

サブセグメント分割に基づく 3D スキャンモデルのインタラクティブセグメンテーション

真殿 航輝*† 五十嵐 健夫† 加藤 大晴† 橋本泰輔† Fabrice Matulic† 高木 士† 樋口啓太†

概要. 3D スキャンモデルにおいてパーツ毎のテクスチャセグメンテーションは、モデルの再編集やマテリアル付与の目的で重要である。従来の手法では手動によるセグメンテーションが手間である一方、自動セグメンテーション技術ではユーザの意図通りに分割される保証がない。本研究では、自動セグメンテーションと最小限の手動操作を組み合わせたインタラクティブセグメンテーションツールを提案する。このツールは、前処理として 3D モデルを多視点から自動セグメンテーションすることでテクスチャをサブセグメントに分割し、実行時にユーザが 3D モデルビューで簡単な線描画でクラスタ化する事で、最終セグメンテーション結果を得る。評価の結果、本アプローチは手動セグメンテーションや標準的な 3D コンピュータグラフィックスソフトウェアに比べて精度と品質を向上させ、UV テクスチャマップの自動セグメンテーションに比べてより詳細なセグメントを生成できることが示された。

1 はじめに

コンピュータビジョンや機械学習技術の進展により、高品質な 3D スキャンモデルが幅広いユーザに利用可能である。3D 再構築技術は、写真から正確なモデルとテクスチャの両方の生成を可能にした。3D スキャンモデル利用が広がるにあたり、後処理が重要となり、色の変更、材質の修正、部品の形状変更など、特定のユーザ要件に対してモデルを編集できる事が求められている。一方で、3D モデリングソフトウェアで作成された 3D スキャンモデルとは異なり、3D スキャンモデルではパーツや意味のあるセグメントに分割する方法は自明ではない。そのため、ユーザは後処理で 3D スキャンモデルのセグメンテーションを行う必要がある。

3D スキャンモデルのセグメンテーション方法として、3D ソフトウェアを使用してテクスチャを手動で分割する方法が挙げられる。例えば、Blender¹は 3D ペイント機能としてブラシや塗りつぶしツールを提供しており、これにより 3D スキャンモデルをユーザの意図通りにセグメンテーションすることが可能である。しかし、特にパーツ間の境界を精度よく分割することは手動セグメンテーションでは手間がかかり、3D ペイントには精密な操作が必要である。また、インタラクティブな 3D のセグメンテーションに関する従来手法として、Click-base [7] の手法や Graph-cut を用いた手法 [2] が提案されている。しかしながら、複数のクラスの分割の際にユーザの意図する形に分割する事が難しく、特に今回

対象としているような 3D スキャンモデルに対しての有効性が十分に検証されていない。

別のアプローチとして、自動セグメンテーション手法を利用する方法がある。k-means クラスタリング [9] やスーパーピクセルセグメンテーション [1] などのヒューリスティック手法はユーザの介入なしにセグメンテーション可能であるが、幾何情報とテクスチャ情報の両方を考慮した手法が少なく、ユーザの意図しないセグメンテーション結果になる事が多い。ディープラーニングに基づくより高度な技術は、意味のあるセグメントの自動抽出において優れた能力を示している [4, 11, 3] が、ユーザのニーズやアプリケーションの要求に合致する結果を保証するわけではない。ユーザの多様な好みや理想的なセグメンテーションに対する精密な結果を自動セグメンテーションだけで達成する事は難しく、3D スキャンモデルを使用するそれぞれのアプリケーションのニーズに適切に対応するためには、インタラクティブなセグメンテーション技術が必要である。

本研究では、手動と自動セグメンテーション手法の欠点を克服するために、2つを組み合わせた半自動インタラクティブセグメンテーション技術を提案する。図 1 に全体像を示す。まず、3D スキャンモデル上でのテクスチャ画像をサブセグメントに分割する。その後、ユーザが 3D ビュー上でサブセグメント同士を連結する。具体的には、ユーザがスクリブルをモデルに描き込むと、交差するサブセグメントにセグメンテーション ID が割り当てられ、ID が周辺のパッチに伝播してセグメンテーションが生成される。このプロセスを繰り返し、ユーザがスクリブルを追加しながら希望するセグメンテーションを作成していく。ユーザ実験を通して本手法が伝統的な手動ペイントインターフェースに比べより効率的

Copyright is held by the author(s).

