スケルトンを用いた Time-Varying Mesh からの動き抽出と類似動作検索

Motion Estimation for Time-Varying Mesh and Similar Motion Retrieval

正会員 唯野 隆一[†], 正会員 山崎 俊彦^{††}, 正会員 相澤 清晴^{††}

Ryuichi Tadano[†], Toshihiko Yamasaki^{††} and Kiyoharu Aizawa^{††}

Abstract We propose an algorithm for motion extraction from Time-Varying Mesh (TVM). TVM is a sequence of 3D models made for real-world dynamic objects. TVM store detailed information about 3D objects such as their shape, color, and motion. Therefore, TVM is an attractive technology for next-generation multimedia. However, the management of a TVM database, which is used for archiving and retrieving, is a significant problem because it is difficult to locate and track feature points in TVM. For efficient and effective archive and retrieval systems, motion extraction and tracking is essential. In our method, skeletons are extracted from TVM using a Reeb graph, and motion tracking is achieved by fitting a reference skeleton to the other skeletons. We developed a geodesic function using principal component analysis (PCA) for fast and noiseless skeleton extraction. Robust fitting was achieved by implementing an end-node tracking strategy. In addition, we have developed an efficient motion retrieval system compatible with conventional motion capture data using the extracted skeletons. As a result, we have developed a low-cost query generation technique using motion-capture systems and cross search between TVM and motion capture data.

キーワード: Time-Varying Mesh, 動き抽出, 類似動作検索

1. ま え が き

近年、複数台のカメラで撮影した多視点映像から、高精細な動的3次元メッシュモデル（以後、本論文では Time-Varying Mesh, TVM と呼ぶ）を生成する研究が盛んに行われている^{1)~4)}。TVM は、従来の CG による3次元オブジェクト合成やモーションキャプチャによる3次元の動き情報取得とは異なり、撮影した3次元シーンを実写ベースで再現したものであり、人間や動物といった実世界の物体の形・色はもちろん洋服のしわにいたるまで忠実に再現することができる。同じ実写ベースの3次元映像技術としては、複数視点の映像をもとに中間視点の映像を補完する Image-Based Rendering (IBR)^{5)~13)} と呼ばれる手法があ

る。これらは TVM とは異なり、シーンの3次元形状を推定しないため、任意視点の映像を高速に生成できるという利点がある一方、視点の移動範囲が制限されやすく、高品質な映像を生成するには、撮影用のカメラを密に並べる必要があるなどの制約がある。これに対して TVM は、IBR の手法に比べて比較的粗に配置されたカメラで撮影可能で、さらに視点移動の自由度が高い。また、3次元形状を持っているため仮想空間と動的3次元モデルを融合した複合現実感の実現¹⁴⁾が容易であるなど、インタラクティブ性が高い。

今後、大規模な TVM のデータベースを構築して実用的に利活用できるようにするためには、検索技術の開発が必要不可欠である。しかしながら TVM の検索技術はまだあまり研究例がなく、我々の知る限りでは山崎らによる研究¹⁵⁾¹⁶⁾が最初の報告例である。彼らは Osada らによって開発された Shape Distribution¹⁷⁾を改良し、TVM の動きセグメンテーションから類似動作検索に至るまで総合的な技術開発を行った。TVM を構成する3次元メッシュモデルから形状に応じた特徴ベクトルを抽出することで、TVM 同士の類似動作検索としては一定の成果が得られたものの、クエリとして TVM を用いる必要があったため、新たなクエリを作成するためには、再び TVM を撮影する必要があると

2007 年 7 月 10 日受付, 2008 年 1 月 16 日再受付, 2008 年 2 月 8 日採録

[†] 東京大学 大学院 新領域創成科学研究科 基盤情報学専攻

(〒113-8656 文京区本郷 7-3-1, 03-5841-6761)

^{††} 東京大学 大学院 情報理工学系研究科 電子情報学専攻

(〒113-8656 文京区本郷 7-3-1, 03-5841-6761)

[†] Department of Frontier Informatics, Graduate School of Frontier Sciences, The University of Tokyo

(7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan)

^{††} Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo

(7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan)

いう問題があった。現段階では、TVM の生成には莫大な演算コストと人手による手間が必要なため、余り現実的とは言えない。

一方本論文では、TVM シーケンスから被写体の動きの情報を抽出し、その情報を用いることで従来からあるモーションキャプチャ技術と TVM との融合を図る。被写体の各関節の動きを得ることができれば、モーションキャプチャと同等のデータが得られたことになり、モーションキャプチャデータのために既に開発されている多数の検索技術^{18)~21)}、がそのまま TVM に適用可能となる。さらに、検索のクエリとしてモーションキャプチャデータを用いることのできるため、新たなクエリ作成のためのコストも大幅に削減することができる。

TVM のように、マーカによる特徴点のないデータから被写体の動きの情報を抽出する技術は、一般にマーカーレス・モーションキャプチャと呼ばれ、多くの研究が行われている^{22)~28)}。文献 22)~25) は何らかの形で人体の構造を表す多関節モデルを用意し、それを撮影されたデータに対して当てはめることで被写体の動きを推定するものであり、これらの手法では、主にボリュームデータと人体モデルのフィッティングを行うが、データ量と自由度の大きさのために演算コストが大きくなり、さらに局所解に陥った際の復帰が難しいという問題を抱えている。その他には、文献 26)27) のように予め様々な動きのテンプレートを用意して、テンプレートマッチングによって動きの推定を行う手法や、文献 28) のようにスケルトンを用いて、ボトムアップに動きと関節構造の推定を行うものなどがある。

本論文では、動き抽出を行うにあたって 3 次元モデルから Reeb graph²⁹⁾ を用いてスケルトンの抽出を行い、スケルトン上で動きの推定を行った。具体的には、ある 1 フレームのスケルトンを時間的に隣接するフレームのスケルトンに対してフィッティング (変形) を行い、その結果をさらに隣のフレームに対して、フィッティングしていくというアプローチを取った。自動的に生成されるスケルトンを直接用いることで、動きの追跡開始時の初期位置合わせを行う必要がなくなり、さらに被写体の構造を特に仮定しないため、バットのような物を持ったシーンも扱うことが可能となった。また、スケルトン抽出においては高速かつロバストなスケルトン抽出法を開発し、Hilaga らによる Integral geodesic distance に基づく Reeb graph³⁰⁾ と比べ、約 100 倍の高速化を実現した。文献 28) においてもスケルトンの変形を用いた動きの推定が行われているが、人体の構造を強く仮定したルールに基づいて処理を行うため、物を持ったシーンに対応できない点、スケルトンの変形手法などが本論文とは異なる。

本論文の構成は以下の通りである。まず第 2 章で筆者らが扱おうとしている TVM のデータ構造について述べる。第 3 章では動き抽出のアルゴリズムについて詳しく説明し、第 4 章で類似検索の手法について述べる。第 5 章で実験結

図 1 TVM の例。モデルの頂点座標、頂点同士の結線関係 (エッジ)、頂点 (もしくは三角パッチ) の色の 3 種類のデータからなる。

Example of TVM. The 3D mesh model consists of three kinds of data: coordinates of vertices, connection, and color.

