

Compatible Intrinsic Triangulations

高山 健志^{† ‡}

[†]国立情報学研究所 [‡]サイバーエージェント

E-mail: kenshi84@gmail.com

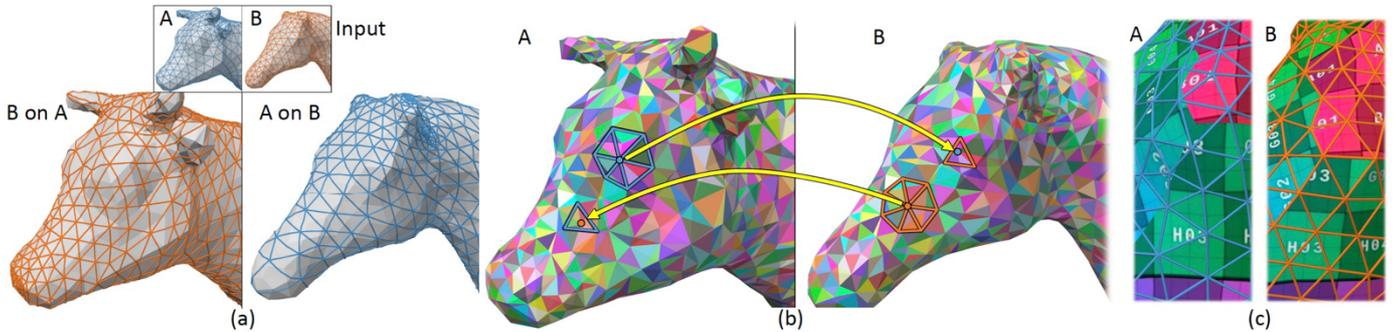


図1. 同じ種数の入力三角形メッシュ A と B に対して, 提案法は歪みの小さい, 連続かつ全単射な写像を構築する. 一方のエッジを他方に写像したものを(a)に示す. 内部的には, すべての頂点・エッジ・三角形が一対一で対応づけられるような Intrinsic Triangulation を A と B の上にそれぞれ構築することで, 写像を定義している. A の元々の頂点が B の元々の三角形の上に挿入され, 逆方向でも同様であることを(b)に示す. 提案法で計算した写像を使って B のテクスチャを A に転写した例を(c)に示す. 特に, B の UV 展開における切れ目(赤色と緑色の境界)が, A の上でも正しく保存されていることに注目されたい.

概要

共通の種数を持つ二つの 3D サーフェス形状の間で, 歪みの小さい同相写像を求める問題は, 3DCG において基盤的に重要である. 本研究では, 平面や球面などの中間的なドメインを介することなく, 入力サーフェス上で直接写像を定義できる, 直感的でシンプルな方法を提案する. 提案法は, 二つの入力三角形メッシュ A と B に対して, Compatible Intrinsic Triangulation (CIT) というものを構築する. これは A と B の上にそれぞれ定義された Intrinsic Triangulation と呼ばれる特殊な三角形分割で, 頂点・エッジ・面の全てが厳密に一対一で対応しており, これをもって同相写像が自明に定義される. 提案する CIT 構築アルゴリズムは経験則のみに基づいて設計されており, 成功の保証は無いものの, 実用上は十分ロバストに動作し, 既存法と同様の修正ニュートン法を使い, 歪みの最適化が可能である. 標準的なデータセットに対して実験を行い, 十分に多くのケースで提案法が成功することを確認した.

はじめに

サーフェス間の対応付け, すなわち写像を計算することは, 様々な場面で必要となる. 特に, 連続かつ全単射であるような写像, すなわち同相写像 (homeomorphism) の計算は, サーフェス間でのテクスチャの転写, テンプレートメッシュのフィッティングといった様々なアプリケーションにおいて, ロバスト性を担保するために重要であるが, その一方で, 同相写像でない一般の写像を求める問題と比べて, 非常に

難しいことが知られている.

