

部分空間の境界面と関節物体との交差判定による高速形状復元手法

濱 崎 省 吾[†] 吉 田 裕 之[†] 重 永 信 一[†]

複数視点のシルエット画像から3次元空間を復元する視体積交差法の本来の復元精度を保証し、実時間処理可能な高速なアルゴリズムについて提案する。従来から、計測空間を部分空間に分割し、物体と部分空間との高速な重なり判定により復元する方法があったが、処理の高速化のために、重なり判定の正確さを犠牲にしており、本来の視体積交差法の復元精度は保証されていない。本手法では、計測物体(人物)の大きさや形状に関する情報を使って、物体と部分空間との正確な重なり判定を行い、従来処理より高速でかつ視体積交差法本来の復元精度を保証できる方式とした。具体的には、計測物体を既知サイズの部品で構成された関節物体と想定し、関節物体がとりうる形状の最小サイズより定めた最初の部分空間から、漸次的に細かく分割してゆき、各部分空間の壁面(境界面)上を、関節物体の最小部品サイズから定まる間隔で、所望の空間分解能のボクセルと物体との交差を走査することで、物体と重ならない部分空間を正確に排除し、高速に復元する。実際に、ポリゴン表現の人物動作を、リアルタイムに、仮想世界に取り込み表示するシステム(PC2台とカメラ4台)を試作し、2,400[mm]立方の部屋で、25[mm]空間分解能で復元する場合、15.9[frame/sec]の性能を確認した。

High Speed Shape Reconstruction Method by Testing Overlaps between Object and Measuring Partial Spaces' Boundary Planes

SHOGO HAMASAKI,[†] HIROYUKI YOSHIDA[†] and SHINICHI SHIGENAGA[†]

This paper presents a high speed algorithm of view volume intersection method for 3D voxel-based reconstruction from multi-viewpoint silhouette images. In a known method, space of interest is divided into partial spaces, and 3D volume of measured object is reconstructed by efficiently testing overlaps between the object and the partial spaces with the loss of testing accuracy. In our method, testing overlaps is performed exactly and at high speed, in terms of size and shape information about measured object. Suppose that measured object consists of known sized jointed parts, space of interest is gradually divided into partial spaces of which size are initially determined by the minimum size of object shapes, and partial spaces without a overlap on the object are eliminated exactly and at high speed by scanning intersection of the object and given spatial resolution voxels on partial spaces' boundary planes at intervals determined by the minimum size of constituent parts. In the prototyped system with 2 PCs and 4 cameras, which takes body motion expressed by polygons in a virtual world in real time, we confirmed reconstruction performance of 15.9[frame/sec] in the case of 25[mm] spatial resolution at 2,400[mm] cubic room.

1. はじめに

人物を対象とした3次元動作計測技術は、CGアニメーション制作、スポーツ解析、健康医療、バイオメカニクス、仮想世界コミュニケーション等の幅広い分野でニーズが高まっている。

人の動きを計測する従来のモーションキャプチャシステムには、磁気式や機械式等の接触センサによる方式、身体部位に装着したマーカをカメラ撮影する方式等がある。これらの方式では、部位の位置情報である

骨格データを得ることができる。

一方、人の動作を仮想世界に取り込み仮想物体とインタラクションを実現するようなシステムにおいては、骨格データではなく体型等の形状データを実時間で正確に得ることが求められる。形状データを得る方式としてレンジファインダがあるが、現状では計測範囲が狭く、また秒オーダーの走査時間が必要であることから、人物動作の計測に用いることは困難である。

近年、装着物なしに汎用 CCD カメラの画像から物体の形状データを計測する研究がさかんである。

主要なものとして、計測空間の周囲に設置した複数台のカメラで撮影した画像に、ステレオ視法¹⁾を適用して得られた距離画像から、物体形状をポリゴンデー

[†] 松下電器産業株式会社マルチメディア開発センター
Multimedia Development Center, Matsushita Electric
Industrial Co., Ltd.

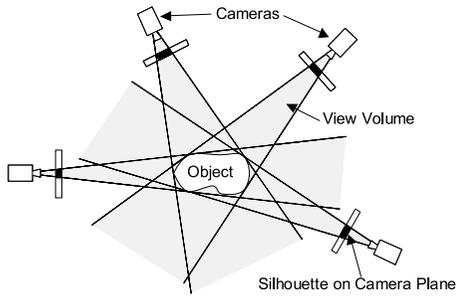


図1 視体積交差法の概念

Fig. 1 The concept of view volume intersection method.

タとして復元する研究²⁾や、また、視体積交差法によりシルエット画像から物体形状を3次元再構成して形状データを得る研究^{3)~5)}がある。

視体積交差法はステレオ視法と比較して、対応点問題が発生しない、アルゴリズムが単純で空間分割による処理の高速化に適している等の特長を持つ。

多視点のシルエット画像から物体形状を3次元再構成する視体積交差法の概念図を図1に示す。カメラ焦点を頂点として物体シルエットを空間に投影した錐体(視体積)を考え、複数カメラによるすべての視体積が空間で交差する共通体積を求める。この共通体積が物体の3次元形状を表現していると見立てる。

