

AR ジオラマにおける領域分割に基づく三次元形状復元手法

馬勝 淳史[†] 間下 以大^{†,††} 清川 清^{†,††} 竹村 治雄^{†,††}

[†] 大阪大学 大学院情報科学研究科 〒 565-0871 大阪府吹田市山田丘 1 番 5 号

^{††} 大阪大学 サイバーメディアセンター 〒 560-0043 大阪府豊中市待兼山町 1 番 32 号

E-mail: [†]umakatsu@lab.ime.cmc.osaka-u.ac.jp, ^{††}{matshita,kiyo,takemura}@ime.cmc.osaka-u.ac.jp

あらまし 近年, 拡張現実感 (Augmented Reality, AR) と呼ばれる, 現実空間に対して計算機情報を重畳表示する技術が身近なものとなり始めている. 筆者らの研究グループでは, 対象の実物体の三次元形状復元, ストローク入力による動作記述が可能である AR ジオラマと呼ぶシステムを提案している. しかし, このシステムが復元するモデルは復元精度が良くないという問題があった. 本研究では AR ジオラマの三次元形状復元に関して新たな手法を検討する. 新たな手法では色情報を用いて領域分割を行うことによって, 三次元形状復元を行う. 特徴点情報ではなく, 色情報を用いることで, 形状復元のテクスチャ依存性を低くする. ユーザからのマウス入力によって復元したい物体を指定してシルエットを抽出し, 得られたシルエットとカメラ自己位置姿勢から三次元形状復元を行う. 実装の結果, 従来手法では復元不可能であったテクスチャが少ない物体の三次元形状復元ができること, 従来手法で復元可能な物体を扱う場合でも従来手法と比べて精度のよい三次元形状モデルの生成ができることを確認した.

キーワード AR, 三次元形状復元, 領域分割, グラフカット, 視体積交差法

1 はじめに

近年, 拡張現実感 (Augmented Reality, AR) と呼ばれる, 現実空間に対して計算機情報を重畳表示する技術が身近なものとなり始めている. 特に, カメラで平面上のパターン (マーカ) を認識して 3D の物体を重畳表示する AR は, 近年盛んに利用されている. このように一般に AR 技術が普及しつつあるのは, 加藤らによる ARToolKit [1] などの, 扱い易いマーカベーストラッキングを行うライブラリの登場によるところも大きい. また, 近年ではマーカを用いない自然特徴点追跡による AR も数多く開発されており, Parallel Tracking And Mapping (PTAM) [2] などのマーカレストラッキングを行うライブラリが公開されている. しかしながら, 多くの AR システムは予め設定された作業環境で予め用意されたコンテンツを再生するのみである. 即ち, AR 環境を体験しながら, ユーザが随時実環境の一部を直接コンテンツとして取り込んだり, それに対する振る舞いを AR 環境にしながら付加・編集していくというようなことは, 従来の AR システムでは困難であった. このような AR システムの実現が難しい原因としては, 実環境から実物体を復元した三次元形状モデル (以下, 単にモデル) を生成することが容易ではないという点が挙げられる. 特に実物体の観測に基づくモデル生成では, 一般に複雑な操作や特殊なシステムが必要である [3]. この問題を解決するために, インタラクティブな三次元形状復元の試みがなされている [4]. しかし未だに使い勝手のよい, 精度の高い手法は確立されていない.

AR 環境においてモデルやその振る舞いを記述するシ

ステム (AR オーサリングシステム) の開発によって, AR システムの利便性やエンタテインメント性の向上が期待できる. 例えば, 実世界の物体表面に沿ってバーチャルな注釈を書き込む, 生成した Computer Graphics (CG) キャラクタが実世界の物体を避けながら歩き回る, 実物体の色や形を変化させたりコピーや削除を行う, といった様々な利用が期待できる.

このようなシステムの先行研究として, 立石らの AR ジオラマシステム [5] (以下, 単に AR ジオラマ) がある. 立石らのシステムでは, 実環境から復元したモデルをインタラクティブに生成, 動作を記述できる特徴を持つ. しかし, モデル精度が良くないという深刻な問題がある. 重畳表示させるモデルの精度が悪ければ, リアリティに欠ける, 衝突判定動作などが正確にできないなどの問題がある. つまり, AR ジオラマを構築する上で重畳表示させるモデルの精度は, 非常に重要である.

本稿では, 上述の問題を解決する新たな三次元形状復元手法を検討する. 新たな手法には, 後述の Bastian らのシステム [6] を参考にし, 機能の追加を行った. 追加された機能に関しては 3 節で述べる. 物体形状復元には, カメラ映像からの色情報を用いて, 復元したい物体の領域 (以下, 前景) と復元したい物体以外の領域 (以下, 背景) の領域分割を行う. そして, 前景のシルエットとカメラの自己位置姿勢推定の結果から視体積交差法によりボリュームデータを生成する. 領域分割では, マウスからのストローク入力によって前景を指定する. 生成されたボリュームデータに対して, 入力画像をテクスチャとしてマッピングすることでモデルを生成する. 本稿の構成は以下の通りである. 次節では AR ジオラマの説明を