果を示し、第 6 章にて結論を述べる。

2. Time-Varying Mesh

本論文で使用する TVM データは文献 2)31) の手法により生成されたものである。同期がとれた 22 台の取り囲みカメラにより撮影された多視点映像から、視体積交差法とステレオマッチングによって生成されている。

TVM は 2 次元映像と同様に 3 次元メッシュモデルによる「フレーム」からなっており、連続的に再生することにより 3 次元映像となる。各フレームは図 1 に示すように、モデルの頂点座標、頂点同士の結線関係 (エッジ)、頂点 (もしくは三角パッチ) の色の 3 種類のデータからなっている。

TVM は被写体の多くが衣服を着た人物など非剛体であることなどを理由に、各フレームは独立に生成され頂点数や結線関係も保存されない。そのため、特徴点の位置特定ならびに追跡が困難であるという特徴をもつ。松山らによって前フレームのモデルを変形しながら現フレームを生成することで、頂点数や結線関係を保存したまま 3D モデルを構築する手法も開発されているが³²⁾、数フレームに 1 回定期的に頂点数と結線関係の初期化を行っており、1 フレームの処理に数時間を要するという演算コスト上の問題も抱えている。

3. 動き抽出

3.1 処理の流れ

複数のカメラ画像から復元された人物の 3 次元モデルを対象として、動きの追跡を行う手法はこれまでに複数提案されているが^{22)~28)}、いずれもあらかじめ追跡用の人物モデルを用意しているか、または対象が人物であることを強く仮定して処理を行っていた。これらの手法では、物を持ったようなシーンを扱うことが難しく、さらに追跡の前に人物モデルの初期位置合わせが必要なことが、ユーザへの大きな負担となっていた。そこで本論文では、Reeb graph²⁹⁾ によるスケルトンを用いた動きの抽出の手法を提案する (図

図2 参照スケルトンの変形による動き解析の流れ
Process flow of motion analysis using reference skeleton.

2). 対象となる3次元モデルから直接抽出されるスケルトンを用いることで初期位置合わせが不要となり、さらに人物であることを特に仮定しない処理が可能となる。

本手法では、まず最初にTVMの各フレームの3次元モデルからスケルトンを抽出する(図2上段)。得られたスケルトン群の中から、ユーザは被写体の構造を最も良く表していると思われるスケルトンを選択する(図2中段)。ここで選ばれたスケルトンを参照スケルトンと呼び、以後の処理では参照スケルトンを用いて被写体の動きを表現する。具体的には、参照スケルトンとして選択されたフレームから始まり、その前後のフレームのスケルトンに対して、参照スケルトンをフィッティング(変形)していく。変形されたスケルトンをさらに隣のフレームのスケルトンに対して、フィッティングするということを繰り返すことで、全フレームのスケルトンの形状を参照スケルトンの変形によって表す(図2下段)。これによってシーケンス中の被写体の動きが参照スケルトンによって表され、動きの情報を得ることが可能となる。なお、参照スケルトン以外のフレームのスケルトンは、フィッティングの目標となる形状を表すことから、目標スケルトンと呼ぶことにする。参照スケルトンの選択をする際には、被写体の構造をきれいに反映しているスケルトンを得るために、手足が胴体から離れているようなフレームを中心に観察する。このようなフレームが存在するシーケンスであれば、比較的簡単に参照スケルトンを得ることができる。また、参照スケルトンを選択すると同時に特徴点の設定を行う。特徴点はモーションキャプチャデータとの親和性を確保するために行うもので、設定は手動で行う。

ここで参照スケルトンを選び、フィッティングによってその他のフレームの姿勢変化を追う理由は、スケルトンが必ずしも全てのフレームで正確に抽出されるとは限らないからである。体の異なる部位同士が接触している姿勢の場合は、メッシュの位相が異なり(例えば、手が胴体に接している場合、手と胴体は分離できず、一体化したモデルとな

図3 浮遊している頂点の例
Example of floating vertices.

る)、スケルトンもそれとともなって異なるトポロジーのものが抽出されてしまう。さらに、後述するように被写体と同じような姿勢をしていても、視体積交差法による3次元形状復元の際のノイズでスケルトンにノイズが発生することがあるなど、得られるスケルトンは必ずしも一定ではない。そのため、被写体の構造を最も良く表しているスケルトンをユーザが一つ選択し、それを参照モデルとして変形していくアプローチを取った。また、ユーザは参照モデルを選択するだけで良いので、モデルベースの手法のような初期位置を配置するための作業が必要ないばかりか、被写体に合ったモデルそのものを用意する必要もない。

3.2 前処理

既に述べた通り、TVMは多視点のシルエット画像を元に生成される。このシルエット画像は、照明の反射や背景色の写り込み等によりクロマキー処理がうまく行われず、ノイズを含むことがある。ノイズを含んだシルエット画像を元に3次元モデルを生成すると、被写体の表面近くや体の内部にゴミのような頂点が浮かんでしまうことがある(図3)。このような浮遊した頂点は、あらかじめ除去しておいた方が以後の処理が行いやすくなる。本論文では具体的に以下の手順で浮遊した頂点の除去を行った。

- (1) $i = 0$ とする
- (2) メッシュを構成するすべての頂点をリスト L に格納する
- (3) L からランダムに1頂点 v を選び、 L から v を削除する
- (4) v からリンクを辿って到達できるすべての頂点を L から削除し、新しいリスト l_i に格納する
- (5) i の値を1増やし、 L が空になるまで3~4を繰り返す
- (6) $l_0, l_1, l_2, \dots$ のうち、最も要素数が大きいものを除外する

3.3 スケルトンの抽出

3次元モデルからのスケルトンの抽出法としてはReeb graph²⁹⁾やボクセルの細線化³³⁾があるが、ボクセルの細線化は、3次元モデルの表面形状に非常に敏感で、わずかな凹凸によって、たくさんのひげ状のスケルトンが発生してしまう。一方、Reeb graphはモデルの分割に用いる関数によっては、多少の表面形状の違いに影響されずにスケ

図 4 高さを μ として与えたときの Reeb graph
Height function as μ .

ルトンの抽出が可能である。したがって本論文では Reeb graph を採用する。以下ではまず一般的な Reeb graph の定義について述べ、さらに本論文で提案するモデル表面上の関数について説明する。最後にスケルトン生成時に発生するわずかなノイズの除去について述べる。

Reeb graph とは、物体のトポロジーを表したもので、これまでに様々な分野に用いられている。静止 3 次元モデルの類似検索³⁰⁾³⁴⁾、人物モデルのパーツ切り分け^{35)~37)}、3 次元モデルの位相的モーフィング³⁸⁾ 等が挙げられる。Reeb graph は以下のようにして定義される：

$\mu : C \rightarrow \mathbf{R}$ をオブジェクト C 上に定義された連続関数とする。同値関係 $(X_1, \mu(X_1)) \sim (X_2, \mu(X_2))$ が次の条件で成り立つとき、Reeb graph は $C \times \mathbf{R}$ 内の μ のグラフの、 $\sim$ による商空間である。

- $\mu(X_1) = \mu(X_2)$ かつ
- X_1, X_2 は $\mu^{-1}(\mu(X_1))$ の連結成分である。

例えば、ドーナツ状のモデルに対して μ を高さ関数として定義した際の Reeb graph は図 4 のようになる。モデルが μ の値に従って切断され、その断片がそれぞれ図中の小さい丸で示されるノードと対応する。これらのノードが切断前のモデルの接続関係に従って接続され、グラフ構造を作る。これと同じ要領で、本論文で扱う 3 次元モデルから実際に Reeb graph を生成する際の処理は、以下のような手順で行われる。

- (1) メッシュモデルを構成する各頂点 $\mathbf{v}$ について、 $\mu(\mathbf{v})$ の値を計算する。
- (2) $\mu(\mathbf{v})$ の値に従って、ある間隔でメッシュを分割する。
- (3) 分割されたメッシュ一つ一つに 1 ノードを割り当て、分割前に接続していたもの同士の間エッジを張り接続する。

モデル表面上の関数 μ として様々なものが考えられるが、どのような関数を用いるかによって、得られる Reeb graph の性質は大きく左右される。本論文で目標としている動き抽出のためのスケルトンとして用いるに当たっては、被写体の回転・平行移動・姿勢変化に対して不変であることが望ましい。そのような性質を持つものとしては、Hilaga らによって提案された Integral geodesic distance³⁰⁾ が挙げられる。Integral geodesic distance は以下のようにして定義

図 5 Integral geodesic distance の値の分布
Value distribution of the Integral geodesic distance.