* 早稲田大学

† Preferred Networks

¹ <https://www.blender.org>

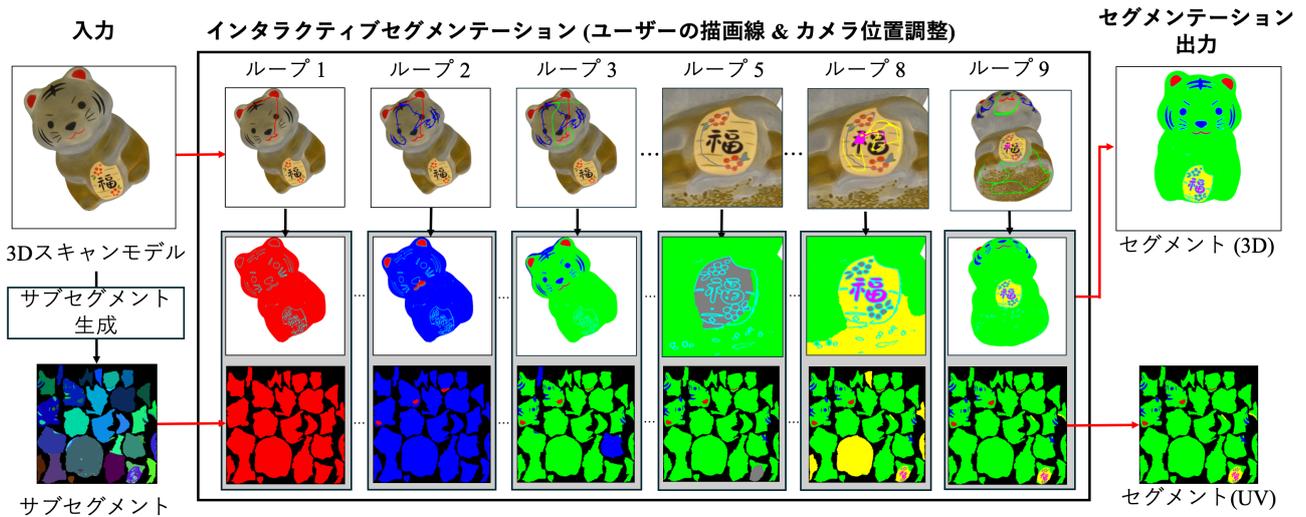


図 1. 3D スキャンモデルのインタラクティブセグメンテーションのワークフロー。自動セグメンテーションを用いて、事前処理ステップでテクスチャをセマンティックかつ十分に小さいサブセグメントに分割する。ユーザはその後、3D スキャンモデル上にスクリブルを描くことで、サブセグメントを反復的にセグメント化する。サブセグメント ID は、ユーザが書いた最も近いスクリブルの ID が割り振られる。

にセグメンテーション出来ることを確認し、自動セグメンテーションの前処理と手動セグメンテーションの後処理を組み合わせた提案法の有効性を明らかにした。

2 関連研究

2.1 自動 3D セグメンテーション

初期の自動 3D セグメンテーションは、K-means [9], およびグラフカット [8] などのヒューリスティック手法が用いられていた。近年では、Segment Anything Model (SAM) [6] が 2D セグメンテーションで顕著な精度改善を達成しており、SAM の応用技術であるマッチングベースのアプローチ、ニューラルラジアンフィールド (NeRF) ベースのアプローチ [10], およびガウスプラットフォームベースのアプローチ [5] などの様々な 3D セグメンテーション手法が提案されている。これらの先行研究は複数の画像からメッシュを生成するアプローチであるのに対し、本研究は 3D スキャンモデルと対応する UV テクスチャを用いており、関連研究とは異なる問題設定である。