行う。3節では精度の良い三次元物体形状復元手法を検討し、4節では復元に必要な領域分割と形状データ生成手法について述べる。5節で4節で示した内容の実装方法について述べ、6節で動作実験の結果を示し、7節において、本稿のまとめを記す。

2 AR ジオラマとは

本節では、AR ジオラマの目的、従来のAR ジオラマの内容と問題を述べる。AR ジオラマの目的は、簡単なインタフェースでユーザの身近な環境をAR 環境として構築・操作を可能にすることである。そのため、システムに用いるインタフェースや入力デバイスなどは容易な入力動作を必要とする。例えば、キーボードを用いずにタッチパネルを用いて画面に直接ストロークを行うことで、直感的な操作をする。また、AR ジオラマは事前の準備の必要なしに様々な環境で利用できることを想定している。従って、環境中にマーカを要するマーカベースのトラッキング手法は適さず、マーカレスで動作する必要がある。

2.1 AR ジオラマの利用方法

次にAR ジオラマがどのように利用されるかを述べる。AR ジオラマのコンセプトは、実環境から物体形状を復元、AR 環境を簡単に構築、タンジブルインタフェースを用いて実物体とバーチャル物体を自由に操ることができる環境を提供することにある。上記の機能を生かして、以下の利用例が考えられる。

- 災害シミュレーション

町のジオラマを生成し、洪水や火災などのシミュレーションを行う。通常の模型は全て実物体であるので、町の外観が変わる（工事など）時に、模型を作り直さなければいけない。逆に、すべてバーチャルのジオラマでは、リアリティに欠けてしまう。そこで、リアルとバーチャルな物体を混合させることで、変更の容易さ、物体の写実性などを生かしたジオラマを作り、シミュレーションに役立てる。

- インテリアレイアウトシステム

例えば、部屋の椅子を変えたいと考えた時、家具屋に出向き、その場で気に入った椅子を復元しておく。家に帰り、自分の部屋に椅子を重畳表示することで、椅子が自分の部屋の雰囲気合ってるかなど確認できる。また、バーチャル物体であるので、色やサイズを容易に変更して、部屋をレイアウトできる。

2.2 従来のAR ジオラマの概要

上記の利用例のようなアプリケーションを提供するために、従来のAR ジオラマでは、モデル生成とモデルの動作記述に取り組んできた。従来のAR ジオラマは、SLAM を用いて求まる物体の特徴点から、特徴点群のドロネー三角形分割を行うことでモデルを生成する。

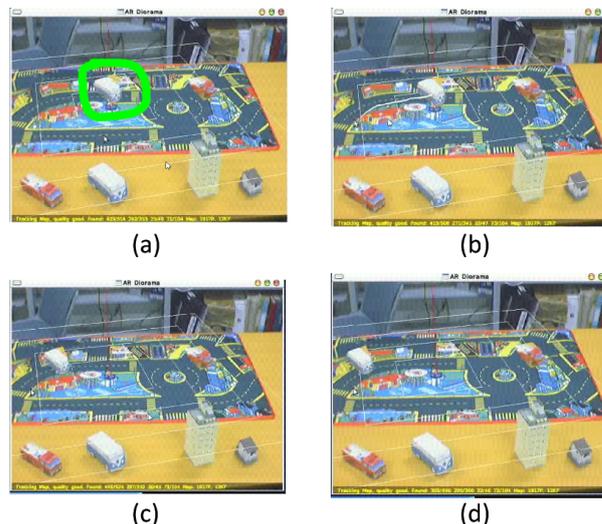


図1 従来のAR ジオラマにおけるストロークを用いたシーン編集[5](a)モデル配置(印内部)(b)動作記述(c)(d)モデルの移動

SLAM とは、カメラの自己位置姿勢と特徴点群等の位置の推定を同時に行う手法である。ユーザからの入力とデバイスからの入力を組み合わせることで、生成されたモデルを実環境に配置し、移動や回転の動作を記述する。図1は移動の動作記述の例である。

2.3 従来のAR ジオラマの問題

従来のAR ジオラマではモデル生成とモデルの動作記述を可能とするAR システムを実現している。しかし、モデルの復元手法に2つの問題がある。1つ目は復元対象の物体に特徴点が存在しない場合、復元が困難もしくは不可能であるという問題である。2つ目はカメラ自己位置姿勢などの誤差により、モデル統合がうまくいかない問題である。上記の問題から、モデルの復元精度が悪い場合がある。モデルの精度がよくない場合、動作記述が正確に行うことができない、物理シミュレーションによる接触判定を正確にできないなどの問題がある。上記の問題を解決するために、本研究では新たな三次元物体形状復元手法を検討する。

3 領域分割に基づく三次元形状復元手法

本節では、本研究で参考したシステムの内容、本システムの三次元形状復元手法の構成について述べる。

3.1 参考システム

本研究では、従来のAR ジオラマを基盤とした上で、Bastian らのシステムの三次元形状復元手法を参考にした。Bastian らのシステムの特徴を以下に示す。

- (1) 色情報を用いた領域分割結果を利用した視体積交差法による物体形状復元
- (2) 単眼カメラとマウス入力による形状復元
- (3) SLAM によるカメラ自己位置姿勢推定

