

# 距離画像センサを用いた 運動する物体のリアルタイム3次元形状復元

森 正樹<sup>1,a)</sup> 床井 浩平<sup>2,b)</sup>

**概要:** 本研究では運動する物体の周囲3方向に距離画像センサを設置し、それらから得たデプスマップを融合して物体の全周形状をリアルタイムに復元する手法を提案する。3次元の全周形状の復元は一般に負荷の高い処理となるため、これをリアルタイムに行うことにはいくつかの困難がある。そこで本研究では、これにGPUによる形状処理手法を導入する。さらに、この技術を用いて、復元した空間内でのインタラクションや、運動する人体の全周へのリアルタイムプロジェクションマッピングの実現を目指す。

## Real-time 3D Reconstruction of a Moving Object Using a Range Image Sensor

MORI MASAKI<sup>1,a)</sup> TOKOI KOHE<sup>2,b)</sup>

**Abstract:** This paper describes a method to reconstruct in real-time the entire shape of an object by fusing the depth map obtained from three range image sensors set up around the moving object. Since 3D reconstruction of entire shape is heavy processing in general, there are some difficulties in processing it in real-time. In this paper, we introduce the shape processing method based on the GPU. Moreover, by using this technique, we aim to implement that interaction in the reconstructed space and Real-Time Projection Mapping to the entire moving human body.

### 1. はじめに

3次元コンピュータグラフィックスの応用の広がりとともに、近年、3次元形状データの需用が急速に高まっている。特に実測にもとづいた3次元形状データは、建設や製造業における周囲の把握、考古学や土木工学などの物体形状の定量的評価、VFX (Visual Effects, 視覚効果) のための素材、3Dプリンタなど、様々な分野で求められている。

このことから、画像から3次元形状を復元する技術に関する研究は近年活発に行われている。

また前述の応用では、単一視点から見た形状のみではなく、全周形状の取得が求められている。これに加えて、運動

する物体の形状が復元できれば、さらなる応用の広がりが期待できる。

### 2. 関連研究

富山ら [1] は視体積交差法とステレオマッチング法を併用することで、多視点画像から動物体の高精細な3次元モデルを生成することに成功している。視体積交差法では複数のカメラから得られる物体のシルエットから形状を求めるため、多数のカメラを必要とする。この手法ではカメラを19台用いており、モデルの生成はリアルタイムに行われていない。

同じく、竹中ら [2] も視体積交差法を用いた形状復元を行っている。こちらはGPUでの並列化処理を行わず、複数の平板ポリゴンを組み合わせたボクセル空間に投影マッピングを行うという方法で高速化している。

また、Furukawaら [3] は近赤外プロジェクタとカメラを円周上に配置し、動物体の全周形状を復元している。こち

<sup>1</sup> 和歌山大学大学院システム工学研究科  
Graduate School of Systems Engineering, Wakayama University

<sup>2</sup> 和歌山大学システム工学部  
Faculty of Systems Engineering, Wakayama University

a) s145047@sys.wakayama-u.ac.jp

b) tokoi@sys.wakayama-u.ac.jp

らはプロジェクタによって、パターン光を物体に投影して形状をアクティブに計測する手法である。このような手法では、プロジェクタはカメラの数と同じだけ必要となり、こちらの研究では6セット用いられている。

### 3. 提案手法

物体の全周形状を復元するために、多くの関連研究では多数のカメラやプロジェクタが使われているが、本研究では安価な距離画像センサ3台のみを用いて、それらから得られるデプスマップを元に、できる限り低コストな方法で3次元形状の復元を行う。

デプスマップとはそれぞれの画素にカメラからの距離(デプス)の値が格納されている画像であり、わずかな計算で物体の形状が計測できるため、3次元形状復元の研究で盛んに利用されている。池畑ら [5], Liら [6] はカラー画像からデプスマップを推定し、融合を行っている。しかし、近年では安価な距離画像センサが実用化されているので、本研究ではそれを利用してデプスマップを得るため、デプスマップの推定の必要はない。