本研究と同じ問題設定である SAM3D [11] は、複数視点のセグメンテーション結果をそれぞれマッチングベースで結合・分割し、自動で 3D 物体のセグメンテーションを生成する技術である。しかし、結合する際のハイパーパラメータに精度が依存し、ユーザが希望するセグメンテーションの獲得を保証するアプローチではない。

本研究では、3D レンダリング画像と UV テクスチャのセグメンテーション結果から小さなサブセグ

メントと対応する距離行列を生成し、ユーザが効率的にサブセグメントをグループ化可能なインタラクティブセグメンテーション技術を提案する。

2.2 インタラクティブ 3D セグメンテーション

インタラクティブ性は、ユーザの負担を減らすために重要な役割を果たしている。3D ペイントソフトウェアである Blender¹ や ZBrush² は、ユーザにとって便利な描画ツールを用意する事で、インタラクティブ性を向上させている。特にバケツツールとブラシツールを組み合わせることで、セグメンテーションの効率を大幅に向上させている。しかし、これらの解決策は一定のソフトウェア習熟度や作業時間を要求するため、すべてのユーザにとって使いやすいとは限らない。

本研究ではより多くのユーザにとって使いやすく効率的なインタラクティブセグメンテーションを実現するために、ユーザのスクリブルのみを入力とするインタラクティブセグメンテーション技術を提案する。

3 ユーザーインターフェース

3.1 対象問題設定

3D スキャンモデルには、3D メッシュと UV テクスチャの両方が含まれる。しかし、パーツ毎にセグメンテーションはされていないため、図 2 のような質感やテクスチャの編集を行うためには、3D メッシュと UV テクスチャに対しセグメンテーションを行う必要がある。ユーザがテクスチャを手動で分類し、UV ピクセルにクラスラベルを割り当てること

サブセグメントに基づく 3D インタラクティブセグメンテーションシステム



図 2. 3D スキャンモデルのセグメンテーションと編集

は可能であるが、パーツ間の境界を精緻に分割することに時間がかかるといった課題がある。

そこで本研究では、UV テクスチャを大きさが異なるサブセグメントに自動で分解し、ユーザがスクリブルを使用してサブセグメントを効率的にグループ化出来るインタラクティブセグメンテーション手法を提案する。手動ラベリングの負担を大幅に削減するために 3D スキャンモデルのセグメンテーションプロセスを簡素化する事で、本研究はユーザ体験と効率を向上させることを目指している。

3.2 ユーザフロー

提案システムではユーザがスクリブルの色を選択し、テクスチャ付き 3D スキャンモデルの表面にスクリブルを描く。システムは、スクリブルと交差するサブセグメントに色に対応しているセグメント ID を割り当てる。その後システムは、セグメント ID を近くのサブセグメントに伝播させる。これにより、スクリブルが交差していないサブセグメントに対してもセグメント ID を割り当てることで、セグメンテーション結果を推定する。システムが推定した結果に応じてユーザは追加でスクリブルを加えて結果を修正する。このようにシステムと反復的に作業する事で、希望するセグメンテーションを得られる。

Figure 1 に例を示す。プロセスは、3D スキャンモデルとセクション 4 のアルゴリズムに基づいて生成されたサブセグメントを初期状態として始める。ユーザとの初回の反復 (ループ 1) では、ユーザが赤色の線を適用して、黒い眉毛・目・ひげなどの特徴を一緒にクラスタリングする。その結果、すべてのサブセグメントが赤色としてセグメント化されている。次の反復 (ループ 2) では青色の線が追加され、セグメント境界が再評価される。この結果、耳や口は同じ赤のセグメントになり、その他の顔の表面は青いセグメントとして区別されている。これらの反復プロセスにより、ユーザはセグメントを修正可能である。またユーザは反復プロセスの過程でカメラ位置の調整によりモデルの視点を変更し、モデル全体のセグメンテーションの精度を確認することができる。