視体積交差法は次の2つの手順から構成される。

投影処理：計測空間の微小領域(ボクセル)を各カメラ平面へ投影する。

交差処理：すべてのカメラ平面で物体シルエットと重なる場合はそのボクセルを残し、それ以外の場合は削除する。

計測空間を分割したすべてのボクセルごとに、投影処理と交差処理を適用することにより、ボクセル集合体で近似された共通体積として計測対象の3次元形状を得ることができる。

より実物体の形状に近づけるためには、“カメラを増やすことで、より多くの視体積を重ねて共通体積をより小さくする”、“計測空間の分割数を増やすことで、より空間分解能を上げてより微小なボクセルで復元する”ことが必要となる。カメラ数を N_c 倍にすると視体積の重合せ回数は N_c 倍に、空間分解能を N_v 倍にすると処理対象となるボクセル数は N_v^3 倍にそれぞれ増大する。

視体積交差法を用いて物体の形状データを計測するシステムを構築するため、これらの処理量増大に対する取り組みが行われている。

パナシールカメラを用いて広い計測空間で形状データを計測する研究⁴⁾では、演算量削減による投影処理

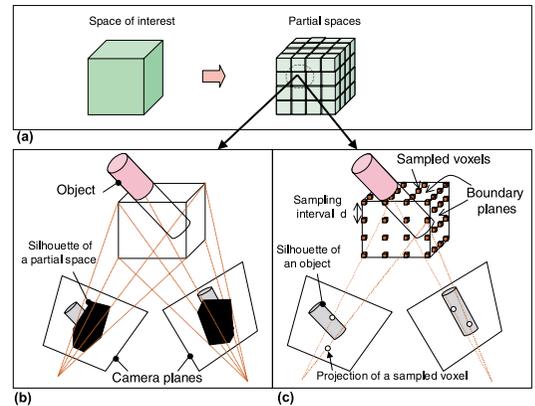


図2 部分空間と対象物体との重なり判定：(a) 部分空間分割，(b) 従来の改良手法，(c) 本手法

Fig. 2 Testing overlaps between partial space and object: (a) Partial space division, (b) Traditional improved method, (c) Our method.

の高速化と、並列処理アルゴリズムによる交差処理の高速化を行い、カメラ9台とPC10台で構成されたシステムにおいて、人物動作を3cmの復元精度で每秒12.37フレームの性能を実現している。しかし、交差処理の処理量そのものの削減には言及されておらず、復元精度の向上ともなう処理量増大への対応は十分とはいえない。

交差処理の処理量削減を目的として、計測空間をOctree構造で管理された部分空間に分割し、物体が含まれていない領域を効率良く排除する研究^{6),7)}がある。

これらの研究では、部分空間と物体との交差を、部分空間をカメラ平面に投影した六角形領域と物体シルエットとの重なりで判定している(図2(a), (b))。しかしながら処理量削減のために、六角形領域を矩形近似しその内部のみを調査するため、重なり判定を誤る可能性があり、正確な復元結果は得られない。

また、部分空間と物体との交差処理をカメラ平面上でサンプリング調査することで高速処理する研究⁸⁾がある。この研究では、部分空間をカメラ平面に投影した六角形領域の内部を一様にサンプリングした点について重なりを判定している。しかしながら、部分空間の内部を一様に調査したことにはならず、やはり誤判定の可能性があり復元結果の正確性は保証できない。

本論文では、計測対象(人物)のヒューリスティックな情報を利用して、視体積交差法の本来の復元精度を保証する高速な処理方法を提案する。以下、2章では、提案手法のアルゴリズム説明、および従来手法との処理量比較について述べ、3章では試作したインタラクティブシステムとその実時間動作に関する評価結果について報告する。

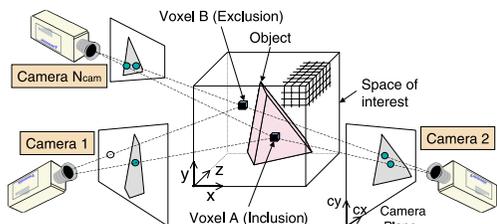


図3 視体積交差法によるボクセルの状態判定
Fig.3 Voxel occupancy determination.

2. ボクセル再構成処理の高速アルゴリズム

2.1 視体積交差法

視体積交差法³⁾について図3を用いて説明する。計測対象となる物体の周囲に N_{cam} 台のカメラを固定設置しシルエット画像を撮影する。カメラパラメータは既知とする。計測空間は X, Y, Z 方向の単位格子であるボクセルに分割し、個々のボクセルについて、物体の内部に存在するか外部に存在するかの状態を判定する。物体の内部に存在する場合を占有状態、外部に存在する場合を空状態と呼ぶ。ボクセルの大きさは空間分解能であり、3次元再構成の復元精度となる。

視体積交差法の処理フローを図4(a), (b)に示す。

最初に、ボクセルの状態を保持する状態テーブルを占有状態に初期化する。ボクセルがカメラ平面に投影される領域をカメラパラメータから決定する。その投影領域が物体のシルエット外部であれば、空状態と判定する。すべてのカメラ平面で投影領域が、シルエット内部であれば、占有状態と判定する。