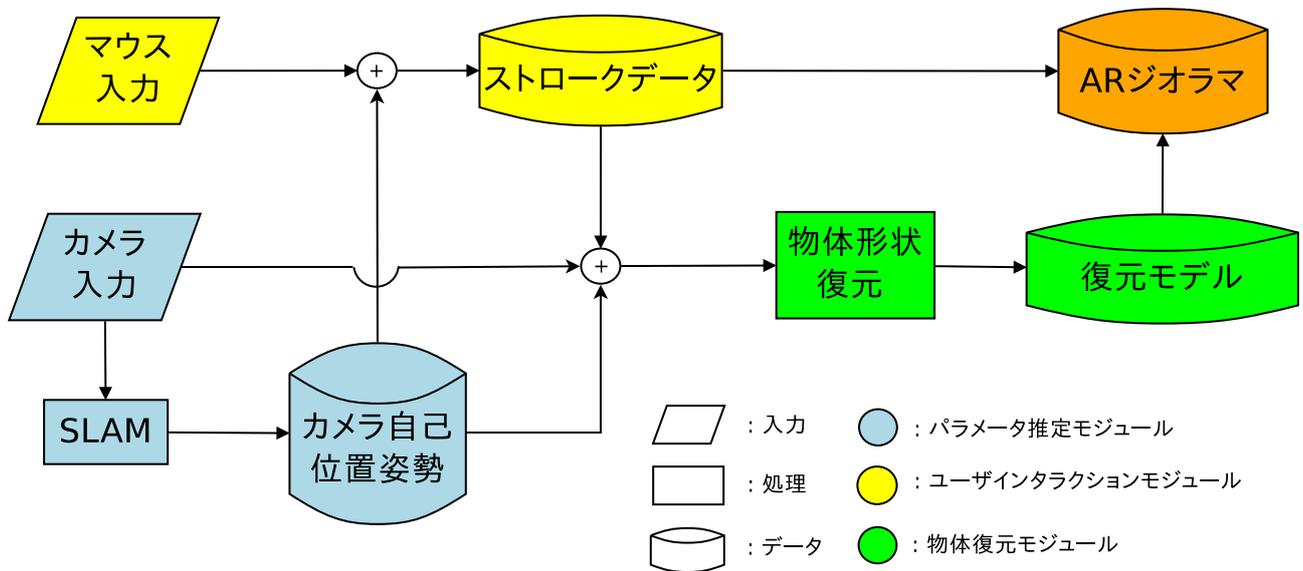


図2 AR ジオラマの構成

(4) 実環境に配置されたバーチャル物体の拡大縮小、移動、回転、オクルージョン処理。

本システムは、従来のARジオラマをベースに開発する。従来のARジオラマでも、カメラトラッキングにはSLAMを用い、インターフェースは単眼カメラとマウスストロークのみである。これらの類似点から、他のシステムと比べ、BastianらのシステムはARジオラマへの移植性が高いと言える。

また、従来のARジオラマでは、物体のテクスチャから得られる特徴点情報のみを用いて、モデルを復元する。そのため、特徴点が少ない物体(e.g. 風船のような特徴点を得づらい物体)の復元は困難である。そこで、色情報を用いて三次元形状復元を行うことで、特徴点情報に依存することなくモデルの復元が可能である。

以上の内容を鑑みて、本研究ではBastianらの手法を参考にARジオラマのモデル復元手法の構築を行う。

3.2 システム構成

ARジオラマ全体の構成図を図2に示す。システムは、以下の3つのモジュールから構成されている。カメラパラメータを推定するモジュール、マウスを用いたユーザインタラクションモジュール、カメラ映像とマウスストロークから三次元物体を抽出する物体復元モジュールである。本研究では、物体の三次元形状モデルの復元が目的である物体復元モジュールの実装を中心に行う。

パラメータ推定モジュールでは、SLAMによるカメラの自己位置姿勢推定を行う。

ユーザインタラクションモジュールでは、ユーザがマウスストロークによって復元する物体を指定する。また、物体復元モジュールの復元結果が誤っている場合は、ユーザが動的にインタラクションを加えることで修正を行うことができる。ユーザが行う修正とは、現時点における前景領域に対して、新たに領域を追加する処

理と不要な領域を削除する処理である。本システムでは、領域を追加するストロークを“Inclusion brush”，領域を削除するストロークを“Exclusion brush”と呼ぶ。Inclusion brush, Exclusion brushは物体復元モジュールの処理に用いられる。本システムでは、Inclusion brushとExclusion brushの切り替えなどをボタンを使わずに、ストロークによる文字入力を介して行う。このモジュールでは、ストローク結果から得られた文字を認識し、認識結果を他のモジュールに渡すことで、システムの動作を制御する。

物体復元モジュールでは、上述したモジュールの処理結果を用いて、復元するモデルを特定し、モデル形状を復元する。復元手法は、視体積交差法を用いる。視体積交差法を用いるには、前景抽出が必要となる。そこで、領域分割を行うグラフカットとユーザによるインタラクションを行うことで前景抽出を実現する。視体積交差法によって生成されたボリュームデータをメッシュデータに変換し、テクスチャマッピングを行い、モデルの復元を実現する。