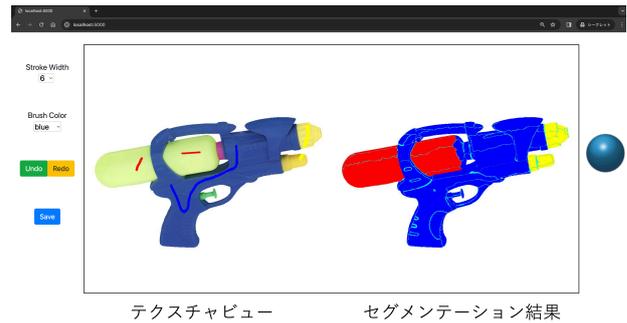


図 3. システムのスクリーンショット：このシステムには、ブラシ調整 (ブラシサイズ, 色), 元に戻す/やり直す操作, UV テクスチャの保存操作, カメラ制御が含まれている。メイン画面では、3D スキャンモデルの UV テクスチャとセグメンテーションを同時に確認でき、サブセグメントの境界線は常にライトブルーで表示される。テクスチャビューに描画された複数色のスクリブルに応じて、右画面のセグメンテーション結果が更新される。

3.3 ツール

図 3 に、提案のインタラクティブセグメンテーションツールのスクリーンショットを示す。本システムは、ユーザが 3D 表面にスクリブルを付与する半自動セグメンテーションを対象としている。以下の項目が提案システムの特徴となっている。

リアルタイムプレビュー インターフェースでは、ユーザがセグメントを作成しながら、3D スキャンモデルのテクスチャとジオメトリの両方を同時に確認できる。システム上では、元のテクスチャを表示するビュー (図 3, 左側) と、セグメンテーション領域を表示するビュー (図 3, 右側) の 2 つが利用されている。

3D 表面への線描画 セグメンテーション境界を手動でトレースする事が難しいため、システムはテクスチャを小さいサブセグメントに分割し、ユーザはスクリブルを用いてサブセグメント同士を連結する。インターフェースでは、2 つのモデルビューのいずれかにスクリブルを追加することが出来る。

ブラシ選択 ユーザは、セグメント ID を表すブラシの色を選択し、スクリブルを付与する。ブラシ選択による分類は、各セグメントに対してレイヤーを作成するよりも直感的で効率的であると考えている。

Undo & Redo ボタン ツールには「やり直す (Undo)」と「元に戻す (Redo)」ボタンが含まれており、ユーザはスクリブルの仮追加・必要に応じたスクリブルの削除が可能である。

