

改良された space carving 法による形状の統合と穴埋め — アルゴリズムと GPU による実装 —

古川 亮[†] 板野 友哉[†] 森栄 晃彦[†] 川崎 洋^{††}

[†] 広島市立大学大学院 情報科学研究科 知能工学専攻 ^{††} 埼玉大学 工学部情報システム工学科

要旨

複数の距離画像を統合し、単一のポリゴンメッシュを得る処理や、統合の際に計測データの欠損などによって生じる穴を補間する処理は、モデリングにおいて重要な処理である。これらの処理について従来提案されているものは、距離画像の数が少ない場合にうまく働かないものが多かった。本論文では、space carving 法を改良した補間アルゴリズムを提案する。提案手法では、符号付距離場において、観測されたボクセルばかりでなく、未観測のボクセルについても、物体の内側か外側かを、ベイズ推定に基づく判別関数によって判定することで、距離画像が少ない場合においても安定した結果を得ることができる。また、提案手法は、GPU による実装が可能であり、効率的に実行することができる。

Integration and hole-filling of shapes using improved space carving — Algorithm and GPU-based implementation —

Ryo Furukawa[†], Tomoya Itano[†], Akihiko Morisaka[†], Hiroshi Kawasaki^{††}

[†] Faculty of information sciences, Hiroshima City University, ^{††} Faculty of engineering, Saitama University

Abstract

Integrating multiple range images to obtain a unified shape model, and filling holes of the shape models caused by data losses are important tasks for modeling for CG. Many of the conventional techniques for integration and hole-filling often produces unstable results if the number of range images is small. In this paper, we propose a new method for these tasks by improving “space carving.” In the proposed method, all the voxels of the signed distance field are classified into inside or outside of objects using a discriminant function based on Bayes estimation. The proposed method is stable. In addition, the algorithm can be implemented on GPU and executed efficiently.

1 はじめに

三次元計測によって得られる形状データは、距離画像の形式になっていることが多い。複数の計測結果から、全周の形状を得るには、これらの距離画像を位置合わせし、統合する必要がある。複数の距離画像を統合する手法として、符号付距離場を用いる方法は、大きな入力データを効率的に処理可能であり、広く研究されるようになってきている。符号付距離場による形状表現は、物体表面の未観測部分などに発生する欠落部分を補間するためにも利用される。Curless と Levoy は、ボリューム空間の各ボクセルを Unseen (未観測)、Near surface (物体表面)、Empty (物

体外部) に場合分けし、Unseen と Empty との間にメッシュを生成することにより欠落部分の補間を行なった [1]。これは、ボリュームデータから、観測された表面の前面を削っていく処理に相当することから、space carving 法 (以下 SC 法) と呼ばれる。SC 法は、観測データが多い場合には、効果的に欠落部分の補間を行うことができるが、利用可能な距離画像が少ない場合には、失敗することが多い。これは、space carving によって削られるボリューム空間の「削り残り」と、対象物体とが連結されてしまうことにより生じる。

このような問題を解決するには、未観測のボクセルを無

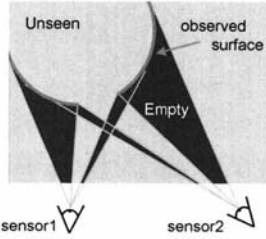


図 1: Space carving(SC) 法による補間のイメージ

くし、すべてのボクセルを「物体内部」と「物体表面」に分類することが考えられる。この時、未観測の部分には、物体の無い部分、物体内部であってもオクルージョンによって観測できない部分、物体表面の反射率が低いことなどによるデータ欠損が含まれるので、これらを考慮して分類を行う必要がある。本論文では、SC法において未観測とのみ定義されるボクセルを、さらに「物体内部」と「物体表面」に分類するアプローチにより物体の統合及び補間の精度向上を実現する手法を提案する。これを実現するために以下2つのアプローチをとる。

- (1) ベイズ推定法に基づいた、未観測ボクセルを分類するための判別関数の定義。
- (2) 計測結果がアクティブステレオ法によるものである場合、光源位置情報の利用。