同相写像を求める問題の難しさを回避するために, すべての先行研究では, 平面や球面といった「中間ドメイン」を導入し, 二つのサーフェスをその中間ドメインに一旦写像することで同相写像を定義する, というアプローチが採用されてきた. Schmidt らの手法[1]はこのアプローチにおける現在の最先端であり, 定曲率計量 (constant-curvature metric) という概念を使うことで, 入力サーフェスをその種数に応じて平面・球面・双曲平面のいずれかに写像し, それをもって同相写像を定義するというものである. この手法は一意化定理 (uniformization theorem) という高度な理論に基づいており, チャレンジングな入力に対してもロバストに動作するという非常に強力なものであるが, 一方でその概念を直感的に理解することは難しく, 数学理論に詳しくない一般の実務者が実装するには敷居が高いものであった.

本研究では, これに代わる直感的でシンプルな手法として, 中間ドメインを経由しない新たな手法を提案する. その要となるのは Intrinsic Triangulation という近年注目を集めている技術[2]である. これは元々, サーフェス上で熱拡散等の偏微分方程式を解くことを主眼として開発された技術であるが, それをサーフェス間の同相写像を求める問題にも応用できる, ということを初めて示すのが本研究である. 提案法では, 同じ種数を持つ二つの三角形メッシュ A と B を入力とし, Compatible Intrinsic Triangulation (CIT) というものを構築する(図1). これは A と B それぞれの上に定義された Intrinsic Triangulation で, 頂点・エッジ・面の全てが厳密に一対一の対応関係にあるような構造である. これによ

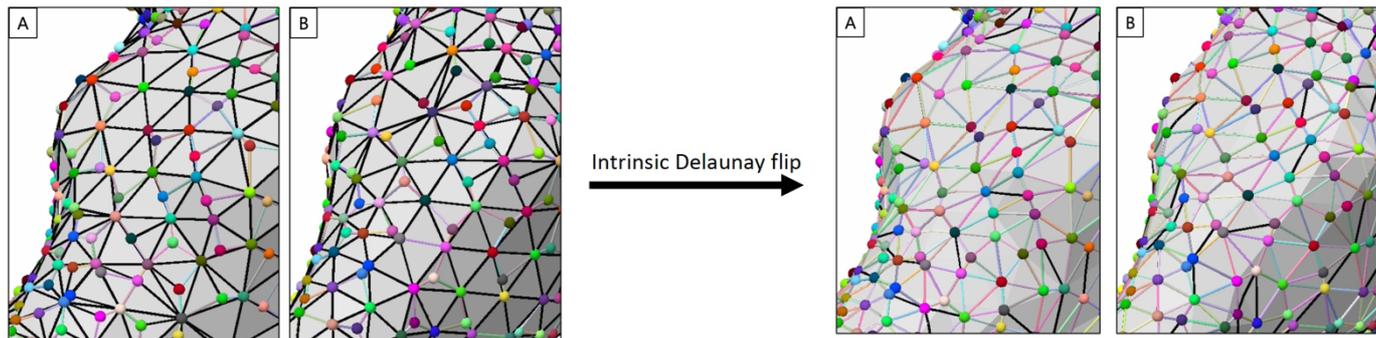


図2. 頂点を挿入した状態(左)と、そこに Intrinsic Delaunay Flip を適用した後の状態(右)

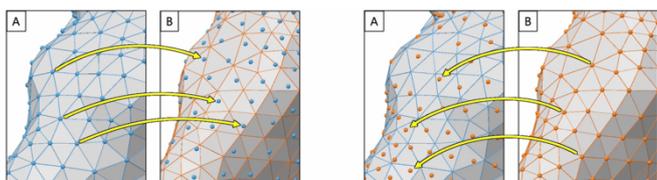


図3. 入力として受け取る頂点イメージ

り, A と B の間の同相写像を自明に定義できる. 提案する CIT 構築アルゴリズムを使い, 既存法[1]と同様の修正ニュートン法による歪みの最適化を行うことで, 歪みの小さい同相写像を既存法[1]と同程度の品質で得ることができる.