4 サブセグメンテーションと実行時処理

図 4 に提案の前処理の全体像を示す。初めに提案手法では、3D スキャンモデルの UV テクスチャ画

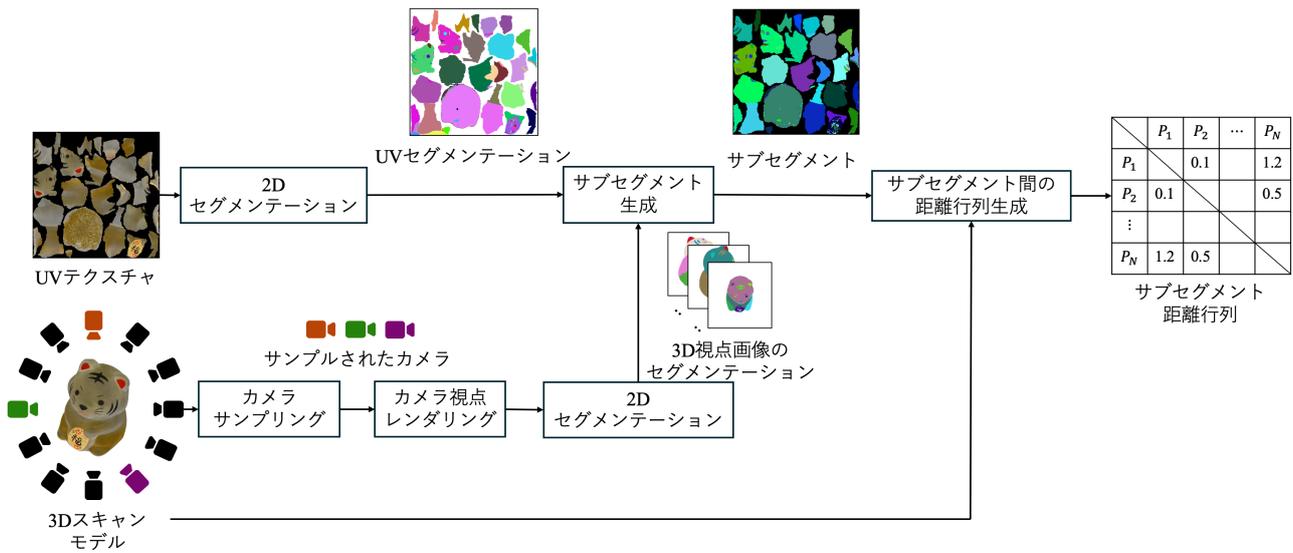


図 4. 前処理のステップは、主に 2 つのフェーズで構成されている。サブセグメント生成とサブセグメント間の距離行列生成である。サブセグメント生成には、3D モデルと対応する UV テクスチャを用いる。最初に、3D スキャンモデル全体を包含するカメラ群をサンプリングする。次に、3D 視点画像と UV テクスチャの両方に対して 2D セグメンテーションを行う。その後、2 つのセグメンテーションに基づいてサブセグメントを生成する。その後、生成したサブセグメントを用いてサブセグメント間の距離行列を計算する。この距離行列には、全てのサブセグメント間の距離情報が保持されており、ユーザーによるインタラクティブセグメンテーションの際に用いる。

像と特定の視点からのレンダリング画像から HQ-SAM [4] を用いて 2D セグメンテーション結果を得る。特定視点からのレンダリングの際に、3D スキャンモデルの周囲を包含するようにカメラの位置を最適化することで、3D スキャンモデル全体のセグメンテーション結果を取得している。上記 2 つのセグメンテーション結果の境界線を取得し、一つの UV テクスチャ上にまとめる。その後まとめた UV テクスチャに対して連結成分解析を計算する事で、サブセグメントを獲得する。その後隣接するサブセグメントの 3D 上での幾何的距離と色情報の 2 つを掛け合わせた値を計算する。その後、ダイクストラ法を用いて全てのサブセグメント間のペアに対する距離を計算し、計算結果をサブセグメント距離行列として生成する。

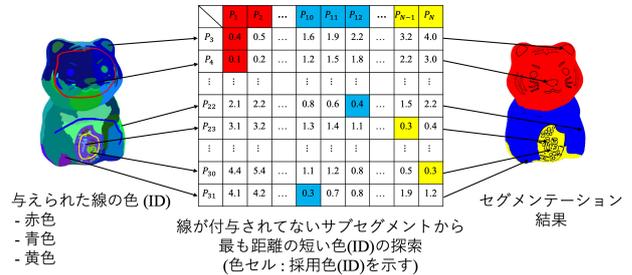


図 5. 実行時処理は、ユーザーのスクリブルに基づいて、サブセグメント距離行列から最も近いユーザーのスクリブルを確認し、スクリブルがないサブセグメントにセグメンテーション ID を割り当てる。その後、最終的に全体のセグメンテーション結果を得る。

図 5 に実行時処理を示す。ユーザが提案インターフェースによりセグメンテーションを行う際には、サブセグメントに対して描画線を付与していき付与された線から付与されていないサブセグメントの距離を事前計算した距離行列から計算することで、描画線が付与されていない箇所のセグメントを自動決定する。具体的には複数の ID (色) の描画線からの距離を計算し、最も距離が短い ID をサブセグメントに割り当てる。

5 ユーザ実験

ユーザインターフェースを用いたセグメンテーション精度の評価を目的として、3D スキャンモデルのユーザ実験を実施した。参加者のパフォーマンスと各インターフェースに対する体験の比較のため、被験者内デザインを採用した。