これらのアプローチは、(1)だけでも利用可能であるが、両方併用することで効果を増す。提案手法では、SC法に見られるような、物体の外の空間の大きな「削り残し」は発生しない。また、すべてのボクセルは、「物体内部」「物体外部」「物体表面」に分類されるため、必ず閉じた形状が得られる。

また、本論文において、ボクセルの分類に用いられる評価式は、GPUによって実装することが可能であり、処理効率を大幅に高めることが出来る。本論文では、このような実装の方法についても述べる。

2 関連研究

符号付距離場の等値面として表された形状データの欠落部分の補間に関しては、これまでいくつかの手法が提案されている。Davisらは、符号付距離場のフィルタリングによって形状の欠損部を補間する手法を提案した[2]。後の実験で示すように、この手法は、等値面の拡散が正しくない方向に起こり、望ましい結果が得られない場合がある。特に、形状の欠落部分が、観測方向に対する自己オクルージョンによって生じている場合、補間の失敗が起こりやすい(図5(b))。

SC法[1]では、まず全てのボクセルをUnseen(未観測)の状態に初期化し、各距離画像から参照される物体の表面と視点位置の間に存在するボクセルをEmpty(物体外)であるとして「削る」この手法では、物体表面として観測された部分の手前の部分しかボリュウム空間を削らないので、実際には物体の外部であるボクセルがUnseen(未観測)として残存し、その部分とEmptyの領域との境界に余分なメッシュが発生する。十分な数の視点位置からの距離画像を統合する場合には、このような余分なメッシュは対象物体のメッシュと連結せず、容易に取り除くことが出来る。しかし、視点の数が少ない場合や、物体の特定の方向からの観測が行われない場合には、余分なメッシュが物体と連結してしまうことが多い(図5(a))。このことは、多くの応用において問題となる。

また、佐川らは、符号付距離場の各ボクセルについて周囲のボクセルとの整合性をとることで、複雑な形状のシーンの補間を行う手法を提案した[3]。この方法では、ユークリッド距離による符号付距離場が利用されており、高い品質の穴埋めが可能であるが、計算コストが高い。

近年、急速にその性能が向上しているGPUの演算性能を、グラフィックス以外の汎用計算に用いる手法が数多く提案されている。2次元輪郭からの3次元再構成にもSC法が用いられるが、Nitschkeらは、この手法をGPUで実装する手法を提案した[4]。また、SudらはGPUを用いて、物体モデルからの距離場を高速に計算する方法を示し、物体同士の距離の判定に用いることを提案した[5]。

3 未観測ボクセルのクラスの推定を用いた複数形状の統合と補間

3.1 概要

符号付距離場は、三次元空間の各ボクセルについて、そのボクセルから物体表面までの距離をボクセル値の絶対値で、そのボクセルが物体外部/内部のいずれであるかをボクセル値の符号で表したスカラー場である(本論文では、内部を負、外部を正とする)。符号付距離場を $D(x)$ と表すと、物体表面は $D(x) = 0$ を満たす等値面で表すことができる。そのボクセルから物体表面までの距離には、ユークリッド距離(ボクセル中心から表面までの最短距離)を使うことが考えられるが、ユークリッド距離による符号付距離場を作成することは計算量が多いため、簡易的な手法として、視線距離(計測における視点位置からの視線上での距離)なども用いられる。視線距離計算は一般のグラフィックハードウェアで実装されており非常に高速に処理できることと、提案手法により精度面の問題を解消できることから、本研究では、視線距離による符号付距離場を用いる。

複数の距離画像を統合して符号付距離場を構築する手法はいくつか提案されている。あるボクセル v の符号付距離 $D(v)$ は、各シーンの距離画像から得られた符号付距離 $d_1(v), d_2(v), \dots, d_n(v)$ にそれぞれの重み $w_i(v)$ を乗じたものを累積して、以下の式により求められる。

$$D(v) = \sum_i d_i(v)w_i(v) \quad (1)$$

重み $w_i(v)$ は各距離値の精度を考慮して決定される。この値を、カメラからの視線方向と物体表面の法線方向のなす角によって決める場合もある。また、 $d_i(v)$ の絶対値がある閾値より大きい場合、そのボクセルは対象となる距離画像の表面からは遠いので、累積の計算には加えないことが多い。