される。

$$\mu(\mathbf{v}) = \sum_{\mathbf{p} \in S} g(\mathbf{v}, \mathbf{p}) \quad (1)$$

ここで、 $g(\mathbf{v}, \mathbf{p})$ はモデル表面上の 2 点 $\mathbf{v}, \mathbf{p}$ を結ぶ測地線距離を表している。ある 2 点間の測地線距離とは、2 点間を結ぶモデル表面上の最短経路の長さである。なお、測地線距離は Dijkstra のアルゴリズムを用いて求めることができる。Integral geodesic distance の値の分布の様子を図 5 に示す。色が暗いほど値が小さく、明るい色ほど大きな値であることを示している。モデルの中心付近ほど値が小さくなり、末端部分に行くに従って値が大きくなる傾向がある。この値の分布は、座標系内におけるモデルの向きには依存せず、さらに姿勢が変化しても、同一の被写体であればその分布はほとんど変化しないという特徴を持つ。このようなメリットを持つ一方で、この Integral geodesic distance は測地線距離の演算を何度を行うため、演算コストが非常に高いという欠点を持つ。(1) 式の通りにそのまま演算しようとする、頂点数の数と同じ回数だけの測地線距離の演算が必要となってしまう。Hilaga らはモデル表面上をクラスタリングし、これらのクラスタ単位で測地線距離の演算を行うことで演算量の削減を行っているが、それでも 100 ~ 200 回程度の測地線距離の演算が必要となるため、依然として演算コストは高い。

これに対し、本論文では被写体の姿勢変化に対してロバストな性質をできるだけ保ちつつ、演算コストの小さい関数を提案する。まず、メッシュモデルを形成する全頂点に対して、PCA(主成分分析)を行い、第一主成分を求める ($\mathbf{n}_{1st}$)。主成分分析を行うのは、座標系内におけるモデルの向きに結果が依存しないようにするためである。次に、この $\mathbf{n}_{1st}$ を法線に持ち、かつモデルの重心を通る平面を求める。このようにして得られた平面によってモデルを切断し、その切断面からの測地線距離を各頂点における μ の値とする。すなわち、 μ は (2) 式で表わされる。

$$\mu(\mathbf{v}) = \min_{\mathbf{p} \in C} g(\mathbf{v}, \mathbf{p}) \quad (2)$$

ここで、 C は切断面と各エッジの交点の集合であり、 $g(\mathbf{v}, \mathbf{p})$ は (1) と同様モデル表面上の 2 頂点 $\mathbf{v}, \mathbf{p}$ の間の測地線距離である。なお、場合によっては切断面が複数生じることもあるが、その際は最も大きな切断面との交点群を C とす

図 6 提案手法による Reeb graph 生成の流れ
Reeb graph generation of proposed method.

図 7 ノイズノードの例. 3 次元モデル上におけるとげ状のノイズにより、スケルトン上に余計なノードが発生することがある。

Example of noise node. Spinal noise on the 3D model could generate some excrement nodes in the skeleton.

る. この関数を用いて Reeb graph を生成するまでの流れを図 6 に示す.

この演算は次の手順で実現できる. まず, 切断面と交差しているエッジを求め, それらを構成する頂点に対して, 切断面との交点からの距離をそれぞれ割当てる. 次にこれらの頂点を開始点として, Dijkstra のアルゴリズムを用いてその他の頂点までの測地線距離を求める. すなわち測地線距離の演算は 1 回だけ行えばよく, Integral geodesic distance に比べて演算量は大幅に小さくなる. PCA によって向きを補正した平面でモデルを切断しているため, モデルの平行移動や回転に対して不変であり, さらに測地線距離に基づいた値であるため, 被写体の姿勢変化によるモデルの変形に対してもロバストであるという特徴を持つ.

ボクセルの細線化ほど顕著ではないが, 抽出したスケルトンには, しばしばノイズのようなノードが含まれることがある. これは TVM が実写ベースであるために, 3 次元モデルの表面に大きなノイズが発生する場合があることに起因する (図 7). そこで本手法では, スケルトン抽出の後処理として, ノイズノードの除去を行う. 以下の条件を満たすノードをノイズノードと定義し, 図 7 (c) に見られるようなノードを取り除いた.

- エッジを一つだけ持つ
- 接続先のノードが三つ以上のエッジを持つ

3.4 フィッティングによる動き推定

本手法では, 参照スケルトン全体のフィッティングに先

立ち, 参照スケルトンにおいて足先やバットの先端などに相当する端点となる端ノードの動きを追跡する. 端ノードは, スケルトンを構成するノードのうち, 一つしかエッジを持たないノードとして定義する. 端ノードの数は高々数個なので, その追跡は比較的正確に行うことができる. 端ノードの追跡結果を用いて, スケルトン全体のフィッティングを低コストで行うことが可能となる.

参照スケルトンの端ノードを $E^r = \{e_1^r, e_2^r, \dots, e_m^r\}$, フィッティング先のスケルトン (目標スケルトン) の端ノードを $E^d = \{e_1^d, e_2^d, \dots, e_n^d\}$ とする. ここで, m, n はそれぞれ, 参照スケルトンと目標スケルトンの持つ端ノードの数を表す. 端ノードの追跡の手順は以下の通りとなる.

- (1) E^r, E^d の中から距離が最小となるような端ノードのペア $p = \{e_i^r, e_j^d\}$ を求める.
- (2) e_i^r の移動先を e_j^d とする.
- (3) e_i^r, e_j^d を, それぞれ E^r, E^d から削除する.
- (4) (1) ~ (3) を E^r の要素がなくなるまで繰り返す.

以上のようにして端ノードの追跡が行われるが, TVM の全てのフレームを通して, 得られるスケルトンの端ノードの数は必ずしも一定ではない. 参照スケルトンの端ノードの数が少ない場合, すなわち $m < n$ となる場合は, すべての $e_i^r \in E^r$ に関して移動先が確定する. ペアに漏れた目標スケルトンの端ノードは, 前節におけるノイズノードの削除の処理において削除しきれなかったノイズとみなされ, 無視される. 一方, $m > n$ となる場合は, 参照スケルトンの端ノードの中に移動先が確定しないノードが存在する. このような場合には, ノード位置の推定値として以下を用いることで対応する.

$$\mathbf{p}'_t = \mathbf{p}_{t-1} + \frac{\mathbf{v}_{t-1}}{2} \quad (3)$$

ここで, $\mathbf{p}$ は端ノードの 3 次元空間内における位置, $\mathbf{p}'$ は推定された端ノードの位置, $\mathbf{v}$ はその前のフレームの追跡結果から得られる端ノードの移動速度を表す. なお, $\mathbf{v}_{t-1}$ を 2 で割っているのは, 端ノードが消滅した状態が何フレームも続いた際にノードの推定位置が移動し続けて, あり得ない位置になってしまうのを防ぐためである.

このように端ノードが消滅するのは, 被写体の体の一部同士が接触することが原因であることが多い. 姿勢の変化によって消えていた端ノードが再び現れた場合には, その端ノードの位置を新たな移動先として用いる. この復帰処理は, 正しく構造を表現していると思われるスケルトンを人手で選択し, 常にそれをリファレンスとして用いるからこそ可能である.