計測空間の各辺を 2^N 分割した場合の総ボクセル数は次式となる。

$$T_N = 2^{3N}. \tag{1}$$

2.2 境界ボクセル法

人物の普段の動作を計測する状況を想定して、計測対象となる人物を既知サイズの部品で構成された関節物体とし、また、その動作を各関節の屈曲伸展としてモデル化する。

このモデル化により、対象物体は次の性質を持つ。

- (1) 関節物体は計測空間内に局所的に偏在する。よって、動作を制約しない広さを持つ計測空間では、物体が存在しない空の領域の割合は多いと考えられる。
- (2) 関節物体の最小状態は各関節を屈曲した姿勢と見なせる。具体的には人物が背を丸めた座位姿勢と想定でき、その大きさは各部品の大きさから算出できる。
- (3) 関節物体の最小部品は計測の要求仕様からユーザが指定する。具体的には指や手首が考えられ、その大きさは既知である。

```

Initialize all entries of status table to be "Occupied"
for each voxel v do
    call Func_intersection_test
end
    
```

(a)

```

Func_intersection_test
for each camera c do
    Project v into pixel p of camera plane
    if p locates outside of the silhouette then
        Set v to be "Empty"
        Break this loop
    fi
end
return
    
```

(b)

```

Initialize all entries of status table to be "Empty" ... (1)
Set P0 to be "Occupied"
for Level m=1 to N do
    for each partial space Pm-1 do ... (2)
        if Pm-1 is "Occupied" then
            Divide Pm-1 into Pm
        fi
    end
    for each partial space Pm do ... (3)
        if Lm >= d then
            Put sampled voxels on boundary plains of Pm into Q
        else
            Put all voxels on boundary plains of Pm into Q
        fi
        for each voxel v of Q do
            Set v to be "Occupied"
            call Func_intersection_test
        end
        if one v of Q is "Occupied" then
            Set Pm to be "Occupied"
        fi
    end
end
end
    
```

(c)

図4 視体積交差法の処理フロー：(a) 従来手法、(b) ボクセルの状態判定、(c) 本手法

Fig.4 A flow of view volume intersection method: (a) Basic method, (b) Overlap test, (c) Our method.

これらの性質を考慮して次の戦略をとる。

- 性質(1)を利用して、計測空間を部分空間に再帰分割することで効率良く空領域を排除しつつ、部分空間の大きさが目的の復元精度に達するまで分割を繰り返す。
- 性質(2)を利用して、部分空間の初期サイズを関節物体の最小状態に外接する大きさとする。このことにより、部分空間と関節物体との重なりは、物体と境界面(部分空間を構成する外壁面)との交差として判定できる。
- 性質(3)を利用して最小部品を検出できるサンプリング間隔を考える。最小部品より大きい部品と部分空間との交差は、その境界面をサンプリング

調査することで判定できる．なお，分割が進み部分空間がサンプリング間隔より小さくなった場合には，境界面上のボクセルを全数調査することで交差を判定できる．

これらの戦略から，本手法は次の特長を持つ．

- ユーザが復元したい最小部品を明示的に指定することで，最小部品より大きな部品について，視体積交差法の本来持つ復元精度を保證できる．
- 計測空間を分割した個々のボクセルごとに状態判定を行う従来手法と比較して，本手法は処理量を削減でき高速処理が可能となる．

以上のように対象物体と部分空間との空間的な重なりを，部分空間の境界面にあるボクセルのみで判定することから，我々は本手法を境界ボクセル法⁹⁾と呼ぶ．

2.3 処理フロー

本手法の処理フローを図4(c)を用いて説明する．使用する主要な変数を次のように定義する．

- N : 部分空間の分割回数．
- m : 分割経過を表す分割レベル．ただし， $0 \leq m \leq N$ ．
- P_m : m 回目の分割で発生する部分空間． P_0 は計測空間と一致する．
- L_m : P_m の一辺の長さ．
- d : 境界面をサンプリング調査する間隔．

(1) 状態テーブルの初期化

部分空間およびボクセルの状態を保存する状態テーブルを空状態に初期化する．

(2) 部分空間の分割

最初 ($m = 1$) は，計測空間 P_0 を初期サイズ L_1 の部分空間 P_1 に分割する．それ以降 m 回目の分割では， $m - 1$ 回目の分割で占有状態と判定した P_{m-1} のみをより小さい P_m に分割する．この処理を合計 N 回繰り返す．

(3) 部分空間の状態判定

部分空間の表面にあるボクセルの状態を対象物体と部分空間との重なりを判定する．処理対象 Q は，部分空間の大きさ L_m とサンプリング間隔 d との大小関係で次のとおり場合分けされる．

- $L_m \geq d$ のとき，図2(c)に示すように P_m の6つの境界面を間隔 d でサンプリングしたボクセルが処理対象 Q となる．
- $L_m < d$ のとき， P_m の6つの境界面上のすべてのボクセルが処理対象 Q となる．

処理対象 Q に含まれるすべてのボクセルを状態判定し，そのうち少なくとも1つが占有状態であれば，その場所で境界面と対象物体が交差しており P_m は占

有状態であると判定する．また，すべてが空状態ならば境界面と対象物体は交差しておらず P_m は空状態であると判定する．

以上を部分空間の分割で発生したすべての P_m について繰り返す．

2.4 処理量の試算

本手法は従来手法に比べて状態判定の処理対象となるボクセル数を削減できることを特長としている．その効果を見積もるため両手法で処理対象となるボクセル数を試算する．試算条件を次に示す．