構築した符号付距離場 $D(x)$ においては、 $D(x) = 0$ を満たす等値面が、統合された形状を表している。統合結果をメッシュモデルとして取り出すためには、マーチングキューブス法を用いる。

CurlessらのSC法[1]では、ボリュウム空間の全てのボクセルをUnseen(未観測)、Empty(物体外部)、NearSurface(表面付近)の三つのいずれかの状態に場合分けする。そして、UnseenとEmptyの境界にメッシュを生成することで形状の補間を行う(図1)。この方法の問題点は、削られなかったUnseenのボクセルには、(1)物体の外部であって、どの距離画像においても観測値のある点の視線上に存在しないボクセル、(2)物体の外部であって、ある距離画像において観測値のある点の視線上に存在するが、オクルージョンにより物体外部と判定されないボクセル、(3)物体の内部、の3つの場合が存在するのにも、これらを区別しな

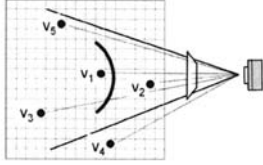


図 2: ボクセルのクラス

いことにある。提案手法では、Unseen のボクセルについて、物体の外部、内部を判定するための判別関数を定義することでこの問題を解消する。

以下、提案手法におけるボクセルのクラス分けについて述べる。あるボクセルについて、距離画像から得られる情報には、次の 4 種類がある。(図 2)。

NearSurface(表面付近) 符号付距離の絶対値が指定した閾値以下の場合、対象のボクセルは“表面付近”とクラス分けされ、その符号付距離を保持する(図 2 の v_1)。

Empty(物体外部) 符号付距離の絶対値が閾値より大きくその符号が正の場合、対象のボクセルは物体とカメラの間に存在し、そこは物体の外部であると断定できるため“物体外部”とクラス分けする(図 2 の v_2)。

Occluded(遮蔽領域) 符号付距離の絶対値が閾値より大きくその符号が負の場合は、一概に外部/内部の断定をすることはできない。その場合、対象のボクセルのクラスを一時的に Occluded とし、後で物体内部、外部のいずれかであるかを推定する。推定の方法は 3.2 節で述べる(図 2 の v_3)。

NoData(データ欠損) 距離画像の欠損部分のため符号付距離が得られないとき、外部/内部の判断をすることはできない。この場合、対象のボクセルのクラスを一時的に NoData とし、後で物体内部、外部のいずれかであるかを推定する(図 2 の v_5)。

図 2 の v_4 のように、対象となるボクセルが距離画像の画角の外にある場合、上記の分類のうち、Empty か NoData として取り扱う。このうちのどちらにするかは、アプリケーションによる。多くのアプリケーションでは、距離画像の画角外は Empty としてよいが、大きな物体をズームして計測する場合などは、画角外は NoData として取り扱う。

複数の距離画像を統合する際、あるボクセルについて、それぞれの距離画像から得られる情報は異なる。この時、NearSurface となる距離画像が一つでもある場合、ボクセルのクラスも NearSurface とする。そうでない場合、Empty となる距離画像が一つでもあればボクセルのクラスも Empty とする。距離画像から得られる情報が、Occluded 及び NoData しかない場合には、3.2 節で述べる判別関数によってそのボクセルが物体の内部であるか外部であるかを推定する。物体の外部であると判定されたボクセルには、Empty のクラスを、内部であると判定されたボクセルには Solid(物体内部)のクラスを割り当てる。

上記のクラス分けを、すべてのボクセルについて行くと、ボリューム内の領域は Solid(物体内部)、NearSurface(表面付近)、Empty(物体外部)の 3 種類に分類される。符号付距離場のボリュームデータとして、NearSurface の領域には通常の符号付距離場を、Solid の領域には負の固定値を、Empty の領域には正の固定値を割り当てる。正負の固定値の絶対値は等しくする。生成された符号付距離場の等値面を生成することで、統合されたポリゴンメッシュが得られる。

3.2 ボクセルのクラスの推定

あるボクセルについて、距離画像から得られる情報が、Occluded 及び NoData のみの場合、物体の外部であるか内部であるかの推定を行う。そのために、正規分布に関するベイズ推定にもとづいて、判別関数を定義する。