上記の処理により異なるフレーム間での端ノード同士の対応関係が明らかになった. さらに, Reeb graph 生成時の μ の値が最小であるセグメントに対応するノード (中心ノードと呼ぶ) 同士の対応も明らかである. そこで, スケルトン全体のフィッティング処理は, 中心ノードと各端ノードを繋ぐパス毎に行う. それぞれのパスについて, 端ノ

図 8 パスの分岐と複数パス上に存在するノード。バットの先端から中心ノードに向かうパス (実線) は途中で分岐している。また、被写体の胸に相当する部分ではバットの先端からのパスと頭からのパス (点線) が同じノードを共有している。

Branched path and nodes on multi paths. The path from the tip of the bat to the center node branches. Two paths from the tip of the bat and from head share nodes around chest of batter.

図 9 ノードのマッピング
Node mapping.

図 10 ポテンシャルフィールドの値
Value of potential field.

ドの追跡が成功した場合と (3) 式による推定値を用いた場合によって以下の通りに分けて処理を行う。なお、図 8 に示すような複数のパス上に存在するノードの位置については、各パス毎にフィッティング処理を行った結果を平均して用いる。

(1) ノードのマッピング

参照スケルトンのパスに含まれるノードを、目標スケルトンの該当するパス上にマッピングする。この際、参照スケルトン側のエッジの長さの比を保つように目標スケルトンのパス上にノードの再配置を行う。また、バッターのモデルのように、両手が接触したものを参照スケルトンとして用いる場合などは、中心ノードと端ノードを繋ぐパスが複数存在する (図 8)。このような場合、中心ノードから各ノードまでのベクトルを足し合わせ、それをノード数で正規化したものを比較することで、参照スケルトンが目標スケルトンのどのパスに対応するかを決定する。

(2) ポテンシャルフィールドにおけるエネルギー最小化参照スケルトンの端ノードに対応するノードが目標スケルトン側に見つからず、(3) 式による推定値を用いた場合は、対応するパスも目標スケルトン側には存在しない。上で述べたノードのマッピングが行えないため、目標スケルトンの周囲にエネルギー関数を定義し、そのコストを最小化することでできるだけ目標スケルトンに近づくように参照スケルトンのパスを変形する。ここで、エネルギー関数としては³⁹⁾で用いられているガウシアンタイプのものを (4) 式の通り 3 次元に拡張して用いる。

$$E(\mathbf{p}) = \min_{\mathbf{k} \in \mathbf{K}} \left\{ 1 - \exp \left(-\frac{d^2(\mathbf{k}, \mathbf{p})}{2\sigma_{\mathbf{k}}^2} \right) \right\} \quad (4)$$

ここで、 $\mathbf{K} = \{\mathbf{k}\}$ 、 $\mathbf{k}$ は目標スケルトン上の点、 $d(\mathbf{l}, \mathbf{m})$ は 2 点 $\mathbf{l}, \mathbf{m}$ の間のユークリッド距離、 $\mathbf{k}, \mathbf{p}$ は 3 次元位置ベクトルを表す。また、 $\sigma_{\mathbf{k}}$ は (点 $\mathbf{k}$ から 3 次元メッシュモデル表面までの最短距離)/3 である。すなわち、このポテンシャルフィールドは、ガウシアン関数の分散をスケルトンからメッシュ表面までの距離に応じて変化させており、単にスケルトンからの距離を反映しているのではなく、3 次元メッシュモデルの体積を考慮した値になっていることがわかる。ここで、 $\sigma_{\mathbf{k}}$ の値を 0.4 とした場合のポテンシャルフィールドの値を図 10 に示す。ここではスケルトンからモデル表面までの距離を $0.4 \times 3 = 1.2$ としているが、 $E(\mathbf{p})$ の値がちょうどメッシュモデルの表面、すなわち 1.2 付近で 1 に収束していることが確認できる。

コストを最小とするようなパスは、各ノードの周辺に探索範囲を設定して動的計画法を用いて求める。ここで、各ノードの探索範囲はパスの両端の動きを用いて予測される仮のパス上に設定される。仮のパスは中心ノードの動きと端ノードの動き推定の結果を用いて、(5) 式に示す重み付き線形和で予測される。

$$\mathbf{p}' = \mathbf{p} + \frac{(d_c \cdot \mathbf{mv}_c + d_e \cdot \mathbf{mv}_e)}{(d_c + d_e)} \quad (5)$$

ここで、 $\mathbf{p}$ は初期位置を推定しようとしているノードの座標、 $\mathbf{p}'$ はその推定値、 $\mathbf{mv}_c, \mathbf{mv}_e$ は中心ノード・端ノードの推定された動き、 d_c, d_e は中心ノード・端ノードからのパス上の距離を表している。なお、予測された仮のパス上の各ノードについて設定される探索範囲の大きさとして、 $\mathbf{mv}_c, \mathbf{mv}_e$ のうち大きい方を採用する。図 11 に、各ノードの初期位置の推定の様子を示す。

4. 類似動作検索

4.1 特徴ベクトル

TVM に関する類似動作検索の技術はまだ報告例が少ないものの、モーションキャプチャデータにおいては、多数の類似動作検索に関する研究が行われている。第 3 章の動き解析において、モーションキャプチャのマーカに相当する部分 (関節等) を参照フレームにおいてユーザが適切に指定

図 11 探索範囲の設定
Setting search area of path prediction.

図 12 類似動作検索の特徴ベクトル
Feature vector of similar motion retrieval.

すれば、モーションキャプチャと同等のデータを得ることができる。これを利用して、モーションキャプチャデータのために開発された類似動作検索の手法が適用可能である。

いくつかある手法のうち、本論文では最も一般的なものとして Chiu らの手法¹⁹⁾をベースに類似動作検索を行うこととした。主な理由としては、特徴量ベースの連続値を用いた手法であり、一部動き推定に誤りがあっても適用可能であること、実装が簡単であることが挙げられる。モーションキャプチャデータの類似動作検索としては Müller らの手法²¹⁾のような論理値を用いたものがあるが、本論文の動き解析はモーションキャプチャほどの精度が保証されていないため、動き推定誤りに敏感になってしまうおそれがある。

Chiu ら¹⁹⁾はモーションキャプチャデータの各フレームにおいて、図 12 に示すような九つの体節に注目した。これらの体節は胴体の体節を根として木構造で表現できるが、それぞれの親となる体節に対する二つの角度(θ, ϕ)を用いて、2次元空間内のベクトルとして表すことができる。これらのベクトル九つを合わせて1フレームを表す特徴ベクトルとし、DP マッチングを用いて類似度の演算を行った。

本論文の動き抽出手法を用いると、各体節の両端の3次元座標(x, y, z)を得ることができる。そこで、一方の端から他方を引くことによって各セグメントの向きを表すベクトルを求め、それらを九つ合わせて1フレームの特徴ベクトルとした。なお、各セグメントのベクトルは長さが1になるように正規化を施した。

4.2 DP マッチング

各フレームから特徴ベクトルを取り出した後は、Dynamic Programming (DP) マッチングを用いてシーケンス間の類似度を計算する。DP マッチングは異なる長さのシーケンスを比較する際にも有効な手法で、音声データや映像データのマッチングなどに広く用いられている。

検索対象となるデータベースに含まれるシーケンスの集合を Y 、クエリを Q とする。ここで、これらは特徴ベクトルのシーケンスに変換されているものとする。データベースの中の i 番目のシーケンスを $Y^{(i)}$ とすると、 $Q, Y^{(i)}$ は以下の通りに表される。

$$Q = \{q_1, q_2, \dots, q_s, \dots, q_l\} \quad (6)$$

$$Y^{(i)} = \{y_1^{(i)}, y_2^{(i)}, \dots, y_t^{(i)}, \dots, y_m^{(i)}\} \quad (7)$$