3.3 Bastianらの手法との比較

本節では、検討手法とBastianらの手法の差分を述べる。図3に、改良後のARジオラマの復元手法の構成を示す。図3に示す3つの要素がそれぞれ追加要素になる。1つ目はモデル復元におけるインタラクションの追加である。このインタラクションは上述のInclusion brushの処理にあたる。2つ目はストローク入力によるジェスチャの認識である。ボタンを用いず、ジェスチャ入力によって、状態遷移を可能にした。3つ目は復元中に失ったボクセルの再構成を行う処理である。2つ目、3つ目の機能は5節で詳しく説明する。

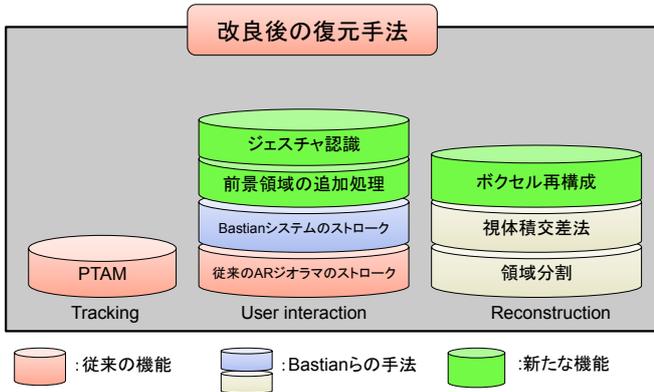


図3 改良後のARジオラマの復元手法の構成

4 物体復元モジュールの構成

本節では、三次元形状復元を行う物体復元モジュールの構成を述べる。

4.1 視体積交差法

本節では、復元したい物体のボリュームデータを生成する視体積交差法の構成について述べる。視体積交差法とは、複数のカメラから得られる多視点の画像から対象物体の三次元形状復元を行なう手法である。まず対象物体を各視点から見たときのシルエットを撮影対象の世界座標系で表される三次元空間（対象空間）に再投影し、視点の位置を頂点、シルエットを断面とする錐体を生成する。これを視体積と呼び、対象物体が存在し得る部分空間を表す。多くの視点からの視体積の共通部分を求めることで、対象モデルの復元を行う。

4.1.1 前景抽出

本研究における前景抽出とは、ユーザが復元したい物体のシルエットを抽出することを指す。グラフカット [7] を用いた領域分割とユーザインタラクションによるシルエットの修正によって構成される。グラフカットによる領域分割は、最初の領域分割とそれ以降の領域分割で異なる式を用いる。ユーザが指定した物体を前景と背景に分割するための最初の領域分割処理を“Initial segmentation”と呼び、分割後に自動でセグメンテーションを行う領域分割処理を“Dynamic segmentation”と呼ぶ。Dynamic segmentation は、グラフカットに距離変換による空間距離画像を用いる。グラフカットによる自動的な領域分割とユーザ自身による手動の領域分割を組み合わせることで、精度の良い前景抽出を行うことを考える。

a) グラフカット

入力画像から図4のようにグラフを作成し、Boykovらの min-cut/max-flow algorithms [8] を用いてグラフを分割する。グラフは、画像の各画素に対応したノードと source と sink と呼ばれるターミナルからなる。source(S) または sink(T) からノードにつながるエッジを t-link と

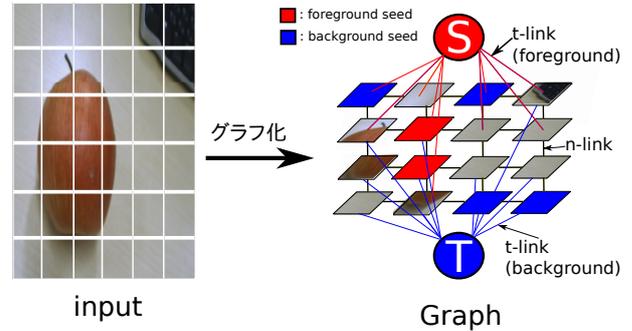


図4 グラフカットにおける画像のグラフ化

呼び、各ノードをつなぐエッジを n-link と呼ぶ。各エッジに与えるコストは式 (1) – 式 (3) で与える。

$$R_f = -\ln \Pr(I_p|O) \quad (1)$$

$$R_b = -\ln \Pr(I_p|B) \quad (2)$$

$$B(p, q) = \exp(-\beta(I_p - I_q)^2) \cdot \frac{1}{\text{dist}(p, q)} \quad (3)$$

O は物体、 B は背景を意味する。 I_p は画素 p の輝度値である。 R_f , R_b は、各画素の t-link (sink, source に繋がるエッジ) に設定する前景と背景の尤度である。 $B(p, q)$ は、各画素間の n-link に用いる。 $\text{dist}(p, q)$ は画素 p , q のユークリッド距離を用いる。

b) ユーザインタラクション

Inclusion brush では領域を追加することができる。Inclusion brush は、マウスストロークとシルエットを結合し、その内部にある領域を追加する。Exclusion brush では領域を除去することができる。Exclusion brush は、シルエットをマウスストロークで分断し、面積が最大となるシルエットの領域のみを保持する。いずれもマウスを用いたユーザインタラクションによって、領域分割結果を修正できる。