物体の外側で正、内側で負となるようなスカラー値を考え、このスカラー値の予測分布をベイズ推定によって決定することを考える。データが無い段階での予測分布(主観確率の分布)は一様分布であるとし、各距離画像から計算されるデータから得られる観測値によって、ベイズの定理に基づいて予測分布を変化させる。正規分布に関するベイズ推定では、観測値のモデル分布を、平均値 μ_h が観測値に等しく、標準偏差 σ_h が観測値の精度の逆数になるような正規分布 $N(\mu_h, \sigma_h)$ として表すことが多い。本研究でもこのような方法によって、ボクセルが物体の内側あるいは外側のどちらにあるかに関する heuristics を表現する。具体的には、 μ_h の符号によって、heuristics による物体の内外についての推定を、 $1/\sigma_h$ によってその heuristics の確信度を表すことにする。

対象ボクセルが、ある距離画像について Occluded である場合、そのボクセルは観測位置から見て表面の後方にある。ボクセルの表面からの距離の絶対値 ($Dist$ で表す) が比較的小さい場合、このボクセルは物体内部にある可能性が高い。逆に表面から遠くなる ($Dist$ が大きい) と、物体内部にある可能性は低くなる。そこで、対象ボクセルが物体内部にある確信度を $1/Dist$ 程度であると考え、heuristics に対応する分布を $N(-1, Dist)$ で表現する。

また、対象ボクセルが、ある距離画像について NoData である場合、そのボクセルは物体の外である場合と、物体の内側でありながら、物体表面が何らかの理由で観測されなかった場合とがある。実際のデータにおいては、距離画像上の画素においてデータが存在しない領域は、物体が存在しないことが多い。この heuristics から、ボクセルは物体外にあり、その確信度は定数程度であると考え、heuristics に対応する分布を $N(1, Const)$ で表現する。ここで、 $Const$ はユーザが決めるパラメータである。実験によれば、この $Const$ は物体の最小の厚み程度に設定すると良いようである。

正規分布に関するベイズ推定においては、観測で得られた heuristics に対応する分布 $N(\mu_h, \sigma_h)$ により、予測分布 $N(\mu, \sigma)$ は以下のように更新される。

$$\mu \leftarrow \frac{\sigma_h \mu + \sigma \mu_h}{\sigma + \sigma_h}, \quad \sigma \leftarrow \frac{1}{\frac{1}{\sigma} + \frac{1}{\sigma_h}} \quad (2)$$

これを全ての距離画像について行い、最終的な予測分布の平均値 μ の符号が正であれば対象のボクセルを“物体外部”に、負であれば“物体内部”にクラス分けする。 μ_h が 1 あるいは -1 のどちらかであることを考慮して計算を行なうと、最終的な μ の具体的な形が計算できるので、これをボクセル v についての判別関数 $C(v)$ と見なし、 $C(v)$ の符号で物体の内外を判定することにする。具体的な $C(v)$ は、以下のような式になる。

$$c_i(v) = \begin{cases} -1/|d_i(v)| & \text{if } Occ(i, v) \\ 1/Const & \text{if } Nod(i, v) \end{cases}, \quad (3)$$

$$C(v) = \sum_i c_i(v) \quad (4)$$

ただし、 $Occ(i, v), Nod(i, v)$ は、ボクセル v について距離画像 i から得られる情報がそれぞれ Occluded, Nodata であるかどうかを、 $d_i(v)$ はボクセルと距離画像 i 表面との符号付視線距離を表す。

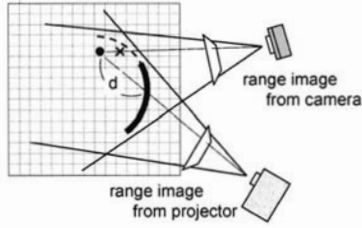


図 3: アクティブステレオ法における光源位置の利用

3.3 アクティブステレオ法における光源位置の利用

アクティブステレオ法は、密な 3 次元形状を得るための代表的な方法の一つである。アクティブステレオ法では、カメラのほかに光源が利用される。このため光源からの Occlusion により、データ欠損が起こる場合があるが、逆にデータが測定されている場合には、その点と光源の間に物体が存在しないことの証拠となる。