ここで、 $q_s, y_t^{(i)}$ はそれぞれ $Q, Y^{(i)}$ の s 番目、 t 番目の特徴ベクトルを表し、 l, m はそれぞれ $Q, Y^{(i)}$ のフレーム数を表す。さらに、二つのフレームの特徴ベクトル $q_s, y_t^{(i)}$ の間のユークリッド距離 $d(s, t)$ を以下の通りに定義する。

$$d(s, t) = \|q_s - y_t^{(i)}\| \quad (8)$$

二つのシーケンス $Q, Y^{(i)}$ の間の距離 (非類似度) D は以下のようにして計算される。

$$D(Q, Y^{(i)}) = \frac{c(l, m)}{\sqrt{l^2 + m^2}} \quad (9)$$

ここで、 $c(l, m)$ は以下のように定義される。

$$c(s, t) = \begin{cases} d(1, 1) & \dots l=m=1 \\ d(s, t) + \min\{c(s, t-1), \\ c(s-1, t), c(s-1, t-1)\} & \dots \text{otherwise} \end{cases} \quad (10)$$

なお、 $d(s, t)$ や $c(l, m)$ において $Q, Y^{(i)}$ の記号は簡単のために省略している。 $c(l, m)$ はシーケンスの長さ (フレーム数) に依存しているため、式 (9) において $\sqrt{l^2 + m^2}$ で正規化している。 D の値が小さいほど、シーケンスは互いに似ていることを示す。

5. 実験と考察

この章では提案した手法を評価するための実験を行う。実験は大きく分けてスケルトン抽出・動き解析・類似動作検索の三つである。

5.1 スケルトン抽出

ここでは、本論文で提案したモデル表面上の関数 ((2) 式) の有効性を示すため、Reeb graph 生成の際の関数 μ として提案手法を用いた場合とその他の関数を用いた場合とを、得られるスケルトンの性質およびその演算時間の観点から比較する。

(1) 得られるスケルトンの性質

実験に使用した3次元モデルを図 13 に示す。これら様々な被写体・姿勢の七つの3次元モデルに対して実験を行った。Reeb graph 生成の際の関数 μ として以下の四つを用い、これらの3次元モデルからスケルトンを抽出した。

- モデルの座標系における高さ (μ_1)
- モデルの重心からの距離 (μ_2)

図 13 実験に使用した 3 次元モデル
3D models used in the experiment.

図 14 モデルの座標系における高さ (μ_1)
Height (μ_1).

- Integral geodesic distance (μ_3)
- 提案手法 (μ_4)

それぞれの関数によって分割されたメッシュと、得られたスケルトンを図 14～図 17 に示す。各図の上段には分割されたメッシュが、下段には得られたスケルトンがそれぞれ示されている。各スケルトンにおける点は、分割されたメッシュの各領域の重心に相当する。なお、Reeb graph 生成の際には μ の値を 0 ～ 1 に正規化し、0.08 刻みで分割を行った。これらの結果から、 μ_1 と μ_2 にはそれぞれスケルトンがうまく抽出されにくい姿勢が存在するということがわかる。 μ_1 は、図 14 (e) のように高さの向きに対して垂直な方向に腕を伸ばしている場合、腕が一つのセグメントの中に含まれてしまい、結果として得られるスケルトンには腕の存在が反映されにくくなる。同様に、 μ_2 は、図 15 (f) のように重心を中心とする同心球平面に沿うような向きに腕を広げている場合は、腕が一つのセグメントの中に含まれてしまい、腕の欠けたようなスケルトンが得られやすくなる。図 15 (g) に関しても、胴体の分割のされかたにおいて同様のことが言える。この二つの関数のように 3 次元空間内にある一定の方向に分割していくような手法は、計算コストが小さい反面、その分割方向に沿うように置かれたオブジェクトのスケルトンはうまく抽出できないという欠点がある。一方、 μ_3 や μ_4 は測地線距離に基づいてモデルの分割を行うため、 μ_1 や μ_2 に見られるような問題は発生していない。特に、腕・足・バットなどの比較的細長い筒状の形をした部分については、両者とも例外なくきれいな輪切り状に分割されており、良好なスケルトンが得られている。

図 15 モデルの重心からの距離 (μ_2)
Distance from the center of gravity (μ_2).

図 16 Integral geodesic distance (μ_3)³⁰⁾
Integral geodesic distance (μ_3).

図 17 提案手法 (μ_4)
Proposed method.

図 18 胴体付近の詳細な分割結果
Detailed segmentation result around the torso.

もう一つ注目すべきなのは、 μ_3 と μ_4 の違いである。腕や足などについてはほぼ同様にモデルが分割されているが、モデルの中心部分の分割のされ方が大きく異なる。 μ_4 では、

図 19 スケルトン抽出の演算時間
Computational time for skeleton extraction.

胴体付近もその他の部分と同じような間隔で分割されているのに対して、 μ_3 では胴体全体が大きな一つのセグメントとなっている。同じ 0.08 刻みでこのような違いが出るのは、 μ_3 の値が小さくなるにつれモデル表面上での分布が広範囲になるためである。そのため、 μ_3 を用いて抽出されたスケルトンでは、胴体全体が一つのノードに対応し、腕や足には多数のノードが対応するというアンバランスな状態になっている。また、モデル中心付近の μ_3 および μ_4 の値の分布を詳しく見るために、分割の刻みを 0.03 まで細かくした結果を図 18 に示す。ここで、(a) は μ_3 、(b) は μ_4 による分割結果を表す。この図から、 μ_4 では胴体付近もその他の部位と同じように均等に分割されているのに対し、 μ_3 は腹を中心に広がる形で同心円状に分割されている。背中にも同じような同心円状の分割が見られる。このため、 μ_3 を用いた場合は、分割の刻みを細かくしすぎると胴体付近にノイズが発生してしまう。また、 μ_2 も重心から同心球状にモデルを分割しているため、分割の刻みを細かくしていったときには μ_3 と同様の問題が発生する。以上のことから、様々な姿勢において提案手法がこれらの中で最も優れたスケルトンを抽出できることが確認できた。

提案手法は、PCA を用いて測地線距離を計算するための基準となる切断面を求めるため、両手が床に接触するほど体を曲げるといった、立位姿勢から極端に異なった姿勢を取る場合には、切断面が必ずしも適切に求まらないおそれがある。我々の手元にあるデータの中では、図 13(g) が最も極端な姿勢を取っていたが、実験によりこの程度の姿勢であれば問題なく処理が行われることが確認できた。また、図 13(b) のようにゆったりとした着物を着ているようなモデルでは、体の形状が服で覆われてしまっているため、いずれの関数を用いても被写体の構造をうまく表現するようなスケルトンは得られなかった。

(2) 演算時間の比較

次に、関数の値を計算するのにかかる時間を比較する。四つの関数のうち、 μ_1 と μ_2 はほぼ無視できる時間内で演算することができるので、ここでは μ_3 (Integral geodesic distance) と μ_4 (提案手法) について比較を行う。実験には、TVM の中から一つのフレームを選択し、三つの異なる頂点

図 20 TVM データに対する動き解析結果 (batting)
Motion tracking result for TVM data (batting).

図 21 TVM データに対する動き解析結果 (walk)
Motion tracking result for TVM data (walk).

図 22 TVM データに対する動き解析結果 (rotate arm)
Motion tracking result for TVM data (rotate arm).

