

3次元ビデオの連結による無限再生可能な自由視点映像の生成

服部 雄一^{t1} 中澤 篤志^{t1,t2} 竹村 治雄^{t1,t2}

3次元ビデオによる自由視点映像は、複数カメラで撮影した映像をもとに3次元復元を行って生成されるもので、新たな形のメディアとして盛んに研究が行われている。本研究では、インタラクティブな3次元映像の実現のための技術として、異なる3次元ビデオクリップを連結する手法を提案する。入力データである時系列3次元メッシュ形状をスケルトン化することで、各クリップ間の連結可能なフレームの探索、2つのメッシュ形状間の対応関係の決定、頂点の補間軌跡の決定を行うことができる。この手法は、人体のような多関節モデルを対象としており、人体のループ動作生成、ランダム動作生成、任意タイミングでの自然な動作切り替えなどを実現する。これにより、閲覧者が没入空間内で映像に「触る」ことで被写体に変化が起こるなど、双方向性をもつメディアの制作を支援することができる。

Using the Sequencing of 3D-Video to Create a Movie with an Arbitrary Viewpoint that can be Played Indefinitely

YUICHI HATTORI,^{t1} ATSUSHI NAKAZAWA^{t1,t2}
and HARUO TAKEMURA^{t1,t2}

3D video with an arbitrary viewpoint, generated by 3D reconstruction of images captured by multiple cameras, is currently actively researched as a new type of media. In our research, we propose a method to connect different 3D video clips in an attempt to realize interactive 3D video. Conversion of time sequence 3D mesh shapes into skeleton data enables us to find frames that can connect each clip, determine a correspondence between the two mesh shapes, and generate an interpolated path of vertices. The target of our method is a multiple joint model such as a human body, so we can create a human's loop, and naturally switch to a random action at any time. So we can apply this to the creation of interactive media, for example, an object changes when the watcher "touches" it in the virtual immersive space.

1. はじめに

3次元ビデオは、複数カメラで撮影した映像をもとに3次元形状の復元を行うことで生成されるもので、任意視点から被写体を観察できる新たな形のメディアとして盛んに研究が行われている¹⁾²⁾。被写体が人物である場合、現実世界における人体の形状や動きを保存することで、運動動作解析をはじめ、人間工学、医療、教育、また仮想空間における人物表現など、様々な分野での応用が期待される。

3次元ビデオは、HMDやCAVEなどの特殊なディスプレイを用いて提示することで、実世界の物体を没入空間において実物大で再現することができ、閲覧者は映像の場面中に実際に入り込むような感覚で閲覧す

ることができる。これを発展させると、閲覧するだけではなく、閲覧者がビデオ映像に対して直接的な操作を行うことで、被写体に何らかの変化が起こるような「さわれるビデオ」の実現も考えられるが、そのためには被写体の動きを任意のタイミングで変化させる必要がある。これを3次元幾何データに対して実現するには、3次元ビデオクリップをいくつか用意しておき、それらをスムーズに連結する技術が必要になる。例えば人間の動きを撮影した3次元ビデオの短いクリップを複数用意しておけば、閲覧者の動きや直接的操作に応じて別のクリップに遷移、あるいはクリップの冒頭に戻るなどすることで、様々な動作を繰り返したり、任意のタイミングで動作を切り替えたりすることができる。これにより、従来は見るだけであったビデオにインタラクティブ性を付加することができる。

本稿では、3次元ビデオの自動的な編集手法として、異なる3次元ビデオクリップを自動的に連結して、連結的に再生する手法を提案する。時系列3次元データの連結アルゴリズムにおいては、3次元データを解

^{t1} 大阪大学大学院 情報科学研究科
Graduate School of Information Science and Technology, Osaka University
^{t2} 大阪大学 サイバーメディアセンター
Cybermedia Center, Osaka University

析して得たスケルトンデータを利用する。スケルトンデータをもとにして、連結可能フレームの探索、2つのメッシュ形状間の対応関係の決定、および頂点の補間軌跡の生成を行う。この手法は人体を始めとした多関節モデルを対象にしており、人体のループ動作の生成、ランダム動作の生成、および任意タイミングでの動作切り替えを実現することができる。

2. 関連研究

本章では、3次元ビデオ、モーション合成、メッシュモーフィングについて、本研究に特に関わりの深い関連研究を紹介する。

3次元ビデオは、複数カメラの撮影画像から生成される3次元オブジェクトを時系列ビデオとして表現したものである。Laurentiniら³⁾は、撮影画像の背景差分からシルエット画像を生成し、Visual hullを再構築し、シーンを構成する多面体を推定することで3次元オブジェクトの復元を実現した。この技術をもとに、ポリゴンモデルや色の付いたボクセルモデルを復元する研究が行われている。松山ら⁴⁾は、復元した3次元ポリゴンに画像ベースのテクスチャをマッピングし、オフライン処理において写実的なモデルを実現している。我々の研究に関連の深い動作補間の先行研究としては、Vedulaら⁵⁾の、視点・時間ともに異なる2つのボクセル構成シーンからシーンフローを推定して時空間補間を行うものがある。