4.1.2 パラメータ推定

パラメータ推定とは、カメラの内部および外部パラメータを推定する処理である。抽出した前景シルエットをカメラ映像に再投影するためにカメラパラメータが必要となる。外部パラメータには自己位置姿勢が含まれ、内部パラメータにはカメラの投影行列が含まれる。外部パラメータは SLAM による特徴点群を用いて推定する。内部パラメータはあらかじめキャリブレーションボードを用いてキャリブレーションを行い、推定する。これらはパラメータ推定モジュールで実行される。

4.1.3 ボリュームデータ生成

前景のシルエットとカメラのパラメータを用いて、ボリュームデータを生成する。この処理では、視体積交差法である Space Carving Method を用いて三次元形状復元を行う。

- Space Carving Method(SCM)

このアルゴリズムでは、対象空間の各ボクセルを各カメラの画像平面に投影したとき、全てのシルエットの内

部に投影されるボクセルを対象物体の内部と判定し、一つ以上のシルエットの外部に投影されるボクセルを対象物体の外部の点と判定する。対象空間はすべてのカメラの視野に共通に含まれる直方体領域として設定する。この手法では得られたシルエット画像のカメラの自己位置姿勢が既知である必要がある。シルエット画像とカメラの投影行列を用いることで、ボクセル空間を削っていく。ボクセル空間を削り出す処理を“カービング”と呼ぶ。そして、最終的に残されたボクセル空間が復元すべきモデルとなる。通常のSCMの処理内容は式(4)に従う。

$$v_t^i = v_{t-1}^i f(I_t^\alpha(\mathbf{P}_t W(v^i))) \quad (4)$$

v_t^i は t フレーム目の i 番目のボクセルを指す。 $v_0^i = 1.0$ で初期化を行う。 \mathbf{P}_t は射影行列、 $W(\cdot)$ は世界座標系からカメラ座標系への変換を意味する。 $f(\cdot)$ は変数にラベルを取る。 $f(0)$ ならば1、 $f(B)$ ならば0を返す。またSCMでは一度ボクセルを削ると、そのボクセルは常に空になる。よって自己位置姿勢が安定していない場合、残すべきボクセルが削られてしまう可能性がある。そこで、複数視点からの処理結果の $vote$ を取ることで、自己位置姿勢推定のエラーによる削りすぎを防ぐ。このときの処理内容は式(5)に従う。

$$v_t^i = \text{sgn}(v_{t-1}^i) \left(v_{t-1}^i + \frac{2f(I_t^\alpha(\mathbf{P}_t W(v^i))) - 1}{n} \right) \quad (5)$$

$\text{sgn}(\cdot)$ は符号関数であり、ボクセル値が正(負)なら1(0)を返し、ボクセル値が0(空のボクセル)ならば0を返す。また、パラメータ n はSCMによってボクセルが削られるまでの $vote$ 数を表す。

4.2 形状復元データ生成

物体の三次元形状はサーフェスレンダリングによって復元できる。しかし、三次元形状のみではどのモデルを復元したかわかりにくい。そのため、テクスチャをはりつけることで形状復元データを生成する。また、復元された物体に対して、単一視点からのマッピングでは側面などに正しいテクスチャがはられない。そこで、複数視点から得られるテクスチャをモデルに復元することで、生成された三次元形状に対して、正しいテクスチャをはりつけることを考える。

5 システム実装

本節では、本研究の復元手法の実装に関して述べる。提案システムでは、高速な処理が必要となる。そこで、複数のモジュールをスレッド化し、マルチスレッドプログラミングを行う。パラメータ推定モジュールをトラッキングスレッドとマッピングスレッドの2つのスレッド、ユーザインタラクションモジュールを1つのスレッド、物体復元モジュールを前景抽出スレッドとボリュームデー

タ生成スレッドの2つのスレッド、合計5つのスレッドで構成する。

5.1 パラメータ推定モジュール

本研究では、カメラの自己位置姿勢推定を行うにあたり、PTAMを利用した。実装には、公開されているライブラリ[12]を利用した。

5.2 ユーザインタラクションモジュール

画面中のストロークを文字として扱い、文字認識を行うために、Tesseract-OCR[13]を用いた。ストローク入力をジェスチャとして解析し、Tesseract-OCRにデータを渡し、Tesseract-OCRから得られた結果をシステムに入力することで、Inclusion brushとExclusion brushの切り替え、SCMのスタートなどを実現した。

5.3 物体復元モジュール

5.3.1 グラフカット

グラフカットを適用するためには、ユーザによる前景指定とグラフモデルが必要となる。前景指定のデータは、ユーザインタラクションモジュールから渡される。グラフモデルは、画像の画素の輝度値と色モデルを用いて生成される。色モデルは複数考えることができるが、本研究ではGaussian Mixture Model(GMM)[10]に当てはめる。GMMの推定にはEMアルゴリズム[11]を用いる。

- 色モデルに基づくエッジコスト

本研究では、画像グラフに与えるエッジコストはGMMのパラメータを用いて、式(6)–(7)と定義する。

$$\Pr(I_p|\cdot) = \sum_{i=1}^K \alpha_i p_i(I_p|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \quad (6)$$

$$p(I_p|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{3/2} |\boldsymbol{\Sigma}|^{1/2}} \cdot \exp \left(-\frac{1}{2} (I_p - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (I_p - \boldsymbol{\mu}) \right) \quad (7)$$

t-linkとn-linkの間の重み係数 $\lambda=0.05$ とする。 λ が大きすぎるとt-linkの影響が大きくなり、色情報の影響が大きすぎる。 λ が小さすぎるとn-linkの影響が大きくなり、隣接画素の輝度値の影響が大きすぎる。それぞれの影響が出すぎないように、適切な値を試行錯誤して設定した。