そこで本研究では、アクティブステレオ法における光源を視点とする仮想カメラからの距離画像を生成し、これを利用する手法を提案する。あるシーンについてカメラ・光源二つの位置をそれぞれ視点位置とした距離画像（以下、カメラ距離画像・光源距離画像とする）を参照することを考える。図 3 で、光源からの光が遮られ欠損となった部分からは、物体表面までの距離を得ることはできない。しかし光源距離画像を参照することによって、表面からの距離を得ることができる。またそれ以外の部分でも、一つの測定結果からボクセルのクラス分けを NearSurface あるいは Empty にすることができる範囲が広くなり、そうでない場合でも 3.2 節で述べた推定をより適切に行うことが可能となる。

カメラ距離画像の全ての画素について、その 3 次元位置を計算し、これを光源位置に仮想カメラがあると仮定し、投影することで光源距離画像を作成する。

ボクセルの分類は、カメラからの距離画像のみを利用する場合と大体同じであるが、ボクセルのクラス分けの優先順位は、カメラ距離画像での NearSurface、光源距離画像での NearSurface、カメラ距離画像での Empty、光源距離画像での Empty の順とする。また、ボクセル v について物体の内外を判定するための判別関数 $C(v)$ は、式 (4) の関数を拡張し、

$$c_i(v) = \begin{cases} \frac{-1}{|d_c^c(v)|} + \frac{-1}{|d_s^l(v)|} & \text{if } Occ^c(i, v) \wedge Occ^l(i, v) \\ -1/|d_c^c(v)| & \text{if } Occ^c(i, v) \wedge Nod^l(i, v) \\ -1/|d_s^l(v)| & \text{if } Nod^c(i, v) \wedge Occ^l(i, v) \\ 1/Const & \text{if } Nod^c(i, v) \wedge Nod^l(i, v) \end{cases}, \quad (5)$$

$$C(v) = \sum_i c_i(v) \quad (6)$$

とする。ただし、 $Occ, Nod, d_i(v)$ の上付記号 c, l は、それぞれカメラ及び光源距離画像に関する量であることを示す。

4 GPU による符号付距離場及び内外判定判別関数の計算

本論文の統合・補間アルゴリズムは、GPU を利用することで、高速に実行することができる。GPU は二次元画像単位の出力しか行なえないため、符号付き距離場のボリュー

ムをスライス単位で計算する。各スライスをフレームバッファとしてレンダリングすることで符号付き距離値 $D(v)$ および内外判定の判別関数 $C(v)$ を計算する。

レンダリングは、プログラマブルシェーダを用いたマルチパスレンダリングによって行う。各パスは、一つの距離画像についてのレンダリングであり、結果は、レンダリング機能を用いて加算する。距離画像 i についてのレンダリングを行う場合、レンダリングの準備として、 i 番目のカメラ距離画像と、対応する光源距離画像を浮動小数点テクスチャに設定する。そして、フレームバッファ全体に一つの 4 角形を描画する命令を発行することで、各画素についてピクセルシェーダの処理が実行される。ピクセルシェーダは、距離画像とボクセル位置を入力値として $d_i(v)w_i(v), c_i(v)$ を計算し、フレームバッファに加算する。この処理を、カメラ距離画像、光源距離画像を切り替えながら、測定形状の数だけ繰り返す。最後に、符号付距離の重み付和 $D(v)$ と判別関数値 $C(v)$ を持つフレームバッファを CPU 上へ読み戻し、距離値の有無と推定結果を統合し、符号付き距離場のスライスを計算する。

距離画像 i についてのピクセルシェーダでの計算は次のような手順となる。RGBA の 4 チャンネルのフレームバッファを利用し、R に $D(v)$, G に NearSurface を表すフラグ、B に $C(v)$, A に Empty を表すフラグを割り当てている。入力：ボクセル位置 v , カメラパラメータ p_c^i , 光源位置の仮想カメラのパラメータ p_s^i , カメラ距離画像のテクスチャ R_c^i , 光源距離画像のテクスチャ R_s^i 。

Step1 : v を p_c^i 及び p_s^i で射影した点 $x_c^i(v), x_s^i(v)$ を計算。

Step2 : カメラおよび光源からの距離 $r_c^i(v), r_s^i(v)$ を $R_c^i(x_c^i(v)), R_s^i(x_s^i(v))$ としてサンプリング。