撮影は図1のように物体の周囲にセンサを配置して行い、それぞれの方向からデプスマップを取得し、それらを融合することによって物体の全周形状を同時に復元する。

形状処理はGPUを用いることで高速化を図り、リアルタイムな動物体の形状復元を実現する。

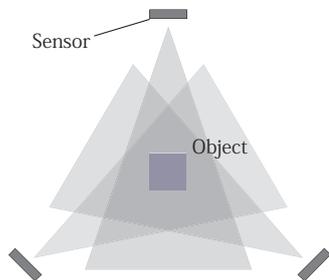


図1 3台の距離画像センサを用いた撮影方法  
Fig. 1 3D scanning method using 3 sensors

### 4. デプスマップを用いた形状復元

本研究で用いた距離画像センサは SoftKinetic 社の DepthSense325 である。画像の取得は DepthSenseSDK を用いて行った。

可視化には OpenGL を用い、DepthSense から得られるデプスマップを元に形状を構成し、同時にカラー画像をマッピングして描画を行った。

#### 4.1 画像の取得

DepthSenseSDK によって DepthSense から取得しているデータは解像度 QVGA (320 × 240) のデプスマップ (値は float 型), VGA (640 × 480) のカラーマップである。

DepthSenseSDK では、後述する実座標を表す頂点情報を得ることもできるが、GPU へのデータ転送量を減らし、同時に CPU の負担を下げるために、提案手法ではデプスマップのみを GPU に転送し、頂点の計算を GPU 上でやっている。また、デプスマップはセンサ3台分の形状の融合処理にも用いる。

#### 4.2 実座標の計算

各センサから得られるデプスマップはデプス値を格納した2次元のデータであり、その  $x, y$  座標は透視投影変換が施された後の位置である。よって、透視投影変換の逆変換を行うことにより、実空間における座標を求める必要がある。

透視投影変換されたデプスマップの画像面と、実空間に存在する物体との垂直方向における位置関係を、図2に表す。ここで、 $fovy$  は DepthSense の垂直方向の画角、 $depth$  はデプスマップに格納されているデプス値である。 $y'$  は実空間の物体上の点の位置であり、透視投影変換によって  $y$  に投影される。このとき  $a$  は画像の前方面の位置  $h$  と  $fovy$  を用いて次式で表される。

$$a = h \tan\left(\frac{fovy}{2}\right) \quad (1)$$

また  $a$  を1とすると  $y$  は式(1)を用いて次式で表すことができる。

$$y = y' h \tan\left(\frac{fovy}{2}\right) \quad (2)$$

最後に  $depth$  と  $h$  の比から、実座標  $y'$  を次式で求める。

$$y' = \frac{depth}{h} \left( y \tan\left(\frac{fovy}{2}\right) \right) \quad (3)$$

水平方向についても同様に求めてみる。

この実座標への変換処理は GPU のバークスシェーダ上に実装し、GPU 上で頂点単位に実行する。画角は DepthSense325 の公表値である、水平方向 74.0°, 垂直方向 57.9° を用いており、この値を式(3)に適用して計算を行ったとき、概ね正確な形状が求められた。

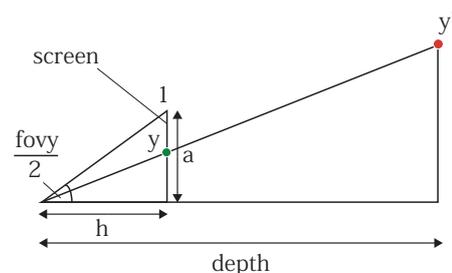


図2 実座標への変換(垂直方向)  
Fig. 2 Transformation to real point (Vertical direction)

### 4.3 全周形状の構成

#### 4.3.1 形状の位置合わせ

3台のセンサで構成した形状を融合させるためには、それぞれの位置関係がわからなくてはならない。システムの汎用性を考えると、各センサで撮影したカラー画像から位置推定を行うべきであるが、まずは各センサを正三角形の頂点に、重心を向くように固定して配置する方法を取った。したがって、プログラム上では各センサがそのような位置にあることを前提とした位置合わせを行った。

3台のセンサで構成した形状を A, B, C とし、位置合わせの手順を図 3 に示す。まず三角形の重心から各頂点への距離を  $d$  とし、A, B, C それぞれ  $z$  方向に  $-d$  だけ平行移動させ、回転の中心を各センサの視点からオブジェクトの中心に移動する。

その後 A に  $120^\circ$  回転行列, C に  $-120^\circ$  回転行列を乗じて、それぞれを同時に描画すると、CG 空間内に 3 台のセンサで取り囲んだ正三角形の空間が再現される。