5.1 実験設定

5.1.1 参加者

当機関から 8 名の参加者を募集した。参加の前提条件として、ゲームや CAD を含む 3D コンピュー

タグラフィックスの経験が必要である。前提条件が満たされていることを事前調査により確認した。

5.1.2 ベースライン

本研究のベースラインとして、3D グラフィックスソフトウェアの Blender を利用した。このインターフェースは、セグメンテーションの際に UV テクスチャと 3D メッシュを同時に表示および操作できる有用なソフトウェアである。また、Blender を用いたセグメンテーションでは、主にブラシツールやパケツツールが用いる。

5.1.3 対象 3D スキャンモデル

本実験では、3D スキャナで作成されたブロッコリーと水鉄砲の 3D スキャンモデルを利用した。ブロッコリーモデルは花蕾と茎から成る。モデルの構造は一見単純に見えるが、花蕾と茎の境界は色境界が曖昧であり識別しづらい。さらに、茎の裏側にブラシを入れる際にカメラや 3D スキャンモデルの位置を調整する必要があり、作業コストが大きい。水鉄砲モデルは、比較的明確な色分けがされているが、水タンク、引き金、ノズル周りなどのアクセスが難しい領域がいくつか存在する。上記のモデル選定は、2つの異なる性質を持つモデルのセグメンテーションを通して、提案技術の性能を確認する事が目的である。

5.1.4 手順

各参加者に対して、合計 1 時間の実験時間を設けた。セッションの開始時に、参加者に実験手順について説明し、同意書への署名を依頼する。提案されたインターフェースおよびベースラインインターフェースの操作方法についてを行った後、練習用モデルを使用して、5~10 分間の両方のインターフェースでトレーニングを行い、ツール利用への理解を促した。

本実験では、参加者にブロッコリーと水鉄砲のモデルを用いたセグメンテーションタスクをそれぞれのインターフェース上で行うよう指示を与えた。すなわち、参加者は合計で 4 つのセグメンテーションタスクを実施することになる。順序効果を減少させるために、インターフェースの順序は参加者ごとに入れ変えた。

各タスクには 5 分の時間制限が設けられた。ただし、参加者がセグメンテーションに満足した場合は、早期終了も可能である。タスクの目的は、正解データにできるだけ近いセグメンテーションを達成することである。ルールとして参加者は、(1) 実験中にセグメンテーション正解図の参照が可能、(2) 基本的な操作についての質問が可能、(3) セグメンテーションツールのより効率的な使い方に関する質問はなし、とした。また参加者は、ペインティングに使

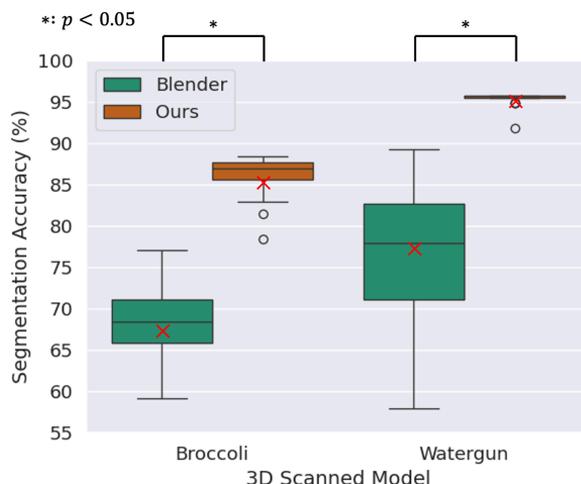


図 6. ボックスプロットの結果は、提案手法はほぼすべてのケースでセグメンテーション精度を改善した事を示している。× は各平均値を示す。

用する色を自由に選択でき、色の違いは最終結果の評価に考慮しないものとした。

5.1.5 評価指標

参加者が作成したセグメンテーションマップの精度を評価するために、正解データとのセグメンテーション結果の UV テクスチャの一致率 IoU を計算することで、最終精度を計算した。