- 立方体の計測空間を8つの部分空間に N 回再帰分割する．ただし $N \geq 2$ ．
- N 回分割後の部分空間 P_N をボクセルと一致させる．よってボクセルの一辺の長さは $a = L_N$ ．
- サンプリング間隔は整数 k ($N > k \geq 0$) を使って $d = a2^k$ と表現する．
- 境界面は隣接する部分空間との間で共用する．

部分空間の m 回目の分割で発生したすべての P_m のうち，占有状態と判定した部分空間の合計体積が計測空間に占める割合を ρ_m ($0 \leq \rho_m \leq 1.0$) とすると， m 回目の分割後に占有状態である部分空間の個数 C_m は次式となる．

$$C_m = \rho_m 2^{3m} \tag{2}$$

次に部分空間の状態判定において， m 回目の分割で新たに処理対象となるボクセル数を $L_m \geq d$ と $L_m < d$ とに分けて見積もる． $L_m = a2^{N-m}$ と $d = a2^k$ であることから， $m \leq N - k$ (サンプリング調査) と $m > N - k$ (全数調査) に分けてそれぞれ試算する．

(1) $m \leq N - k$ のとき

部分空間 P_m の状態判定では，6つの境界面上にあるボクセルが処理対象となる．図5に示すように，6面のうち3面は親である P_{m-1} の境界面の一部(たと

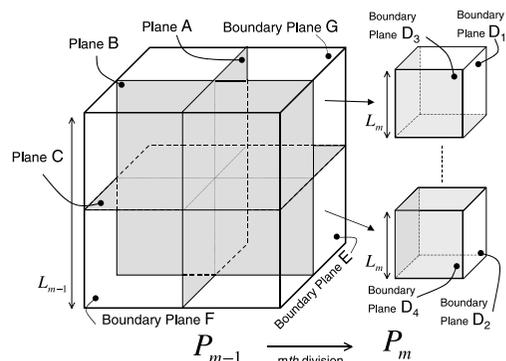


図5 m 回分割時の境界面発生
Fig. 5 Boundary plane generation at m -th division.

例えば、境界面 D_1, D_2 は境界面 E の一部) で、すでに判定済みである。また、残りの 3 面は P_{m-1} を 8 分割する分割面の一部 (たとえば、境界面 D_3, D_4 は、分割面 B の一部) である。よって m 回目の分割で新たに発生する境界面は、 P_{m-1} を 8 分割する分割面 A, B, C と一致する。各分割面上のボクセルを間隔 d でサンプリングする場合のボクセル数は $(L_{m-1}/d)^2$ であることから、 m 回目の分割で処理対象となるボクセル数 V_m は、占有状態にある P_{m-1} の個数 C_{m-1} を使って次式で表現できる。

$$V_m = 3(L_{m-1}/d)^2 C_{m-1}, \quad (m \leq N - k). \quad (3)$$

(2) $m > N - k$ のとき

部分空間 P_m の状態判定では、 $m \leq N - k$ のときと同様に、6 つの境界面のうち 3 面は親である P_{m-1} の分割面 A, B, C の一部であり、分割面 A, B, C 上にあるすべてのボクセルが処理対象となる。分割面-分割面、分割面-境界面の重なった領域にあるボクセルを重複カウントしないように、 m 回目の分割で新たに処理対象となるボクセル数 W_m を試算する。隣接する P_{m-1} 間で境界面を共有していることから、図 5 に示すように 1 つの P_{m-1} について 3 つの境界面 (ここでは E, F, G とする) と 3 つの分割面 A, B, C との交差について考慮する。分割面の一辺にあるボクセル数を $g = L_{m-1}/a$ とすると、各分割面上にある処理対象となるボクセル数は次のとおりである。

- 分割面 A は判定済みの境界面 F, G と交差するので、処理対象のボクセル数は $(g-1)^2$ 個となる。
- 分割面 B は判定済みの境界面 E, G, 分割面 A と交差するので、同じく $(g-1)(g-2)$ 個となる。
- 分割面 C は判定済みの境界面 E, F, 分割面 A, B と交差するので、同じく $(g-2)^2$ 個となる。

よって、 m 回目の分割で処理対象となるボクセル数 W_m は、占有状態にある P_{m-1} の個数 C_{m-1} を使って次式で表現できる。

$$W_m = [(g-1)^2 + (g-1)(g-2) + (g-2)^2] C_{m-1}, \quad (m > N - k). \quad (4)$$

ただし、 $m = N - k + 1$ のときは、 P_m の 6 つの境界面のうち P_{m-1} の境界面の一部である 3 面についてはサンプリング調査のみの状態であるので、 P_{m-1} の 6 つの境界面上のボクセルを全数調査する必要がある。よって、処理対象となるボクセル数は式 (4) に、占有状態である P_{m-1} の 6 つの境界面上のすべてのボクセル数 U を加えたものとなる。隣接する P_{m-1} 間で境界面を共有していることから、1 つの P_{m-1} について 3 つの境界面 (ここでは E, F, G とする) 上にあるボクセル数を新たに処理対象とすればよい。3 つの

境界面が互いに交差していることを考慮し、境界面の一辺にあるボクセル数を $h = L_{N-k}/a$ とすると、各境界面にある処理対象となるボクセル数は次のとおりである。