- Dynamic segmentation

Dynamic segmentationの処理内容では、画像全体へグラフカットを適用するのではなく、前景とその周辺のみ 픽셀に対して行うことで高速化を図る。また式(13)–(14)を用いることで、尤度に重みをつける(式(8)–(11))。

$$R_f = -\ln \Pr(O|I_p) \quad (8)$$

$$R_b = -\ln \Pr(B|I_p) \quad (9)$$

$$\Pr(O|I_p) = \Pr(O) \Pr(I_p|O) \quad (10)$$

$$\Pr(B|I_p) = \Pr(B) \Pr(I_p|B) \quad (11)$$

– 距離変換

距離変換による確率分布生成では、境界からの距離 d を 0.5 から 1 に正規化する。正規化を行う際に、正規分布における累積分布関数 (式 (12)) を距離変換画像に適用し、確率分布 (式 (13)–(14)) を形成する。

$$d = \frac{1}{2} \left(1 + \operatorname{erf} \frac{x - \mu}{\sqrt{2}\sigma} \right) \quad (12)$$

$$\Pr(O) = \begin{cases} d_{fore} & \text{if } d_{fore} \geq d_{back} \\ 1 - d_{back} & \text{otherwise} \end{cases} \quad (13)$$

$$\Pr(B) = 1 - \Pr(O) \quad (14)$$

5.3.2 ボリュームデータ生成

SCM を実行するために、カービングを行う視点のカメラの自己位置姿勢と前景シルエットが必要になる。キーフレームは、複数回の Dynamic segmentation の実行が行われた後に取り出される。キーフレームのカメラの自己位置姿勢と前景シルエットを用いてカービングを行う。カービングには PTAM の推定誤差も含めた式 (5) を用いる。式 (5) のパラメータ n は、実際に動作実験を行った結果から $n = 8$ に設定した。新機能のボクセル再構成では、領域分割によって得られた前景領域内のピクセルのうちボリュームデータの投影結果と重ならない部分のみを実空間に逆投影し、既にカービングされたボクセルと交差した箇所を復元する。

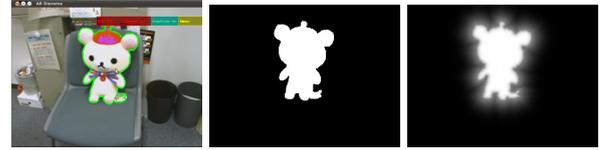
5.3.3 形状復元データ生成

ボリュームデータをサーフェスレンダリングの 1 つである Marching Cubes 法 (MC 法) を用いてメッシュにする。MC 法を用いて得られたメッシュに最も近いカメラ位置にあるテクスチャを各メッシュにマッピングする。

6 動作実験

本節では、実装した物体復元モジュールの動作実験に関して述べる。動作実験に 3.40GHz Intel(R)Core(TM) i7-2600, nVidia GeForce GTX 460, 8GB メモリを使用した。画像キャプチャには PointGrey Flea3, 解像度 648x480 に画角 84.9x68.9° の 3.5mm レンズを装着した。

各処理における実行時間を表 1 に示す。三次元形状復元に必要な各ステージの処理時間の合計は約 1.5 秒である。ただし、5 節で説明したように、マルチスレッド処理を行うので、ユーザが感じる遅延は少なく、前景抽出は毎秒 10 回程度で動作し、形状復元は毎秒 5 回程度で動作している。前処理のステージでは、前フレームの領域分割結果に対するラベル割り振り、距離変換などを行った。Dynamic segmentation の処理において、カメラ映



(a) Dynamic segmentation に用いる確率分布生成



(b) Inclusion brush



(c) Exclusion brush

図 5 前景抽出処理

像中の前景の面積が背景の面積よりも大きい場合が多い。そのため、前景と背景の色モデルの更新において、前景の更新の方が処理に時間がかかっている。MC 法の処理は、同時に複数視点からのテクスチャマッピングも行うために、処理に時間がかかっている。