5.2 結果

セグメンテーション精度 図 6 に、ユーザが生成したセグメンテーションの精度をまとめたボックスプロットを示す。平均して、提案インターフェースはブロッコリーの 3D スキャンモデルに対して $85.40\% \pm 3.29$ の精度を達成し、ベースラインの平均精度 $67.71\% \pm 5.90$ を上回っている。また水鉄砲の 3D スキャンモデルでは、提案インターフェースが平均精度 $95.58\% \pm 0.28$ を示し、ベースラインの $77.27\% \pm 10.19$ を上回った。これらの結果は、提案インターフェースが両方のモデルにおいて平均精度でベースラインを一貫して上回っていることを示している。これらの違いの統計的有意性を、結果の非正規分布を示唆する Shapiro-Wilk 検定の後に実施した Mann-Whitney U 検定によって確認した。その結果 Broccoli モデルで $p < 0.05$ 、Watergun モデルで $p < 0.05$ と記録され、統計的有意性を確認した (p 値は Benjamini-Hochberg 手法により補正済み)。

図 7 に、ユーザ実験の中で最も精度の高いセグメンテーション結果と自動 3D セグメンテーションモデルである SAM3D [11] の参考結果を示す。図 7 の黒いボックスを詳しく見ると、提案手法のセグメ