- 境界面 E で処理対象のボクセル数は h^2 個となる。
- 境界面 F は判定済みの境界面 E と交差するので、同じく $h(h-1)$ 個となる。
- 境界面 G は判定済みの境界面 E, F と交差するので、同じく $(h-1)^2$ 個となる。

よって、 $N - k + 1$ 回目の分割において、追加で処理対象となるボクセル数 U は、占有状態にある P_{m-1} の個数 C_{N-k} を使って次式で表現できる。

$$U = [h^2 + h(h-1) + (h-1)^2] C_{N-k}. \quad (5)$$

以上より、本手法による 3 次元再構成で状態判定の処理対象となるボクセル数 S_N は、各分割レベル m で新たに処理対象となるボクセル数の総和であり次式で表現できる。

$$S_N = \begin{cases} \sum_{m=1}^{N-k} V_m + \sum_{m=N-k+1}^N W_m + U, & (N > k \geq 1) \\ \sum_{m=1}^N V_m, & (k = 0) \end{cases} \quad (6)$$

ただし $N \geq 2$.

ここで、各分割レベルにおいて計測空間全体に対する占有状態の部分空間の割合を一定値 ρ と仮定すると、 $C_m = \rho 2^{3m}$ となり、式 (6) は次式で表現できる。

$$S_N = \begin{cases} 3(2^{N-k} - 1)\rho 2^{2(N-k)} \\ + (1 - 3 \cdot 2^{-k} + 3 \cdot 2^{-2k} - 2^{-3k})\rho 2^{3N} \\ + (3 \cdot 2^{-k} - 3 \cdot 2^{-2k} + 2^{-3k})\rho 2^{3N}, & (N > k \geq 1) \\ 3(2^N - 1)\rho 2^{2N}, & (k = 0) \end{cases} \quad (7)$$

ただし $N \geq 2$.

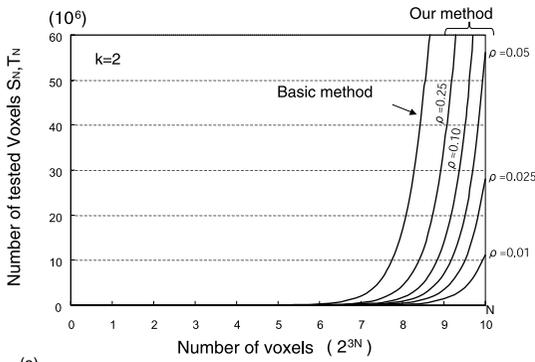
一方、従来手法では式 (1) に示した計測空間を構成するすべてのボクセルが処理対象となる。

2.5 比較

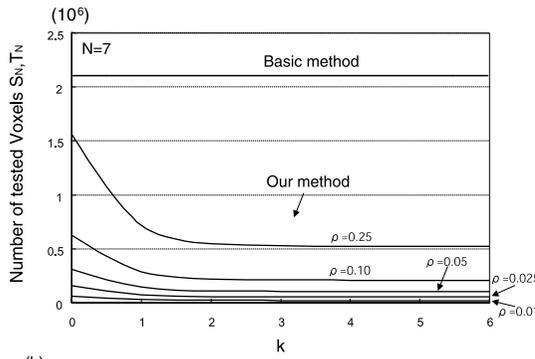
本手法と従来手法との処理量を比較する。

(1) 総ボクセル数と処理量

計測空間を分割した総ボクセル数は 2^{3N} 個である。総ボクセル数を変化させた場合の式 (1), (7) の値を図 6 (a) に示す。本手法は従来手法に比べて、総ボクセル数の増加 (N の増加) ともなう処理量の増加が少ないことが分かる。その効果は総ボクセル数が増加するほど顕著となっている。



(a)



(b)

図 6 従来方式と本手法との処理量比較：

(a) S_N, T_N と N , (b) S_N, T_N と k

Fig. 6 A comparison of number of tested voxels between basic method and our method:

(a) S_N, T_N against N , (b) S_N, T_N against k .

(2) 対象物体の大きさと処理量

計測空間の大きさと正規化した物体の大きさである ρ を変化させた場合の式 (7) の値を図 6 (a) に示す. 本手法は従来手法と比べて, 計測空間に対して対象物体が小さい (ρ が小さい) ほど, 処理量が少ないことが分かる.

(3) 最小部品大きさと処理量

最小部品を検出できる間隔は $d = a2^k$ である. k を変化させた場合の式 (7) の値を図 6 (b) に示す. 最小部品が大き (k が大きく) になると処理量は減少するが, $k \approx 2$ 以上でほぼ横這いとなっている. このことは, ρ を変化させた場合のグラフから分かるように, ρ に依存せず同じ傾向を示している.

(4) カメラ数と処理量

復元精度を向上させるためには, 図 1 に示したようにより多くのカメラを使って視体積を重ねる必要がある. ボクセルの状態を判定する処理 (図 4 (b)) において, for ループの繰返し回数がカメラ数であることから, カメラ数と処理量との間には線形比例の関係が

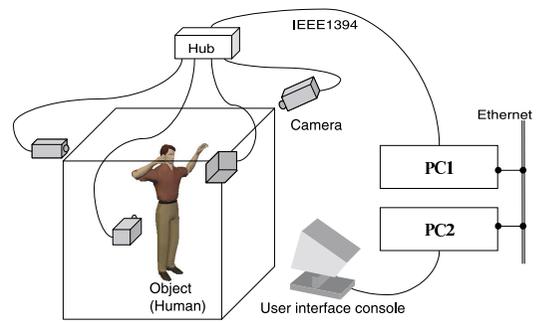


図 7 システム構成

Fig. 7 System architecture.