本研究では、3次元復元したモデルをスケルトン化して被写体の動きを補間する。動きの補間に関して、モーションキャプチャデータの合成や連結に関する研究が過去に数多く行われている。Witkinら⁶⁾はモーションキャプチャデータを自然に連結するMotion Warpingを提案した。Pullenら⁷⁾は、exampleモーションから得た統計をもとに新たなモーションを合成する手法を提案した。Perlinら⁸⁾は、ルールベースのシステムを用いて、生成したモーションを一貫性のあるストリームに内挿している。Kovarら⁹⁾が提案したMotion Graphsは、複数のモーションキャプチャデータを合成して新たなモーションキャプチャデータを生成するもので、我々のとるアプローチに近い。

また、モーションの遷移部分に関する研究としては、Perlenら⁸⁾がこの分野に早くから取り組んでおり、2つのクリップを繋ぐためのスムーズな補間手法を提案している。Leeら¹⁰⁾は、この連結にさらに整合性を持たせるために方向フィルタを定義しており、また、Roseら¹¹⁾は、運動学的制約と基本的な動的設定を保持したモーション遷移手法を提案している。

メッシュ形状の補間は、3次元メッシュモーフィングに関連する問題になるが、この分野の研究も非常に盛んに行われており¹²⁾、主な議論主題は対応関係の取り方と頂点軌跡である。対応点については、各頂点をト

ポロジ保持のための多角形や球体へマッピングして対応関係を得る方法¹³⁾もあるが、より一般的なのは形状を部分分解してから各形状に対してマッピングを行う方法¹⁴⁾である。頂点軌跡に関しては、Alexaら¹⁵⁾が二次元ポリゴンを対象とした歪みの少ない補間手法を提案し、道川ら¹⁶⁾が多重解像度補間メッシュを用いてこれを三次元拡張している。また、Blandingら¹⁷⁾は、物体のスケルトン構造を利用して頂点軌跡を求めたアプローチを取っている。

メッシュの色補間に関しては、2次元画像のモーフィングの研究が参考になる。こちらもソースとターゲットの対応関係をどう取るかが最大の問題となり、多くの先行研究が存在する。画像をメッシュ分割して補間を行うMesh Warping¹⁸⁾や、2つの画像に曲線を引いて対応をとるField Morphing¹⁹⁾など数多くの手法が提案されており、ワーピングや色変化のコストを最小化する問題などについても議論されている²⁰⁾。

3. 時系列3次元データ連結アルゴリズム

提案するアルゴリズムは、2つの異なる3次元ビデオクリップ（時系列3次元メッシュデータ）を自動的かつスムーズに連結するものである。

特定の2フレームの3次元形状を繋ぐためには補間形状を生成しなければならないが、そのためには各形状の頂点の対応関係の情報が必要になる。ここでは人体を始めとする多関節モデル同士の補間を行うが、実際の人間の形状は腕が曲がりたり体が反ったり、さまざまな変形が起こり、正確な部分対応を取るのが困難である。また、頂点の対応関係が正確に判明しても、それらの補間軌跡をどのように求めるかという問題が残る。単純な線形補間を行うと、人間の動きとして整合性が取れなくなる可能性がある。

これらの問題を解決するために、本研究では各形状データを解析して得られるスケルトンデータを取得する。スケルトン化することで被写体が連結元と連結先において、どの部位がどのように動いたか知ることができ、対応を取る際の手助けとなる。また、スケルトン構造の補間軌跡を参考にすることで、破綻の起きにくい3次元形状の頂点補間を行うことができる。