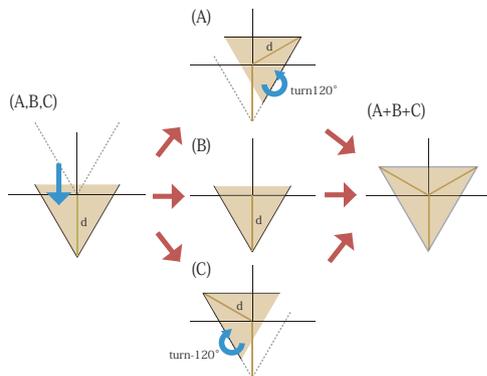


図 3 形状の平行移動と回転

Fig. 3 Translation and rotation of the shapes

#### 4.3.2 デプスマップの比較

先述した処理によって 3 台のセンサで構成された形状の合成ができる。しかし、そのまま描画すると重なる部分や、物体の影にあたる部分に不必要なポリゴンが生成される。よって描画の際にフラグメントシェーダにおいて 3 つのデプスマップ全てを参照、比較し、描画する点の可視判定を行う。この処理をデプスマップの融合とし、図 4 がそのイメージである。

デプスマップ比較の例を図 5 に示す。まず前処理として、シェーダ側にテクスチャとして GL\_FLOAT 型のデプス値を格納したデプスマップ 3 台分と、 $120^\circ, -120^\circ$  回転行列を送っておく。A, B, C の各センサは上段の図のような向きから対象物体を撮影している。太線  $a$  はセンサ A によって描かれるポリゴンである。ここで、B から得られるデプスマップ  $b$  と、 $a$  の  $z$  値を比較するために、 $a$  に B のカメラから得られる位置に移す変換を施し、 $a'$  とする。その上で比較を行うと、 $a'$  の破線部分はデプスマップ  $b$  の

外側にある (デプス値が小さい) ため、描かないと判定する。同様に、 $a$  を C のカメラから得られる位置に移すと  $a''$  となり、デプスマップ  $c$  と比較し可視判定を行う。

以上の処理を A, B, C について行うことで、図 4 のような 3 つのデプスマップの論理積にあたる形状が得られる。

図 6 は以上の処理を行ったプログラムの実行画面である。

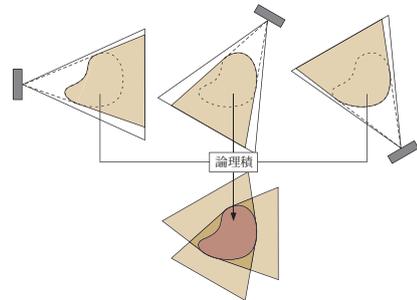


図 4 デプスマップの融合

Fig. 4 Depth map fusion

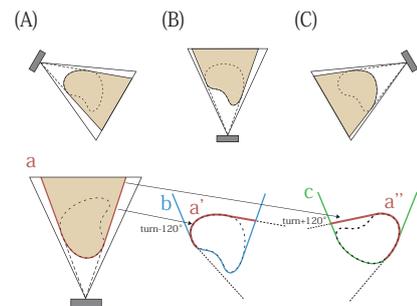


図 5 デプスマップの比較

Fig. 5 Comparison of depth maps

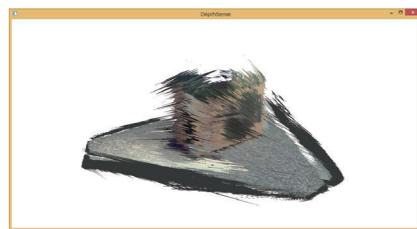


図 6 プログラム実行画面

Fig. 6 Screenshot of running program

### 4.4 ノイズの除去

図 6 を見ると、depth 方向にかなりのノイズが出ている。これはセンサから得られる各点の depth 値が微小に変動し、不安定であるために生じているものである。

そこで、時間方向で depth 値を平均化することを考えた。まず値を平均化するフレーム数を  $N$  とし、Vertex Buffer Object (VBO) を  $N$  個分用意する。プログラムを実行すると各フレームで得られるデプス値を VBO の先頭から順

に格納し、 $N$  番目まで格納されれば、また先頭に戻って格納するということを繰り返すことで、 $N$  個の VBO には常に直近の  $N$  フレーム分のデプス値が入るようにする。