数 (メッシュの細かさ) でモデリングしたものを用いた。関数の値を求めるのにかった演算時間を図 19 に示す。いずれの頂点数においても、提案手法の演算時間の方が Integral geodesic distance に対して 1/100 程度に収まっていることがわかる。これは、Integral geodesic distance が 100~200 回程度の測地線距離の演算を要するのに対し、提案手法では 1 回だけ測地線距離の演算を行えばよいことに起因する。

5.2 動き解析

(1) TVM データへの適用

本手法を実際の TVM データに適用し、動きの解析を行った。この実験では batting, walk, rotate arm の三つのシーケンスを用いた。なお、これらのシーケンスは 10 fps で撮影されている。動き解析の結果を図 20~22 に示す。ここで、それぞれのシーケンスにおいて最も左に示した画像が参照フレームとして用いたものである。輪郭線の中に引かれた黒い線は、各フレームのスケルトンに対して参照スケルトンをフィッティングした結果を表し、色の付いた丸は参照スケルトンにおいて指定した追跡点を表す。なお、赤は被写体の右半身、青は左半身、緑は体の中央としている。ここで、動きの解析はスケルトンを用いて行われるため、スケルトン上の任意の点を追跡点とできることに注意されたい。追跡点は、その属しているエッジを成す二つのノードからの距離を用いてパラメータ化して保存される。

これらの図から、本手法が全体として良好に機能していることがわかる。10 fps という低いフレームレートながら、

バットのスイングのような素早い動きが正しく追えている(図 20)。動き推定の際に、処理の候補を絞った端ノードの追跡を行っていることが、このような素早い動きへの対応を可能にしている。さらに、3.4 節(1)で述べたように、参照スケルトンのエッジの長さの比を保つように変形していくことから、端ノードの追跡が成功している限りは追跡フレームを重ねる毎に誤差が蓄積されることはなく、安定して追跡が行われることがわかる。すなわち、図 20 の左から 2 番目のフレームのようにバットと頭が接触・一体化したためにスケルトンの端ノードが一つ消滅し、トポロジーが大きく変化している場合でも、そのフレームにおいては追跡誤差が発生するものの、後のフレームに大きな影響は見られない。

一方で、端ノードの追跡がうまく行かない例もわずかに見られた。図 22 では、左から 3 番目と 4 番目のフレームにおいて、右手と左手の端ノードが入れ替わって追跡されてしまっていることが確認された。これは腕を交差する場面で、両腕の端ノード同士が非常に近いところを通るため、場合によっては追跡点の入れ替わりが起きてしまうためである。以前のフレームにおける動きを使うなど予測の導入や、今後のフレームレートの向上によってこうしたノードの入れ替えの問題は回避されていくと考えられる。

以上から、本論文で提案した動き解析手法は以下の 3 点を併せ持っている点で、従来手法に対して優れていることがわかる。一つは、端ノードの追跡が正しく行われている限りは追跡誤差が蓄積しない点である。シーケンスを通して連続的に行われる追跡には限られた数の端ノードが対象となっているため、そこでは誤差は発生しにくく、文献²⁴⁾²⁸⁾で見られるような、フレームを重ねるたびに誤差が蓄積していくということはない。もう一つは、追跡開始時フレームにおけるモデルの初期位置合わせが不要である点である。これは従来手法の多くに共通の問題で、初期位置合わせは手動で行うのが一般的である。3 次元空間において、追跡モデルと被写体の位置を合わせこむのは非常に煩雑な作業であるため、被写体が決められた動作を撮影時に行うことで初期位置合わせを行っている手法²³⁾もあるが、いずれにしてもユーザにかかる負担は大きい。一方で、我々の手法では参照スケルトンを選択し、追跡点を設定するだけで初期位置合わせが不要であるという利点がある。最後は、我々の手法はモデル依存性が低いという点である。人体の多関節モデルを用いて追跡を行う手法^{22)~25)}では、被写体の体型によってモデルのパラメータを変更したり、物を持っているようなシーンではモデルに修正を加える必要がある。その点、我々の手法では被写体の 3 次元モデルから直接生成されたスケルトンを追跡に使用するため、このような問題は発生しない。実際に、何も持っていないシーンとバットを持っているシーンの両方を同じフレームワークで扱っている。生成されたスケルトンを用いるという点では文献²⁷⁾²⁸⁾も同様であるが、共に人体を強く意識した過程を置いて処

図 23 追跡精度の評価に用いた CG シーン
CG scene used for evaluation of motion tracking accuracy.

図 24 追跡点の設定の様子 (Hip の場合)
Setting tracking point (Hip).

理しているため、やはりモデル依存性が高く、物を持ったシーンなどにそのまま対応することは難しい。

(2) CG データによる精度の定量的評価

本手法の精度を評価するため、モーションキャプチャデータ*を用いて CG シーンを作成し、動きの追跡を行った。CG データの作成には、商用ソフトウェアである Poser 6 を用いた。評価には図 23 に示すような人物が歩行を行うシーンを用いた。

精度の評価は、CG モデルの体のいくつかのパーツについて、本手法で得られた推定位置と CG データから直接得られる正解位置のユークリッド距離を用いて行った。なお、本手法を適用するにあたり、CG シーンの第 1 フレームの抽出されたスケルトンにおいて、CG データから得られる各パーツの位置に最も近いスケルトン上の点を追跡点として設定し、動きの追跡を行った。追跡点の設定の様子を、Hip を例に取ったものを図 24 に示す。ここで、図左のモデルにおいて、黒い部分は Hip として定義されているパーツを表し、図右において黒の球はその重心を、白の球は設定された追跡点を表す。なお、灰色の球はスケルトンの各ノードを表している。参照スケルトン上にマッピングされた追跡点は、マッピングされたエッジの両端のノードと、そのエッジを内分する比を用いてパラメータ化される。追跡点が各フレームにおいて変形されたスケルトン上のどの位置にあるかは、このパラメータを用いて求めることができる。

評価結果の一部を図 25 に示す。シーケンス全体の様子は、両腕を大きく広げた状態から腕を下ろし図 23 のように歩き始め、何往復かした後で再び最初の姿勢に戻るといったものである。フレームレートは 30 fps で全体の長さは 743 フレームであった。ここでは、使用した人体 CG モデルの身長を 180cm として距離を計算を行っている。

* モーションキャプチャデータは <http://www.mocapdata.com> のデータを使用した

図 25 追跡精度の評価 (Abdomen, Right hand, Right collar, Right shin)
Evaluation of motion tracking accuracy (Abdomen, Right hand, Right collar, Right shin).

図 26 TVM シーケンス (walk)
TVM sequence (walk).

図 27 TVM シーケンス (rotate arm)
TVM sequence (rotate arm).

図 28 TVM シーケンス (lift arm & step)
TVM sequence (lift arm & step).

なお、使用した CG モデルの中に定義された体のパーツの総数は 73 であったが、それらのうち互いに近すぎるなど不適切なものを除いて Hip, Abdomen, Chest, Neck, head, Right_Collar, Right_Shoulder, Right_Forearm, Right_Hand, Left_Collar, Left_Shoulder, Left_Forearm, Left_Hand, Right_Thigh, Right_Shin, Right_Foot, rToe, Left_Thigh, Left_Shin, Left_Foot, lToe の 21 個に絞って追跡点として設定している。これら全体を合わせた誤差の平均は約 7.5 cm, 分散は約 4.5 であった。初期フレームにおける追跡点の設定時 (図 24 参照) に発生する誤差が平均で約 2.7 cm であったことを考慮すると良好な結果と言える。

5.3 類似動作検索

第 4 章で説明した手法を用いて、TVM とモーションキャプチャデータの間の類似動作検索を行った。TVM から walk (a)~(d), rotate arm (a)~(d), lift arm & step

図 29 TVM シーケンス (batting)
TVM sequence (batting).

図 30 モーションキャプチャデータ (walk)
Motion capture data (walk).

図 31 モーションキャプチャデータ (rotate arm)
Motion capture data (rotate arm).