6.1 前景抽出処理

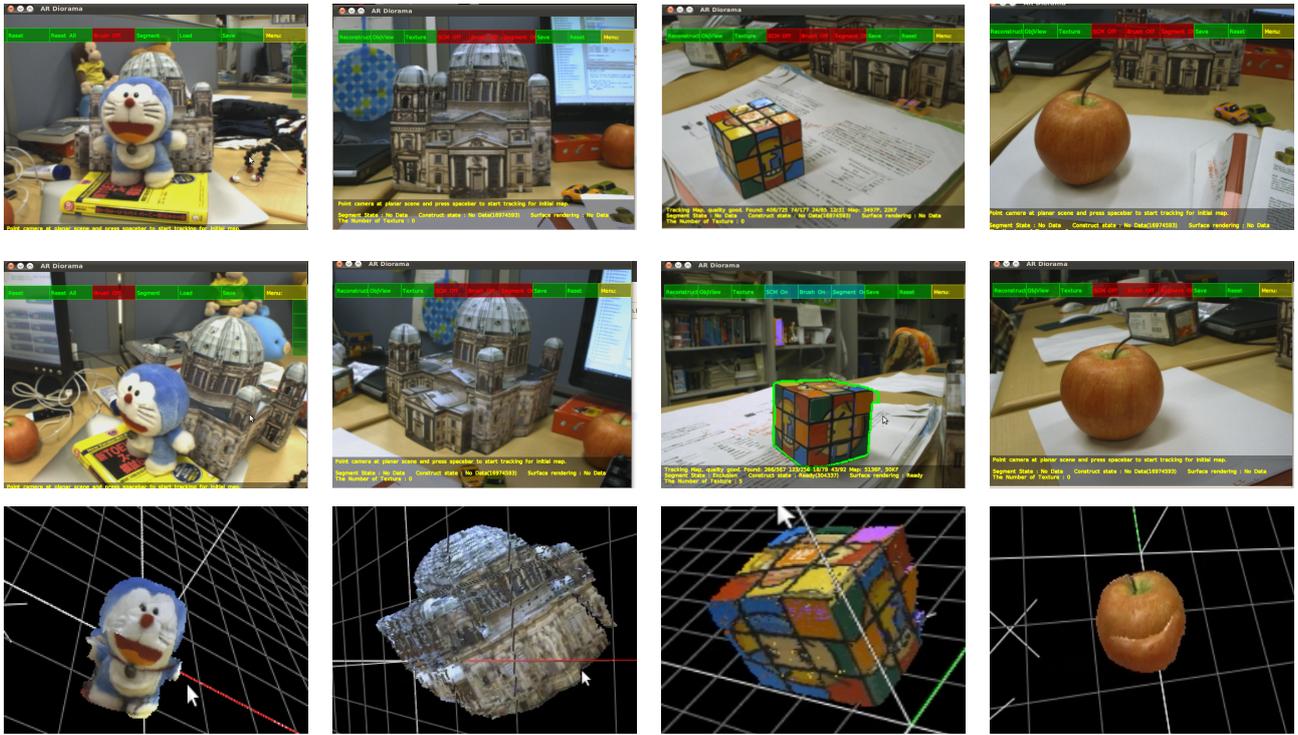
視体積交差法に用いる前景抽出処理の結果のうち、Dynamic segmentation の距離変換結果と領域分割結果、ユーザの Inclusion brush, Exclusion brush による領域分割結果の修正の処理結果を示す。

図 5 に前景抽出処理結果を示す。前景の中心の確率分布は一様であるのに対し、エッジ付近に近づくに従って、前景である確率が急激に小さくなる確率分布を形成することができている。図 5(b) から、領域分割におけるユーザインタラクションの Inclusion brush を行うことで、前景と見なされていない領域を前景として追加、また、図 5(c) から、Exclusion brush を行うことで、不要な領域を削除、これらの結果から領域分割の精度が悪い場合でも、ユーザが簡単に結果を修正できることを確認した。

6.2 三次元物体形状復元

視体積交差法で生成されたボリュームデータを MC 法

処理内容	処理時間 [sec]
前景抽出の前処理	0.1690
前景の色モデル更新	0.1012
背景の色モデル更新	0.0450
グラフカット	0.03214
SCM	0.2405
MC 法	0.940



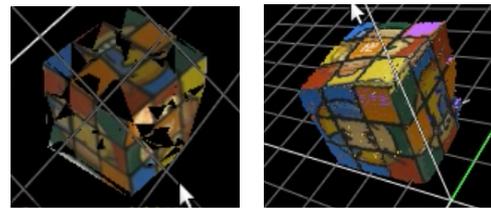
(a) (b) (c) (d)

図6 三次元物体形状復元 (上段・中段:テクスチャとして利用したキーフレーム, 下段; モデル) (a) 人形 (b) 紙の宮殿 (c) ルービックキューブ (d) リンゴ

によってメッシュデータに変換し、テクスチャマッピングを行うことで、モデルが生成できる。実環境の物体の形状を復元した様子を図6に示す。しかし、視体積交差法による削り残しの問題とテクスチャに用いる画像に輝度差が存在する問題のため、図6では模様の不整合が発生している場合がある。また、本手法には復元困難な物体が存在する。これは、どの視点から見た場合でも領域分割したシルエットにカービングされるべき部分が含まれていて、シルエット内部の領域がカービングされないような物体を復元する場合である。そのため、図6(b)の例では宮殿の前方の不要な領域が残ってしまっている。視体積交差法の問題であるので、本手法では凹物体の復元が困難である。この問題を解決するためには、photoconsistency(色の一貫性)やテクスチャマッチングなどを行い、シルエット情報だけでなくシルエット内部の色情報を用いる、特徴点ベースの手法と組み合わせて復元を行うなどの手法を今後検討する。

6.3 従来手法と本手法の復元結果比較

最後に、従来のARジオラマで用いられた特徴点ベースの復元手法と、本研究の領域ベースの復元手法を比較する。比較物体は図6(c), 図6(d)の2つの物体とする。図6(c)のキューブには、特徴点が多く存在する。そのため、図7(a)のように、従来手法を用いた場合でもおおよその形状を復元できる。しかし、ドロネー三角形分割の誤差から必要なメッシュも削除しているためモデルに欠