あることが分かる.

3. 試作システムと評価実験

境界ボクセル法のアルゴリズムを使って, 仮想世界とインタラクションを行うシステムを試作し¹⁰⁾, 実際の人物を対象とした実験でシステムの性能を検証した.

3.1 システム概要

本システムは, 医療現場で行われるリハビリテーションの 1 つで患者の関節運動能力の改善訓練を支援することを目的とした. 医療現場では療法士が患者の周囲に提示した風船等の目標物を, 患者が触れる動作を繰り返して訓練している. 我々はこの訓練を目標となる空間位置にユーザが身体の一部を重ねる繰返し動作として定式化し, 下記のオブジェクトで構成した仮想世界を設計した.

- アバタ: ユーザを形状復元したモデルで, ポリゴンデータとしてフレームごとにモデリングする.
- 目標オブジェクト: 小動物(蝶)のオブジェクトで, 羽ばたく動きを音と映像で表現する.
- 背景オブジェクト: 公園にある道, 木々, 池等で, 臨場感を演出する.

仮想世界で表現された公園にユーザはアバタとして入る. アバタの周囲には, ランダムに蝶が飛来し, パタパタという効果音とともに一定時間とどまる. その間にアバタの一部を蝶に接触させて捕獲し得点を得る. 以上のインタラクションをあらかじめ設定した時間繰返す. 得点が患者の運動能力の目安となる.

3.2 システム構成

本システムの構成を図 7 に示す. 計測空間の人物を撮影する 4 台のカメラと, 画像処理および仮想世界を制御する 2 台の PC と, 仮想世界の映像をフィードバックするディスプレイで構成している. カメラは IEEE1394 インタフェースを持ち, 撮影した画像を非圧縮のまま PC に転送する. カメラは計測空間の全体が

```

NEXTFRAME:
Capture run-time images ... (1)
Determine frame id FID
for each camera c do ... (2)
    Exec SilhouetteExtraction
    Put silhouette and FID into imgBuff[c]
end
Send imgBuff to PC2
goto NEXTFRAME:

```

PC1

```

Receive imgBuff from PC1 at any time
Initialize frame counter fc=0
NEXTFRAME:
for each camera c do #Frame Sync. ... (3)
    SKIP:
    Get silhouette and FID form imgBuff[c]
    if FID < fc then goto SKIP:
    if FID > fc then
        Unget silhouette to imgBuff[c]
        fc = fc+1
        goto NEXTFRAME:
    fi
end
Exec VoxelReconstruction
Exec PolygonGeneration_VRworldManagment ... (4)
fc = fc+1
goto NEXTFRAME:

```

PC2

図 8 システム処理フロー

Fig. 8 System flow on PC1 and PC2.

視野に収まるように天井および壁面に配置し、カメラキャリブレーション¹¹⁾を行う。PCはAT互換機でそれぞれ2つのCPUを搭載している(PC1: PentiumIII 850 MHz dual, PC2: 同1 GHz dual)。OSにはMS-Windows NTを、CG描画ライブラリにはOpenGLを使用している。ネットワークは、PC間を接続する100 MbpsのEthernetと、カメラ-PC間を接続する400 MbpsのIEEE1394を使用している。

3.3 動作概要

図8に示す処理フローに沿って、システムの具体的な動作を説明する。

(1) 同期撮影

外部トリガによって同時刻に4台のカメラで撮影する(図9(a))。画像データはIEEE1394バスによりPC1へ転送する。ボクセル再構成処理時に必要となるデータ転送同期(同時刻のシルエットを揃える処理)を行うため、画像ごとにフレームIDを決定し、画像データとともにバッファに格納する。フレームIDは、IEEE1394バスに流れる画像パケットのヘッダ情報を参照し、同期用ビットSY(各画像データの最初のパケットのみセット、それ以降ではリセット)がセット状態のパケットをカウントすることで決定する。

(2) シルエット抽出

あらかじめ撮影した人物を含まない背景画像と実行

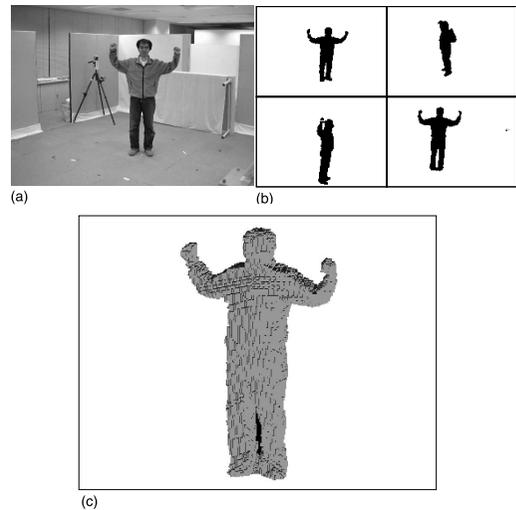


図9 ボクセル再構成の過程: (a) 撮影画像, (b) シルエット画像, (c) ボクセル表現

Fig. 9 Images from voxel reconstruction: (a) Run-time image, (b) Extracted silhouette images, (c) Voxel representation.