Step2 : $r_c^i(v)$ が欠損値でなければカメラ視点からの符号付距離 $d_c^i(v)$ (視線距離) を計算。同様に $r_s^i(v)$ が欠損値でなければ光源視点からの符号付距離 $d_s^i(v)$ (視線距離) を計算。

Step3 : $d_c^i(v)$ あるいは $d_s^i(v)$ が計算され、絶対値が閾値以下の場合 (NearSurface の場合) $d_c^i(v), d_s^i(v)$ のどちらが有効であるかによって画素値 $(d_c^i(v), 1, 0, 0)$ あるいは $(d_s^i(v), 1, 0, 0)$ を返す。 $d_c^i(v)$ あるいは $d_s^i(v)$ が計算され、絶対値が閾値以上の場合 (Empty の場合) 画素値 $(0, 0, 0, 1)$ を返す。 **Step4** : $r_c^i(v), r_s^i(v)$ の欠損によって場合分けを行い、画素値 $(0, 0, c_i(v), 0)$ を返す。

処理後のフレームバッファの内容は、 $G > 0$ の場合 NearSurface であるため $R(D(v))$ の値を符号付き距離とする。 $G = 0, A > 0$ の場合は Empty に該当するため物体外部の値を割り当てる。 $G = 0, A = 0$ の場合 $B(C(v))$ の値によって物体外部と内部を判断し、それぞれ値を割り当てる。符号付距離の重み付き和をとる場合には、距離画像の各画素についてあらかじめ重みを計算しておき、テクスチャとすることで、簡単に実装できる。

ボリューム全体のうち、等値面形状にかかわる部分は、一部のみである。最初に荒い解像度で上記のアルゴリズムの計算を行い、得られたボクセルのうち、NearSurface と判定された領域、あるいは物体の外部と内部が接する領域のみについて、より細かい解像度でレンダリングを行うと、処理を高速化することができる。後に述べる実験では、提案手法の実装に、このような階層的手法による高速化を用いた。

5 実験

提案手法の有効性を示すため、二種類のデータを用いて実験を行った。一つは、メッシュデータとして存在する形状モデルから得られた合成データである。形状モデルから、視点位置を変えながら 7 個の距離画像を作成した。このと

き、アクティブステレオ法による計測を想定して、光源からの occlusion が発生している箇所は、欠損データとした。もう一つは、実際の物体をカメラとプロジェクタを用いたアクティブステレオ法で計測し、得られた実測データ(猪の形をした置物)である。計測は、視点位置を変えながら8回行った。

それぞれのデータを、SC法、volumetric diffusion法[2](以下VD法)、佐川らによって提案された複数の距離画像から符号付距離場に関する「コンセンサス」を得る方法[3](以下コンセンサスの頭文字からC法)、及び、提案手法(アクティブステレオによる光源位置の情報を使用した場合)によって統合し、メッシュモデルを作成した。SC法とVD法では、符号付距離場は視線距離によって計算され、C法では、符号付距離場はユークリッド距離によって計算されている。VD法ではボリュームの拡散を30回繰り返した。また、ボリュームのサイズは、 $512 \times 512 \times 512$ とした。高速化のため、提案手法ではまず $62 \times 62 \times 62$ の解像度で処理を行い、表面の存在する領域についてのみレンダリングを行った。また、提案手法では、補間された部分におけるエイリアシングを防ぐため、ボリュームに対して $3 \times 3 \times 3$ の平滑化フィルタを適用した場合と、適用しない場合の処理を行った(SC法およびVM法では行わなかった)。SC法、VD法、提案手法はIntel社Zeonプロセッサ(2.8GHz)のCPUと、nVidia社GeForce8800GTXのGPUを搭載したPCで実行し、C法については、Opteron 275(2.2GHz) \times 2個(合計4CPU)構成のPCで実行した。

補間なしの場合の統合結果を図4、それぞれの方法による補間処理の結果(提案手法は平滑化を適用した場合)を図5,6に示す。また、補間なしの場合、SC法、VD法、本手法について、合成データを処理したときの符号付距離場と、判別関数 $C(v)$ を、ある平面で切断した画像を図7に示す。それぞれの手法での実行時間と、提案手法(平滑化あり)における実行時間の内訳をそれぞれ表1、表2に示す。