バーテックスシェーダには  $N$  個の VBO 全てが送られてくる状態とし、全てのデプス値の平均をとった上で、先述の実座標の計算を行うようにする。

$N = 10$  として以上の処理を行うと、デプス値が安定し、形状が滑らかになったが、対象物体の動作に対して頂点の移動が遅れ、リアルタイム性が損なわれるという結果になった。

## 5. 提案手法の応用

### 5.1 現実空間への没入

本研究の提案手法によって復元された空間を、バーチャルリアリティ用の HMD (ヘッドマウントディスプレイ) などで見ることによって、現実空間への没入を実現することができる。

これによってその場にいながら、復元した物体を自由な視点で見ることが可能となる。複数名のユーザーの形状を復元し、それぞれ他のユーザーの姿を HMD で見れば、仮想空間内での複数人のインタラクションも可能であると考えられる。

また将来的に大規模な距離を計測できるセンサが実用化されれば、スポーツの試合や音楽ライブなどの空間を復元し、その空間に没入するという、新しい形の中継スタイルの実現にも繋がると考えられる。

### 5.2 運動する人体の全周へのリアルタイムプロジェクションマッピング

まずプロジェクションマッピングとは、プロジェクタにより建物などの形状に合わせて映像を投影し、映像を動かすことで、あたかもその物体自身が変形しているかのように見せる技術である。しかし、これまで行われているのは建物などの静止物体の、その一面に対してのみ映像を投影するものであった。

そこで本研究では、提案手法を応用して、運動する人体の全周に対して動的にプロジェクションマッピングを行うことを考える。

人体の全周へのプロジェクションマッピングについては DrawLight net が行ったもの [4] があるが、これはあらゆる角度から見たモデルの形状に合わせて画像を作成し、それらを繋ぎ合わせて投影していると考えられる。

それに対し本研究では、前述の提案手法により、運動する人体の 3 次元形状モデルが得られ、それをリアルタイムに加工し、プロジェクタの位置を計算して人体に投影することで、運動する人体の全周へのリアルタイムプロジェクションマッピングが実現できると考える。

## 6. まとめ

本研究の提案手法により三台の距離画像センサを用いて、リアルタイムに運動する物体の全周形状を復元することができた。しかし、ノイズの除去が不十分であり、平均化するフレーム数を増やすと、リアルタイム性が損なわれるという問題が生じた。よって今後はリアルタイム性を保ちつつ、ノイズを除去する方法を検討する必要がある。

また、現在は距離画像センサをあらかじめ定めた位置に固定しているため、センサが動いてしまうと正確な形状が得られないことになる。これでは汎用性があるとは言えないので、今後は指標となるものを置く、カラー画像から特徴点を抽出するなどして、各センサの位置を推定できるようにする。

## 参考文献

- [1] 富山仁博, 片山美和, 岩館祐一, 今泉浩幸: 視体積交差法とステレオマッチング法を用いた多視点画像からの 3 次元動オブジェクト生成手法, 映像情報メディア学会誌, Vol. 58, No. 6, pp. 797-806, 2004.
- [2] 竹中 史雄, 藤本 忠博, 原美 オサマ, 千葉 則茂: 投影マッピングを利用した多視点カメラ映像からのリアルタイムな仮想視点映像生成法, 芸術科学論文誌, Vol.10, No.4, pp. 263-275, 2011.
- [3] R Furukawa, R. Sagawa, H. Kawasaki, K. Sakashita, Y. Yagi, N. Asada: one-shot entire shape acquisition method using multiple projectors and cameras, Fourth Pacific-Rim Symposium on Image and Video Technology, IEEE Computer Society, pp. 107-114, 2010.
- [4] DrawLight.net: Rabarama 360° Full 3D Mapping Projection, 入手先 (<http://www.drawlight.net/en/project-details/items/rabarama-360-full-3d-mapping-projection.html>) (2011.05.10).
- [5] 池畑諭, 山崎俊彦, 相澤清晴: 多視点ステレオのための信頼度に基づく高精度デプスマップの推定と融合, 画像の認識・理解シンポジウム (MIRU 2011)
- [6] Jianguo Li, Eric Li, Yurong Chen, Lin Xu, Yimin Zhang: Bundled depth-map merging for multi-view stereo, IEEE Computer Vision and Pattern Recognition, pp. 2769 - 2776, 2010.