時の入力画像との差分から、ptracker¹²⁾で使用されている方式によりシルエットを抽出する(図9(b))。抽出結果はただちにEthernetへ送出する。PC2ではデータが到着し次第、FIFO形式の画像バッファに格納する。

(3) ボクセル再構成

視体積交差法の処理では同時刻に撮影したシルエットを入力する必要があるが、PC間をEthernetで接続しデータ転送しているため、遅延やデータ欠落が発生する可能性がある。そこで、以下のようにフレームIDを用いてデータ転送の同期を保証する。

- データが遅れて到着した場合、その内容を破棄し同一カメラの次のフレームを読み出す。
- データが欠落した場合、再構成処理を中断して次のフレームの処理を開始する。

● 上記以外の場合、次のカメラの画像を読み出す。同時刻に撮影した4枚のシルエットが揃うと、境界ボクセル法によりボクセルデータを再構成する(図9(c))。ボクセルデータはOctree構造で管理する。なお、ボクセルのカメラ平面への投影処理は、カメラキャリブレーション時に作成しておいたボクセルと各カメラ画素との対応テーブルを参照することで高速化している。

(4) ポリゴン生成

ボクセルデータからポリゴンデータを生成する代表的な手法としてMarching cubes法¹³⁾がある。Marching cubes法では、8つの隣接するボクセルの中心位置を頂点とする立方体(キューブ)を考え、物体表面



図 10 仮想世界映像の例

Fig. 10 Screen shots of the user interface console.

の位相が正しくなるように各キューブにポリゴンを貼り付けている。Marching cubes 法的高速化手法として Octree 構造を使ってポリゴンを生成するキューブのみを効率良く処理する手法¹⁴⁾がある。本システムでも、境界ボクセル法で使用した Octree 構造を使ってポリゴンデータを高速に生成する⁵⁾。ポリゴンで表現された人体動作はフレームごとに仮想世界のシーングラフに追加され、アバタの動きとして表現される。アバタと仮想物体との接触判定はポリゴンどうしの重なりとして判定する。また、オペレータの操作により、任意の視点からの仮想世界映像をユーザに提示することが可能である(図 10)。

以上のステップを実時間処理することで、インタラクティブなシステムを実現する。

3.4 実験

境界ボクセル法の効果を検証するため、本システムの処理速度と復元精度を評価する実験を行った。

3.4.1 実験条件

大人が立位で運動ができる広さを考慮して、計測空間は $L_0 = 2,400$ [mm] の立方体とした。背を丸めた姿勢から $L_1 = 800$ [mm] として、計測空間を最初に $3 \times 3 \times 3$ 分割し、以降再帰的に 8 分割する。手首を最小部品として $d = 50$ [mm] とした。復元精度であるボクセルの大きさは $a = 12.5$ [mm] (このとき $N = 7$, $k = 2$) および $a = 25$ [mm] (同 $N = 6$, $k = 1$) と

表 1 システム性能
Table 1 System performance.

	a [mm]	
	25	12.5
Capturing time [ms]	19	19
Silhouette extraction time [ms]	55	55
Voxel reconstruction time [ms]	24	49
Polygon generation time [ms]	63	337
System latency [ms]	155	460
System throughput [frame/sec]	15.9	3.0

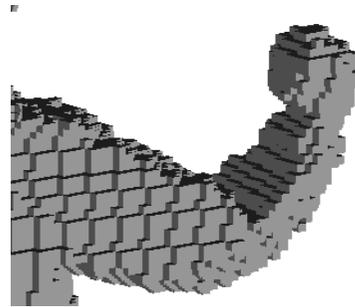


図 11 上腕のボクセル再構成の例 (復元精度 12.5 mm)

Fig. 11 Example of arm reconstruction ($a = 12.5$ mm).

した。

実験室での雑多な静止背景において、数分間のインタラクティブ動作を行い、そのうち 100 フレームについて各処理に要した時間、システム全体のスループットとレイテンシを計測し、平均値を算出した(表 1)。

3.4.2 結果と考察

復元精度 $a = 25$ [mm] の場合、本システムのスループットは 15.9 [fps], レイテンシは 155 [ms] であり、インタラクティブな動作を実現できている。

同期撮影とシルエット抽出の処理時間は復元精度に依存していない。一方で、ボクセル再構成とポリゴン生成の処理時間は復元精度の向上とともに増加している。 $a = 12.5$ [mm] と 25 [mm] の場合を比較すると、総ボクセル数は 8 倍となるが、ボクセル再構成の処理時間は約 2 倍の増加に抑えられていて、境界ボクセル法の高速度処理の効果が現れている。

$a = 12.5$ [mm] の場合、ポリゴン生成処理が最大のボトルネックとなっている。これは、アバタを構成するポリゴン数の増加にともなう、ジオメトリ計算およびレンダリングの処理量増加に起因していると考えられる。さらに復元精度を向上させるには、グラフィックカードをより高性能なものに置き換える等の対策が必要となる。