なお, 提案法は既存法[1]に対して何らかの実用上の優位性を持つものではない. 実際のところ, 既存法の方が(多少非現実的ではあるものの)チャレンジングな入力を与えられても良好な解に収束するという点(文献[1]の図13を参照)で優れている. しかしながら本研究は, 中間ドメインを經由せず同相写像を求める初めてのアプローチであること, アルゴリズムがシンプルで実装しやすいこと, 実験で十分なロバスト性を確認したこと, およびソースコードを一般公開したこと(*)等の点で, 当該分野の発展に寄与するものである.

アルゴリズム概要

提案法では, 入力として二つの三角形メッシュ A と B に加えて, 「頂点イメージ」というものを受け取る(図3). これは A→B の方向と B→A の方向でそれぞれ定義されるもので, 前者は A の各頂点について, サーフェス B 上の点(三角形 ID と重心座標で表現される)を割り当てる(後者も同様). この頂点イメージを得る既存技術はいくつもあり, どれを使っても良いが, その品質が十分に高いことが本アルゴリズムの成功の必要条件である. ここでいう「品質」を明確に定義することは難しいが, 経験的には, 同相写像に基づく計算で得られたものであれば問題ないことが分かっている. ここでは, 具体的には Aigerman らによる同相写像に基づく手法[3,4]を用いている.

頂点イメージが与えられると, アルゴリズムはまず A の各頂点について, その頂点イメージによる写像先の点を B のメッシュに挿入する(逆方向も同様). この時, A も B も

Intrinsic Triangulation として扱い, Intrinsic Triangulation の操作として新しい頂点を挿入することに注意されたい. この時点で, A と B の Intrinsic Triangulation の間で, まず全ての頂点が一対一に対応している(頂点イメージに基づいて頂点を挿入したため)ことから, 頂点 ID に固有の色を割り当てて可視化することができる(図2). さらに, A のエッジと B のエッジの間で, その両端の頂点の ID が同じであるものは, 「対応済み」エッジと見なし, 共通の ID とそれに対応する(明るい)色を割り当てることができる. そうでないエッジを「未対応」エッジとし, 黒い色を割り当てる. このような方法で描画したのが, 図2左である. この時点では, 明るい色のついた対応済みエッジはごく少数であることに注意されたい.

提案法の次のステップでは, A と B それぞれに対し独立に, Intrinsic Delaunay Flip [2]を実行する. このアルゴリズムは鈍角な三角形をできるだけ少なくするような効果があり, 偏微分方程式を解く際に有用であることが知られている. ここでは, 入力の頂点イメージが高品質であるという前提があるので, A と B の間で対応する頂点の相対的な位置関係が大まかに類似する傾向があり, そのおかげでこのステップの実行後は, 未対応なエッジがごく少数残されるだけになる(図2右). 本稿では説明を省略するが, 提案法の残りのステップでは, これらの少数の未対応エッジを, 経験則に基づいて設計されたエッジ切り替えアルゴリズム等を使うことで, 対応済みエッジに修正していく. 逆に, この Intrinsic Delaunay Flip の後で未対応なエッジが大量に存在する場合, 提案法は失敗する.

実験結果

図4に, 先行研究[1]と同じデータセットに提案法を適用した結果のテクスチャ転写による可視化を, 表1に写像の歪みの値の比較を示す. CIT の生成に関しては10個全部のケースで成功し, また最適化後の歪みについても Ant-Octopus と Pig-Armadillo という著しく形状が異なる難しいケースを除けば, 同程度のエネルギー値に収束している.

提案法がどの程度ロバストかを検証するために, 形状処理アルゴリズムで広く使われているプリンストン大学のデータセット[5]を元に, 88個の形状ペアをランダムに作成し, 提