(a)~(c), batting の 12 クリップを、モーションキャプチャデータからは walk 1~4, rotate arm 1~4 の 8 クリップを用いてデータベースを作成した。各動作の様子を図 26~31 に示す。各クリップのフレーム数を表 1 に示す。TVM は 10 fps, モーションキャプチャは 30 fps で撮影されている。なお、各クリップのインデックスは同種の動きをしている異なるクリップを区別するためのものであり、特別な意味は持たない。これらのシーケンスに関して、動きの類似度を計算した結果を図 32 に示す。色が暗いほどお互いの動きが似ていることを表している。図からわかる通り、TVM とモーションキャプチャデータの walk 同士や rotate arm 同士が似た動きとして判定されており、良好な結果が得られたと言える。

加えて、この結果を用いてモーションキャプチャデータをクエリとして、TVM を対象とした検索を行った際の、各クエリに対する上位 6 シーケンスを表 2 に示す。この表からも、クエリ walk3 に対して、batter が 1 位になっていることを除けば、すべてのクエリについて対応する TVM が上位を占めていることがわかる。

walk3 に対して batter が 1 位となってしまう原因を調べるために、DP マッチングの際にどのフレーム同士がマッチしているかを検証したところ、batter 側のある特定のフレーム (#29) が walk3 側の 4 割近くのフレームとマッチしていることがわかった。batter 側のある特定のフレーム (#29) が walk3 側の 4 割近くのフレームとマッチしていることがわかった。図 34 に batter と walk3 の DP マッチングの結果を示す。このフレームは、図 33 に示す

表 2 各クエリに対する上位のシーケンス
High order sequences for each query.

query	1st	2nd	3rd	4th	5th	6th
rotate arm 1	rotate arm (c)	rotate arm (b)	rotate arm (a)	rotate arm (d)	walk (a)	walk (c)
rotate arm 2	rotate arm (c)	rotate arm (b)	rotate arm (a)	rotate arm (d)	walk (a)	walk (b)
rotate arm 3	rotate arm (a)	rotate arm (c)	rotate arm (b)	rotate arm (d)	walk (c)	walk (a)
rotate arm 4	rotate arm (c)	rotate arm (a)	rotate arm (b)	rotate arm (d)	walk (c)	walk (d)
walk 1	walk (a)	walk (b)	walk (d)	walk (c)	rotate arm (c)	rotate arm (b)
walk 2	walk (b)	walk (c)	walk (d)	walk (a)	batter	rotate arm (c)
walk 3	batter	walk (b)	walk (c)	walk (d)	walk (a)	rotate arm (c)
walk 4	walk (b)	walk (d)	walk (c)	walk (a)	batter	rotate arm (c)

表 1 各シーケンスのフレーム数
Number of frame of each sequence.

sequence name	1/(a)	2/(b)	3/(c)	4/(d)
walk (TVM)	31	31	32	33
rotate arm (TVM)	50	51	52	53
lift arm & step (TVM)	24	10	13	-
batting (TVM)	35	-	-	-
walk (mocap)	102	102	73	64
rotate arm (mocap)	156	258	180	138

図 32 モーションキャプチャデータと TVM シーケンスの間の距離
Dissimilarity of motion capture data and TVM sequences.

ようにちょうどバッターがバットを振り下ろしたシーンで、比較的立位に近い姿勢を取っている。さらに、振り下ろしたバットにスケルトン全体が引っ張られるような形になり、スケルトンが前かがみ気味の姿勢になっていることがわかる。一方 walk3 として使用したモーションキャプチャデータも、やや前かがみになりながら歩いているシーンであった。そのため、図 33 のフレームと、良くマッチしてしまったものと考えられる。また、walk4 は walk3 と同じモーションキャプチャシーケンスの異なる時間のデータを用いているため、walk1, walk2 と比較すると batter との類似度が高くなっていることが確認できた。この問題は、batter の特定の 1 フレームが図 34 のように不自然に引き伸ばされる形でマッチングされて起きたもので、このようなことが起きないようにするためには、DP マッチングの際にマッチさせるフレーム数に上限も設けるなどして対応する必要があると考えられる。

図 33 walk3 の多数のフレームとマッチした battering のフレーム (#29) と動き推定結果

A frame of battering (#29) which matches many frames of walk3.

図 34 batter と walk3 との DP マッチングの結果
Result of DP matching between batter and walk3.

6. む す び

本論文では、多数の同期カメラによって取得されフレーム間の対応関係が未知である TVM データから、被写体の大まかな動きの情報を推定した。動き推定を行うにあたって、スケルトンを用いたボトムアップな手法を提案し、従来多く用いられてきたモデルベースの手法で問題となっていた初期位置の問題を解決した。さらにユーザが選択した参照フレームを変形するというアプローチを取ることで、モデルベースの手法では対応できないような物を持ったシーンの動きを推定することに成功した。

スケルトン抽出には Reeb graph を用いた。Reeb graph 生成の際に行うモデルの分割において、測地線距離を用いることで様々な姿勢の 3 次元モデルから安定的にスケルトンを抽出することができた。また、演算時間の点においても、同様の分割結果が得られる Integral geodesic distance と比較して約 1/100 程度にまで抑えた。

さらに、動き推定の結果を応用して TVM の類似動作検索を行った。提案した動き推定手法によって TVM からモー

ションキャプチャと親和性の高いデータを獲得し、モーションキャプチャデータのために開発された類似動作検索の手法をTVMに適用した。実験では、モーションキャプチャデータをクエリとして類似した動作のTVMシーケンスの検索を行い、モーションキャプチャとTVMの融合の可能性を示した。これによって、これまでに蓄積されている大量のモーションキャプチャデータが利用可能となり、TVMの類似動作検索における先行研究¹⁶⁾では検索のクエリ自体もTVMでなければならなかったのに対して、クエリ作成のコストを大幅に削減することができた。また、今回適用したDPマッチング以外の様々なモーションキャプチャデータのための類似動作検索技術がTVM適用可能になった点も、大きな利点である。

今後の課題として、参照スケルトン選択の自動化が挙げられる。被写体の構造をきれいに反映したスケルトンが得られるフレームは、手足を広げ、胴体から離れた姿勢であることが多い。このような姿勢はメッシュモデルにおける表面の融合部分が少なくなるため、メッシュモデルの表面積を指標にするなどして、参照スケルトン選択の自動化を図れるか検討する必要がある。さらに、本論文で獲得した動き情報を、文献40)で行われているようなTVMデータの編集に応用していくことも可能であると考えている。

本研究の一部は、文部科学省「知的資産の電子的な保存・活用を支援するソフトウェア基盤技術の構築」プロジェクトの支援により行われた。実験に使用したTVMデータはNHK放送技術研究所から提供を受けている。

【文 献】

- 1) K. G. Cheung, S. Baker, and T. Kanade: "Shape-from-silhouette across time part I: Theory and algorithms," *International Journal of Computer Vision*, **62**, pp. 221-247 (2005)
- 2) 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," *映情会誌*, **58**, 6, pp.797-806 (2004)
- 3) T. Matsuyama, X. Wu, T. Takai, and T. Wada: "Real-time dynamic 3-d object shape reconstruction and high-fidelity texture mapping for 3-d video," *In IEEE Trans. Circuit And System For Video Technology*, **14**, pp.357-369 (2004)
- 4) 富山仁博, 片山美和, 折原豊, 岩館祐一: "局所的形状特徴に拘束された3次元形状復元手法とそのリアルタイム動画表示," *映情会誌*, **61**, 4, pp.471-481 (2007)
- 5) R. Skerjanc and J. Liu: "A three-camera approach for calculating disparity and synthesizing intermediate pictures," *Signal Processing: Image Communication*, **4**, 1, pp. 55-64, Nov. (1991)
- 6) S.E. Chen and L. Williams: "View interpolation for image synthesis," *In Proc. Siggraph 93*, pp.279-288 (1993)
- 7) S.E. Chen: "Quicktime vr - an image-based approach to virtual environment navigation," *In Proc. Siggraph 95*, pp.29-38 (1995)
- 8) N.L. Chang and A. Zakhor: "Arbitrary view generation for three-dimensional scenes from uncalibrated video cameras," *In Proc. IEEE International Conf. Acoustics, Speech, and Signal Processing*, pp.2455-2458 (1995)
- 9) S.M. Seitz and C.R. Dyer: "Physically valid view synthesis by image interpolation," *In Proc. Workshop on Dynamic Representation of Visual Scenes*, pp.18-25 (1995)
- 10) L. McMillan and G. Bishop: "Plenoptic modeling: an image-based rendering system," *In Proc. Siggraph 95*, pp.35-46 (1995)
- 11) M. Levoy and P. Hanrahan: "Light field rendering," *In Proc. Siggraph 96*, pp.31-42 (1996)
- 12) S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen: "The lumigraph," *In Proc. Siggraph 96*, pp.43-54 (1996)
- 13) M. Tanimoto and T. Fujii: "Ftv - free viewpoint television,"

