Kim, B. Juttler: Modeling and deformation of arms and legs based on ellipsoidal sweeping. In *11th Pacific Conference on Computer Graphics and Applications (PG'03)*, 2003.
- [13]S. H. Yoon, M. S. Kim: Sweep-based freeform deformations. *Computer Graphics Forum (Proceedings of Eurographics 06)* vol.25, no.3, pp.487–496, 2006.
- [14]E. B. Shaff, A. B. J. Kuijlaars: Distributing many points on a sphere. In *Springer-verlag new york 19,1,1997*.
- [15]D. Julius, V. Kraevoy, A. Sheffer: Dcharts: quasi developable mesh segmentation. *Computer Graphics Forum (Proceedings of Eurographics 2005)*, vol.24, no.3, pp.981–990, 2005.
- [16]J. M. Lien, N. M. Amato: Approximate convex decomposition of polyhedra. *Technical Report TR06-2002, Texas A M University* (2006).
- [17]L. Lu, Y. K. Choi, W. Wang, M. S. Kim: Variational 3d shape segmentation for bounding volume computation. *Computer Graphics Forum*, vol.26, no.3, pp.329–338, 2007.