また、拳を握った上腕のボクセルデータ(図 11)から、最小部品である手首の形状が復元できていることを確認できる。

より多くのカメラを使用する場合には、シルエット抽出とボクセル再構成の処理時間がカメラ数に線形比例して増加するため、現状のシステム性能を維持するには PC の演算能力を向上させる必要がある。また、IEEE1394 バスはアイソクロナス転送モードで 320 Mbps の転送能力を持つため、現在の画像データ (320 × 240 × 16 bit × 15.9 fps) で最大 17 台までのカメラ画像を滞りなく PC へ転送することができる。それ以上の台数に対応するには、IEEE1394 バスを複数に分割する等の対策が必要となる。

4. ま と め

本論文では、計測対象となる物体を既知サイズの部品で構成された関節物体とモデル化できる場合に、物体の最小姿勢と最小部品から決まるパラメータを使って、計測空間を分割した部分空間と物体との重なり判定を正確かつ高速に行うことで、ボクセルサイズで規定される復元精度を保証できる視体積交差法の高速アルゴリズムについて述べた。

また、運動している人物の動きをカメラ撮影したシルエット画像から、その形状をボクセルで 3 次元再構成し、ポリゴンベースの仮想世界に取り込んで仮想物体とインタラクションを行うシステムを試作した。試作システムでは、復元精度を 12.5 [mm] から 25 [mm] へ向上させた場合、総ボクセル数は 8 倍となるが、ボクセル再構成の処理時間を 2 倍程度に抑えることができた。高速処理と正確な復元精度が求められるシステムにおいて、境界ボクセル法が有効であることを明らかにした。

参 考 文 献

- 1) Okutomi, M. and Kanade, T.: A Multiple Baseline Stereo, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Lahaina, Maui, Hawaii, pp.63-69, IEEE Computer Society (1991).
- 2) Saito, H., Baba, S., Kimura, M., Vedula, S. and Kanade, T.: Appearance-Based Virtual View Generation of Temporally-Varying Events from Multi-Camera Images in the 3D Room (1999).
- 3) Moezzi, S., Tai, L. and Gerard, P.: Virtual View Generation for 3D Digital Video, *IEEE MultiMedia*, Vol.4, No.1, pp.18-26 (1997).
- 4) Wada, T., Wu, X. and S. Tokai, T.M.: Homography Based Parallele Volume Intersection: Toward Real-Time Volume Reconstruction Using Active Camera, *Proc. CAMP2000 Computer Architecture for Machine Perception*, pp.331-339 (2000).
- 5) 濱崎, 吉田, 重永: 多視点シルエット画像からの高速な 3 次元形状復元手法, 第 7 回画像センシングシンポジウム講演論文集, pp.59-64 (2001).
- 6) Potmesil, M.: Generating Octree Models of 3D Objects From Their Silhouettes in a Sequence of Images, *Computer Vision, Graphics and Image Processing*, Vol.40, pp.1-29 (1987).
- 7) Szeliski, R.: Rapid Octree Construction from Image Sequences, *CVGIP: Image Understanding*, Vol.58, No.1, pp.23-32 (1993).
- 8) Cheung, G.K. and Kanade, T.: A Real time System for Robust 3D Voxel Re-construction of Human Motion, *Proc. CVPR'2000*, pp.714-720 (2000).
- 9) 濱崎, 吉田: 部分空間の境界面と関節物体の交差判定による高速 3 次元再構成, 第 6 回画像センシングシンポジウム講演論文集, pp.231-236 (2000).
- 10) 濱崎, 吉田, 重永: 多視点カメラ画像からの高速なポリゴン形状復元手法を用いたインタラクティブシステム, 情報処理学会研究報告 CVIM-129-8, pp.55-62 (2001).
- 11) Tsai, R.: A Versatile Camera Calibration Technique for High-Accuracy 3-D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses, *IEEE Journal of Robotics and Automation*, RA-3,4, pp.323-344 (1987).
- 12) 岡松, 大橋, 乃万, 松尾, 江島: 奥行き情報を利用した人物追跡システム, 電子情報通信学会研究会 PRMU99-210, pp.45-52 (2000).
- 13) Lorensen, W.E. and Cline, H.E.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm, *Computer Graphics*, Vol.21, No.4, pp.163-169 (1987).
- 14) Shekhar, R., Fayyad, E., Yagel, R. and Cornhill, J.F.: Octree-Based Decimation of Marching Cubes Surfaces, *IEEE Visualization*, pp.335-342 (1996).

(平成 14 年 3 月 6 日受付)

(平成 14 年 9 月 12 日採録)

(担当編集委員 向川 康博)



濱崎 省吾

1991年九州大学大学院工学研究科機械工学専攻修士課程修了。同年松下電器産業株式会社入社。コンピュータビジョン，ヒューマンインタフェースに関する研究開発に従事。

2001年第7回画像センシングシンポジウム優秀論文賞受賞。日本バーチャルリアリティ学会会員。



重永 信一（正会員）

1982年九州大学大学院総合理工学研究科情報システム専攻修士課程修了。同年松下電器産業株式会社入社。自然言語処理技術の開発，知的学習支援システムの開発，人物・顔を対象とする画像処理技術の開発等に従事。人工知能学会会員。

を対象とする画像処理技術の開発等に従事。人工知能学会会員。



吉田 裕之（正会員）

1984年電気通信大学電気通信学部卒業。現在，松下電器産業株式会社マルチメディア開発センターに所属。知的CAI，コンピュータビジョン，ヒューマンインタフェースに関する研究開発に従事。人工知能学会会員。

する研究開発に従事。人工知能学会会員。