(*)<https://github.com/kenshi84/compatible-intrinsic-triangulations>

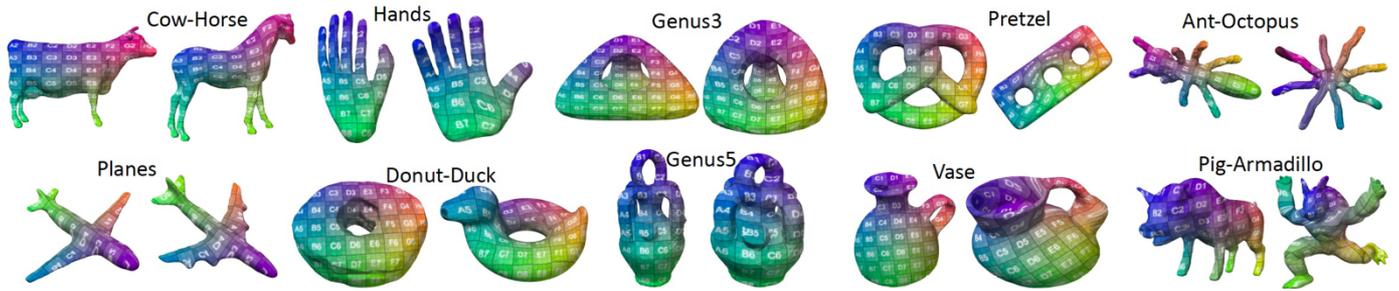


図4. 先行研究[1]と同じデータセットに提案法を適用した結果を、テクスチャ転写で可視化した様子(Aの上で単純な投影により生成したテクスチャをBに転写).

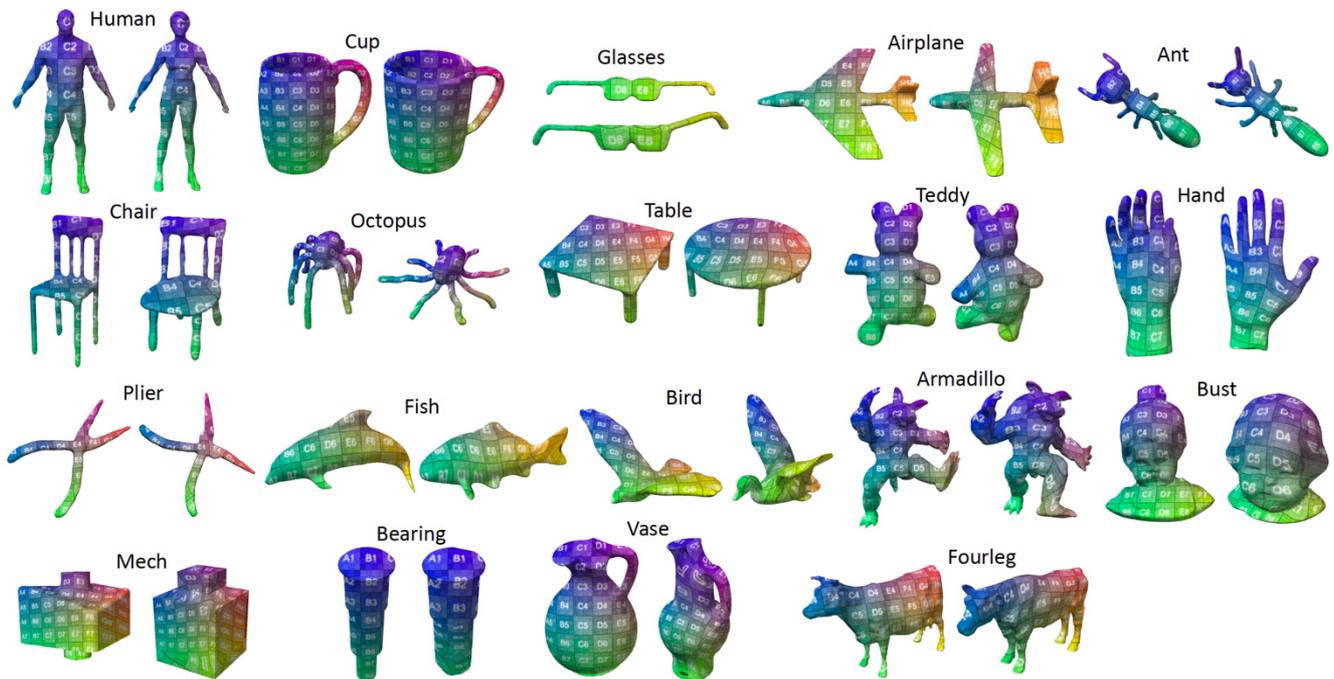


図5. 19個のカテゴリから成るプリンストン大学のデータセット[5]を元に作成したランダムな形状ペアに対し、提案法を適用した結果.