(a) 従来手法 (b) 本手法

図7 図6(c)の復元結果

損が生じている。本手法を用いた場合、メッシュに欠損を作ることなく物体形状が復元できていることが確認できる。しかし、モデルのエッジ部分が少し削り取られて完全なキューブの形状をなさない場合がある。この問題は復元処理中のカメラをゆっくり移動させることで解決できる。

図6(d)のリンゴには特徴点が多く存在しなかった。そのため、従来手法では物体の復元不可能の結果に終わった。本手法では、図6(d)の下図に示す結果が得られた。

従来手法と本手法の比較を行った結果、従来手法では復元できない物体を本手法で復元できることを確認した。また、特徴点が多い物体の場合についても、従来手法ではメッシュの欠損が発生する問題がある。これに対して、本手法ではメッシュの欠損を発生させることなく形状を復元した。

以上の結果から、本手法はテクスチャ依存性が低い利点を生かした復元が可能であることを確認した。

7 結 論

本研究では、立石らのシステムをベースとして、Bastianらのシステムを組み合わせることで、立石らのシステムにおける実物体の三次元形状復元精度を向上させることを検討した。

本研究手法では、従来システムの特徴点を用いた復元手法ではなく、領域分割を用いた手法を用いた。色情報に基づく領域分割を行うことで、多視点からのシルエット画像を取得できる。そして、カメラ自己位置姿勢推定結果とシルエット画像から、視体積交差法によってボクセルデータとしての形状を復元する。最後に、得られたボクセルデータに対して、MC法とテクスチャマッピングを行うことで三次元形状モデルを生成した。

復元結果を見ると、領域分割の精度が高くない部分がある。また、処理速度の問題によりカメラをゆっくり動かさなければいけない。そして、領域分割の精度はカービング精度にも影響を与えるため、高精度なモデルを復元できない場合がある。しかし、ユーザがカメラをゆっくり動かし、領域分割がうまく動作するように、復元対象の物体の周囲をユーザが移動することで、高精度な、写実性の高い三次元形状モデルを生成できることを確認した。

今後の課題としては、領域分割の前処理の手法の再検討、処理フローの改善をすることで、処理速度の高速化と領域分割の精度向上を図る。また、生成されるモデルの精度の向上のために、GPU Shaderを用いることで、高速なモデル生成を検討する。領域分割によるモデル生成完了後、システムのアプリケーション部分を考える。本システムは、将来的にモバイル端末(iPadなど)を用いてAR環境を構築することを考えている。モバイル端末上で、ユーザインタラクション(つまむなど)によって、モデルの形状や色などを自在に変更できるIn-situ mobile interactive applicationの構築を目指す。

文 献

- [1] H. Kato and M. Billinghurst: "Tracking and HMD Calibration for A Video-based Augmented Reality Conferencing System," Proc. of the 2nd International Workshop on Augmented Reality, pp. 85-94, 1999.
- [2] G. Klein and D. Murray: "Parallel Tracking and Mapping for Small AR Workspaces," Proc. of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 1-10, 2007.
- [3] Y. Yemez and C.J. Wetherilt: "A Volumetric Fusion Technique for Surface Reconstruction from Silhouettes and Range Data," Computer Vision and Image Understanding, Vol. 105(1), pp.30-41, 2007.
- [4] M. Brown, T. Drummond, and R. Cipolla: "3D Model Acquisition by Tracking 2D Wireframes," Proc. of the 11th British Machine Vision Conference, pp. 11-14, 2000.
- [5] 立石 昂裕: "単眼カメラとストローク入力を用いた三次元復元によるARジオラマシステム", 大阪大学 大学院

情報科学研究科 修士学位論文, 2010.

- [6] J. Bastian, B. Ward, R. Hill, A. Hengel and A. Dick: "Interactive Modelling for AR Applications," Proc. of the 9th IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 199-205, 2010.
- [7] 石川博: "グラフカット(チュートリアル)", 情報処理学会研究報告. CVIM, pp. 193-204, 2007.
- [8] Y. Boykov and V. Kolmogorov: "An Experimental Comparison of Min-cut/max-flow Algorithms for Energy Minimization in Vision," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26(9), pp. 1124-1137, 2004.
- [9] 永橋和行: "グラフカットによる領域セグメンテーションに関する研究", 中部大学大学院工学研究科 修士学位論文, 2007.
- [10] A. P. Dempster, N. M. Laird, and D. B. Rubin: "Maximumlikelihood From Incomplete Data via the EM Algorithm," Journal of the Royal Statistical Society. Series B, Vol. 39, No. 1, pp. 1-38, 1977.
- [11] C. Stauffer and W.E.L. Grimson: "Adaptive Background Mixture Models for Real-time Tracking," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 246-252, 1999.
- [12] Georg Klein Home Page: "Parallel Tracking and Mapping for Small AR Workspaces," <http://www.robots.ox.ac.uk/gk/PTAM/>. (最終閲覧日:2011年2月20日)
- [13] Google Project Hosting: "tesseract-ocr," <http://code.google.com/p/tesseract-ocr/>. (最終閲覧日:2011年4月13日)