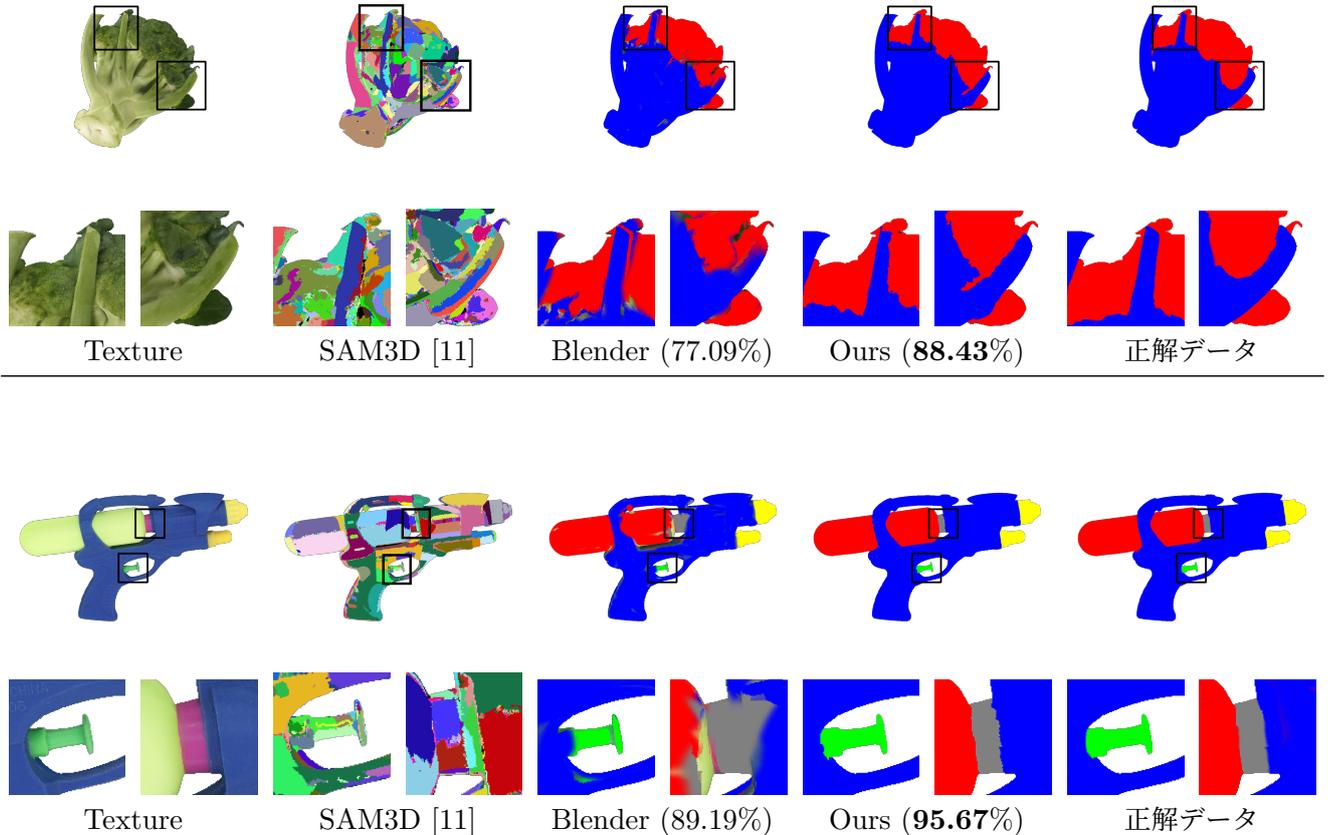


図 7. ユーザー研究におけるセグメンテーション結果を示す。各条件において最も高いセグメンテーション精度を示すモデルを可視化する。Blender に対する改善を黒いボックスで強調する。

ンテーションがテクスチャに沿って分離されており、ベースラインに比べても正解セグメンテーションに対するエラー度合いが小さいことを確認した。この結果は、3D スキャンモデルの塗りやすさがインターフェースのパフォーマンスに大きく影響することを強調しており、半自動セグメンテーションの重要性を示唆している。

ユーザの主観評価 ユーザの好みを評価するためにベースラインと提案手法のどちらのインターフェースを好みかを5段階で質問した：ベースラインを強く好む、ベースラインを好む、どちらでもない、提案インターフェースを好む、提案インターフェースを強く好む。その結果、7人の参加者が提案のインターフェースを提案インターフェースを強く好むと評価し、1人の参加者が提案インターフェースを好むと評価した。参加者は提案インターフェースの方が、より少ないユーザ介入でセグメンテーションを迅速に行うことが可能だと評価した。一部の Blender に慣れた参加者はベースラインの方が直感的であると評価したが、提案手法が有効であることを認めた。

完了時間 Broccoli 3D スキャンモデルの場合、ユー

ザは両方のインターフェースにて与えられた5分の経過後に作業を打ち切った。Watergun 3D スキャンモデルでは、ユーザは提案インターフェースを使用して平均247秒で完了し、ベースラインでは与えられた5分の経過後に作業を打ち切った。

6 まとめ

本研究では、3D スキャンモデルに対する半自動3D インタラクティブセグメンテーション手法を提案した。提案手法では、はじめに自動前処理フェーズで小さなサブセグメントを生成し、ユーザが3D スキャンモデルビュー上にスクリブルを描くことで、サブセグメント群をセグメント毎にクラスタリングすることを可能にした。実験では提案手法が従来の手動セグメンテーション技術に比べて、精度と品質向上に貢献する事を確認した。将来研究では、セグメンテーションの高精度化に向けたセグメンテーションパッチ生成過程やインタラクティブインターフェースの改善を行う予定である。

参考文献

- [1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.
- [2] D. Freedman and T. Zhang. Interactive graph cut based segmentation with shape priors. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 1, pp. 755–762 vol. 1, 2005.
- [3] H. Guo, H. Zhu, S. Peng, Y. Wang, Y. Shen, R. Hu, and X. Zhou. SAM-guided Graph Cut for 3D Instance Segmentation. *arXiv preprint arXiv:2312.08372*, 2023.
- [4] L. Ke, M. Ye, M. Danelljan, Y. Liu, Y.-W. Tai, C.-K. Tang, and F. Yu. Segment Anything in High Quality. In *NeurIPS*, 2023.
- [5] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4), 2023.
- [6] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.
- [7] T. Kontogianni, E. Celikkan, S. Tang, and K. Schindler. Interactive object segmentation in 3d point clouds. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2891–2897. IEEE, 2023.
- [8] M. Landy and J. A. Movshon. *The Plenoptic Function and the Elements of Early Vision*, pp. 3–20. 1991.
- [9] J. MacQueen, et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1, pp. 281–297. Oakland, CA, USA, 1967.
- [10] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.
- [11] Y. Yang, X. Wu, T. He, H. Zhao, and X. Liu. Sam3d: Segment anything in 3d scenes. *arXiv preprint arXiv:2306.03908*, 2023.