表1. 先行研究[1]との比較における数値. #Fは二つの入力三角形メッシュの面の数の合計, #Nは(提案法の)最適化のステップ数, #Tは最適化に要した時間, Eは収束後の画像の歪みエネルギー.

Case name	#F	#N	T	E	[Schmidt et al. 2020]
PLANES	25k	231	15.0h	4.35	4.69
COW-HORSE	10k	225	3.8h	4.73	5.28
HANDS	32k	366	37.6h	4.35	4.83
GENUS3	3k	440	2.9h	4.21	4.36
GENUS5	8k	767	15.0h	4.49	4.74
PRETZEL	12k	532	11.2h	5.25	5.53
DONUT-DUCK	24k	585	78.9h	4.68	4.77
VASE	10k	542	9.1h	4.60	5.00
ANT-OCTOPUS	8k	4	1m	98.3	16.2
PIG-ARMADILLO	22k	370	19.9h	8.10	7.78

案法を実行したところ, 81個のケースで成功し, 92%という高い成功率を得た(図5).

今後の課題

提案法の失敗例を図6に示す. このように著しく異なる形状のペアに対しては, Intrinsic Delaunay Flipを実行した時点で, 非常に多くの未対応なエッジが残ってしまい, 経験則に基づく提案アルゴリズムは, そのような場合に対応できない.

また, 最適化が途中で局所解に陥ってしまう傾向が, 既存法[1]よりも顕著であることを確認している. 最後に, 入力

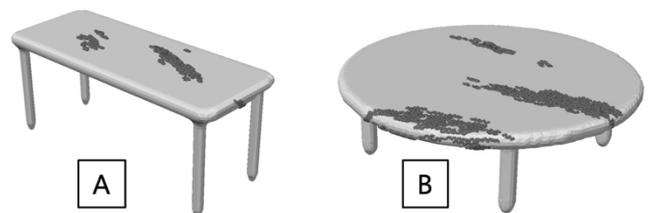


図6. 提案法の失敗例.

のメッシュのサイズが大きくなると、計算時間が長くなってしまふ問題がある。これに関しては、多重解像度のメッシュを用い、階層間の全単射な写像を求める手法[6]を利用することで、高速化できる可能性が考えられる。

謝辞

建設的な批判とコメントを寄せてくれた匿名の査読者に感謝する。また安東遼一氏には、大規模な実験を行う上で必要不可欠な潤沢な計算機資源を提供していただいた。なお、本研究は著者が国立情報学研究所に在籍中に投稿し、サイバーエージェントに異動してから修正と発表を行ったものである。

参考文献

- [1] Patrick Schmidt, Marcel Campen, Janis Born, and Leif Kobbelt, “Inter-Surface Maps via Constant-Curvature Metrics,” in *ACM Trans. Graph.* 39, 4, Article 119 (August 2020), 15 pages, 2020.
- [2] Nicholas Sharp, Yousuf Soliman, and Keenan Crane, “Navigating Intrinsic Triangulations,” in *ACM Trans. Graph.* 38, 4, Article 55 (August 2019), 16 pages, 2019.
- [3] Noam Aigerman, Roi Poranne, and Yaron Lipman, “Seamless Surface Mappings,” in *ACM Trans. Graph.* 34, 4, Article 72 (August 2015), 13 pages, 2015.
- [4] Noam Aigerman and Yaron Lipman, “Hyperbolic Orbifold Tutte Embeddings,” in *ACM Trans. Graph.* 35, 6, Article 217 (November 2016), 14 pages, 2016.
- [5] Xiaobai Chen, Aleksey Golovinskiy, and Thomas Funkhouser, “A Benchmark for 3D Mesh Segmentation,” in *ACM Trans. Graph.* 28, 3, Article 73 (August 2009), 12 pages, 2009.
- [6] Zhongshi Jiang, Teseo Schneider, Denis Zorin, and Daniele Panozzo, “Bijective Projection in a Shell,” in *ACM Trans. Graph.* 39, 6, Article 247 (December 2020), 18 pages, 2020.