- ISO/IEC JTC1/SC29/WG11 M8595 (July 2002)
- 14) 富山仁博, 岩館祐一: "動的3次元映像を使用したサッカーのシミュレーションシステム," 第2回デジタルコンテンツシンポジウム, 3-3 (2006)
- 15) T. Yamasaki and K. Aizawa: "Similar motion retrieval of dynamic 3d mesh based on modified shape distribution," *In Proc. Eurographics2006*, pp.9-12 (2006)
- 16) T. Yamasaki and K. Aizawa: "Motion segmentation and retrieval for 3d video based on modified shape distribution," *EURASIP Journal of Applied Signal Processing, Special Issue on Search and Retrieval of 3D Content and Associated Knowledge Extraction and Propagation*, 2007, (Jan 2007)
- 17) R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin: "Shape distributions," *ACM Transactions on Graphics*, **21**, 4, pp.807-832, (oct. 2002)
- 18) Yasuhiko Sakamoto, Shigeru Kuriyama, and Toyohisa Kaneko: "Motion map: image-based retrieval and segmentation of motion data," *In SCA '04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp.259-266 (2004)
- 19) C. Y. Chiu, S. P. Chao, M. Y. Wu, S. N. Yang, and H. C. Lin: "Content-based retrieval for human motion data," *Journal of Visual Communication and Image Representation*, **15**, pp.446-466 (2004)
- 20) Lucas Kovar and Michael Gleicher: "Automated extraction and parameterization of motions in large data sets," *In SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pp.559-568 (2004)
- 21) M. Müller, T. Röder, and M. Clausen: "Efficient content-based retrieval of motion capture data," *In SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pp.677-685 (2005)
- 22) K.M. Cheung, T. Kanade, J. Bouguet, and M. Holler: "A real time system for robust 3d voxel reconstruction of human motions," *In CVPR'00*, **2**, pp.714-720 (2000)
- 23) K. G. Cheung, S. Baker, and T. Kanade: "Shape-from-silhouette across time part II: Applications to human modeling and markerless motion tracking," *International Journal of Computer Vision*, **63**, pp. 225-245 (2005)
- 24) 宮本新, 鷲見和彦, 松山隆司: "骨格線モデルと円筒多関節モデルを併用した人体の姿勢推定," 情処学研究報告 (CVIM), 第2006巻, pp.137-144 (2006)
- 25) Ivana Mikić, Mohan Trivedi, Edward Hunter, and Pamela Cosman: "Human body model acquisition and tracking using voxel data," *Int. J. Comput. Vision*, **53**, pp.199-223 (2003)
- 26) M. Pierobon, M. Marcon, A. Sarti, and S. Tubaro: "3-d body posture tracking for human action template matching," *In ICASSP 2006*, **2**, pp.II-501-II-504 (2006)
- 27) 田中秀典, 中澤篤志, 町田貴史, 竹村治雄: "ボリウムデータの細線化とグラフマッチングを用いた事例ベース人体姿勢推定," 画像の認識・理解シンポジウム (MIRU2006), pp.70-77 (2006)
- 28) T. Mukasa, S. Nobuhara, A. Maki, and T. Matsuyama: "Finding articulated body in time-series volume data," *In AMDO 2006*, pp.395-404 (2006)
- 29) G. Reeb: "On the singular points of a completely integrable pfaff form or of a numerical function," *Comptes Rendus Acad. Sciences Paris*, **222**, pp.847-849, (1946)
- 30) M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii: "Topology matching for fully automatic similarity estimation of 3d shapes," *In Proceedings of ACM SIGGRAPH 2001*, pp.203-212 (2001)
- 31) K. Tomiyama, Y. Orihara, M. Katayama, and Y. Iwadata: "Algorithm for dynamic 3d object generation from multi-viewpoint images," *In SPIE*, **5599**, pp.153-161 (2004)
- 32) T. Matsuyama, X. Wu, T. Takai, and S. Nobuhara: "Real-time 3d shape reconstruction, dynamic 3d mesh deformation, and high fidelity visualization for 3d video," *International Journal on Computer Vision and Image Understanding*, **96**, 3, pp.393-434 (2004)
- 33) N. Gagvani and D. Silver: "Parameter-controlled volume thinning," *In CVGIP: Graphical Models and Image Processing* (1999)
- 34) T. Tung and F. Schmitt: "Augmented reeb graphs for content-based retrieval of 3d mesh models," *In the Shape Modeling International 2004 (SMI'04)*, pp. 157-166 (2004)
- 35) Y. Xiao, P. Siebert, and N. Werghi: "A discrete reeb graph approach for the segmentation of human body scans," *In Proceedings of Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM '03)*, pp.378-385 (2003)
- 36) Y. Xiao, N. Werghi, and P. Siebert: "A topological approach for segmenting human body shape," *Proceedings of the 12th International Conference on Image Analysis and Processing (ICIAP '03)*, pp.82-87 (2003)

- 37) Y. Xiao, P. Siebert, and N. Werghi : “Topological segmentation of discrete human body shapes in various postures based on geodesic distance,” Proceedings of the 17th International Conference on Pattern Recognition (ICPR ’04), pp.131–135 (2004)
- 38) P. Kanonchayos, T. Nishita, S. Yoshihisa, and T.L. Kunii : “Topological morphing using reeb graphs,” In Proceedings of the First International Symposium on Cyber Worlds (CW02), pp.465–471 (2002)
- 39) Y. Guo, G. Xu, and S. Tsuji : “Tracking human body motion based on a stick figure model,” Journal of Visual Communication and Image Representation, **5**, 1, pp.1–9 (1994)
- 40) 徐建鋒, 山崎俊彦, 相澤清晴 : “Time-Varying Mesh の編集: 動きの分解と再合成 (Motion Parsing and Composition for Editing Time-Varying Mesh),” 映像学誌, **62**, 2, pp.216–221 (2008)

ただの りゅういち
唯野 隆一 2005 年, 東京大学工学部電子情報工学科卒業. 2007 年, 東京大学大学院修士課程修了. 現在, ソニー株式会社. 正会員.

やまさき としひこ
山崎 俊彦 1999 年, 東京大学工学部電子工学科卒業. 2004 年, 東京大学大学院博士課程修了. 現在, 東京大学大学院情報理工学系電子情報学専攻講師. 主として 3 次元ビデオの圧縮と検索に関する研究に従事. 正会員.

あいざわ きよはる
相澤 清晴 1983 年, 東京大学工学部電子工学科卒業. 1988 年, 東京大学大学院博士課程修了. 現在, 東京大学大学院情報理工学系研究科電子情報学専攻教授. 画像情報処理, マルチメディア応用等の研究に従事. 正会員.