表1: 実行時間。ただし、C法における、合成データのボリュームサイズは $128x128x128$ 。

| 手法 | 処理時間(秒) | |
|------|----------------------------|-----------------------------|
| | 合成 | 実測 |
| SC法 | 55 | 38 |
| VD法 | 168 | 120 |
| C法 | 15分 (統合:15分, 補間:18秒) | 7時間 (統合:7時間, 補間:280秒) |
| 提案手法 | 25(平滑化なし) 36*(平滑化あり) | 21(平滑化なし) 28*(平滑化あり) |

*内訳は表2を参照

図5(a)及び図6(a)から、合成データ、実測データどちらの場合でも、SC法では対象物体の周囲に余分なメッシュが生じている。これは物体の周囲の符号付距離場に、Unseen(未観測)の「削り残し」(図7(b)における対象物体外部のグレーの領域)が発生し、UnseenとEmptyの間に余分なメッシュが生じたためである。対象物体のメッシュとこれらのメッシュが分離していれば、こうしたメッシュは取り除かれるが、今回用いたデータでは余分なメッシュと対象物体が連結し、図に示すような結果となった。

また、図5(b)を見ると、VD法ではウサギの像の底面部分の穴は補間されているが、耳の下部や胸の部分において望ましくない方向に穴の周辺のメッシュが広がっている。

表2: 提案手法(平滑化あり)における実行時間の内訳

| 処理内容 | 実行時間(秒) | |
|----------------|---------|----|
| | 合成 | 実測 |
| 初期化と荒い解像度での処理 | 4 | 4 |
| GPUでのレンダリング | 11 | 7 |
| ボリューム処理(平滑化含む) | 12 | 10 |
| メッシュ生成(MC法) | 9 | 7 |
| 合計 | 36 | 28 |



(a) 合成データ (b) 実測データ

図4: 統合結果(補間なし)

これは、図7(e)から見て取れるように、これらの領域においては、形状の法線方向と距離画像の計測方向が大きく違うため、符号付距離場のNearSurfaceとUnseenの境界面(図7(e)の赤い線)が、法線方向とは異なる方向に生じ、これにより、フィルタによる等値面の拡大が、形状の表面の広がりとは違う方向に起こるため(図7(f)では右下方向に等値面が拡大している)であると考えられる。同様な現象は、実データにおいても起こっている(図6(b))。

一方、C法では全ての穴が埋まり、穴の部分に生成された曲面が平坦なものになったことから、補間の品質は実験された手法の中では最も良いものになった。ただし、C法では、符号付距離場の計算にユークリッド距離を用いるため、処理には大きな時間がかかった。

提案手法では、穴の部分に生成された曲面は、不自然になっている部分があるが、全ての穴が埋まり、補間の品質としてはC法に次いで良いものになっている。また、メッシュのトポロジーは、球体と等しくなった。このことから、補間された曲面に対してメッシュの曲率の最小化などの処理を施すなどの後処理により、結果の改善が可能であると思われる。

また、図7(g)より、判別関数 $C(v)$ が正となる領域は、対象物体の形状の粗い近似になっていることがわかる。

計算時間については、表1より、提案手法では、平滑化を行わない場合(SC法、VD法では平滑化をしていない)にはSC法の0.45~0.55倍、VD法の0.15~0.17倍、平滑化を行った場合でもSC法の0.65~0.74倍、VD法の0.21~0.23倍の計算時間で終了したことが分かる。このことから、提案手法をGPUで実装することにより、高い効率で、かつ品質の高い補間結果が得られたことが分かる。

6 おわりに

本論文では、space carving法を改良し、距離画像が少ない場合においても安定した結果を得ることができる補間のアルゴリズムを提案した。これは、符号付距離場におい

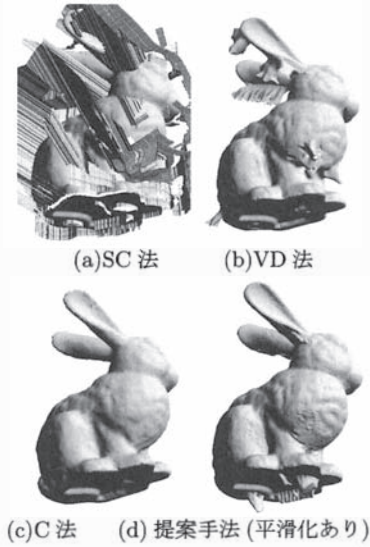


図 5: 統合結果 (合成データ)

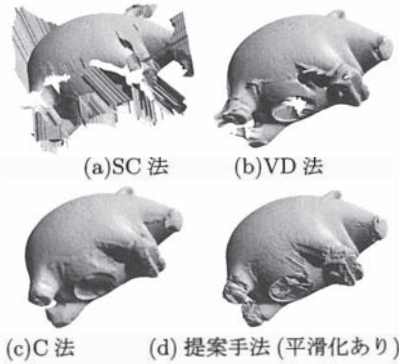


図 6: 統合結果 (実測データ)

て、未観測のボクセルについても、物体の内側か外側かを判定するための、ベイズ推定に基づいた判定関数を定義することで実現される。また、アクティブステレオ法によって得られる距離画像を用いることで、この判定関数の精度を向上させる手法についても提案した。さらに、提案手法を GPU によって実装する方法について述べ、大幅な計算時間の短縮を実現した。最後に、提案手法と既存手法について実験を行い、本手法の有効性を確認した。

参考文献

- [1] B. Curless and M. Levoy: "A volumetric method for building complex models from range images", *Computer Graphics*, **30**, Annual Conference Series, pp. 303-312 (1996).
- [2] J. Davis, S. R. Marschner, M. Garr and M. Levoy: "Filling holes in complex surfaces using volumetric diffusion.", *3DPVT*, pp. 428-438 (2002).

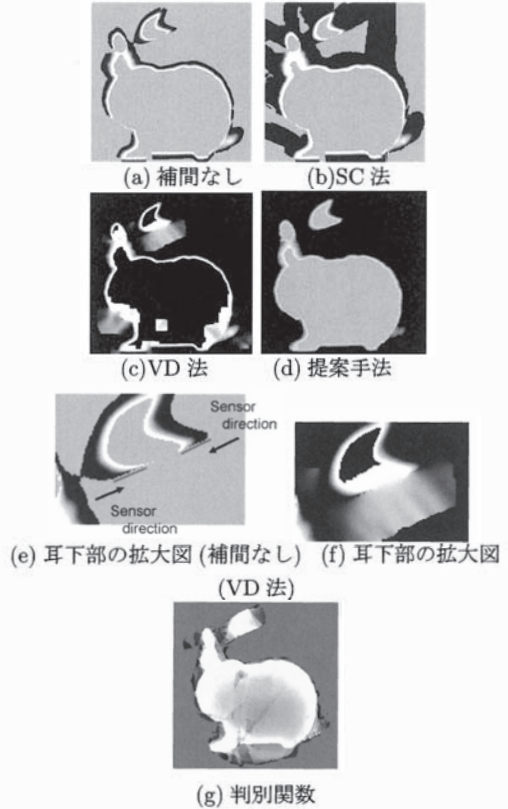


図 7: (a)-(f):符号付距離場の断面図: 符号付距離場の断面図:補間なし,SC 法, VD 法では均一なグレーが Unseen, 黒が Empty, 変化するグレーが NearSurface 領域の符号付距離を表す。提案手法では、均一なグレーが物体内部を、黒が物体外部を表す。(g):判別関数の断面図: 変化するグレーの輝度値は正の値を、暗い均一なグレーは負の値であることを表す。

- [3] 佐川, 池内: "符号付距離場の整合化による形状モデル補間手法", 電子情報通信学会論文誌 D, **J88-D2**, 3, pp. 541-551 (2005).
- [4] C. NITSCHKE, A. NAKAZAWA and H. TAKE-MURA: "Real-time space carving using graphics hardware", 画像の認識・理解シンポジウム (MIRU) 2006, pp. 928-933 (2006).
- [5] A. Sud, N. Govindaraju, R. Gayle and D. Manocha: "Interactive 3d distance field computation using linear factorization", *SI3D '06: Proceedings of the 2006 symposium on Interactive 3D graphics and games*, New York, NY, USA, ACM Press, pp. 117-124 (2006).