提案する時系列3次元データ連結アルゴリズムの概略を図1に示す。手順は以下の通りである。

- (1) 連結可能フレームの探索
- (2) スケルトンベースでの補間
- (3) 補間形状データの生成
- (4) 連結クリップの生成

以下、順を追って説明する。

3.1 連結可能フレームの探索

異なる時系列3次元データ同士を連結する際、形状が大きく異なるフレーム同士では良い連結結果は望めない。まずは、どのフレームとどのフレームを繋ぐの

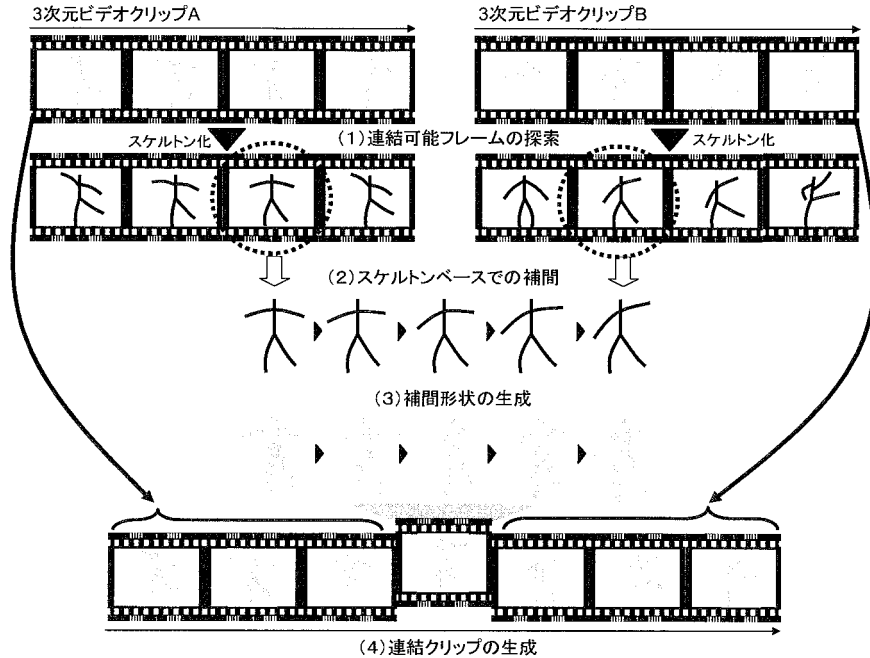


図 1 時系列 3 次元データ連結アルゴリズムの概略

が最適であるかを調べる必要がある。

ここでは、齋藤ら²¹⁾の手法を用いて各 3 次元データに対応するボリュームデータに対して細線化を行い、時系列スケルトンデータを得る。スケルトンデータはボクセルの集合によって表現されており、これを解析することで、スケルトンがいくつの部位から成るかを知らることができる。連結可能であるかどうかの最低条件は、この部位数が一致していることである。

続いて、候補として挙げられた連結元クリップのスケルトンと連結先クリップのスケルトンをマッチングし、連結可能なフレームを探索する。

2つのスケルトン S_1, S_2 のをマッチングを考える。最初に、2つのスケルトンの位置と向きを揃えるために、重心と最適回転量を求める。重心は全構成点の平均座標を求めればよい。各スケルトンを構成する点の座標を $s_1(i), s_2(j)$ とおく (i, j は各構成点の ID) と、最適回転行列 R_{best} は以下の式より求められる。

$$R_{best} = \arg \min_R \sum_i dist(s'_1(i), R s'_2(i)) \quad (1)$$

$$s'_1(i) = s_1(i) - g(S_1) \quad (2)$$

$$s'_2(i) = s_2(near(i)) - g(S_2) \quad (3)$$

ここで、 $dist(a, b)$ は 2 点 a, b のユークリッド距離を、 $g(S_1), g(S_2)$ は S_1, S_2 の重心座標を、 $near(i)$ は回転後の状態で ID : i の構成点に最も近い位置にある構成点の ID を指す。

なお、一般には R_{best} は全ての回転を考慮するのではなく、被写体の性質によっては回転軸を制限することもある。具体的には、人体の場合は鉛直方向の軸を中心とした回転のみを考え、一定幅ずつ回転させながら式 (1) の $\arg \min$ 以下の値を最小とする回転角を探していく。

求めた最適回転量を用いて、2つのスケルトンの類似度の評価値 $sim(S_1, S_2)$ を求める。

$$sim(S_1, S_2) = \sum_i dist(s'_1(i), R_{best} s'_2(i)) \quad (4)$$

この値が小さいほど、スケルトン同士の類似度が高い。

クリップ同士のマッチングは、各連結フレーム候補同士をマッチングしていくが、単独フレームのみをマッチングすると、連結した際に被写体の動きが突然変化する場合があります。そこで、動きの文脈的意味を重視し、図 2 に示すように対象となるフレームを中心として前後数フレームをマッチングし、類似度評価値の総和で連結フレームを選ぶ。クリップ A, B の連結可能フレーム $\{f_A, f_B\}$ は以下の式で求める。

$$\{f_A, f_B\} = \arg \min_{\{f_1, f_2\}} \sum_{t=-N}^N sim(A_{f_1+t}, B_{f_2+t}) \quad (5)$$

A_n, B_n は各クリップの第 n フレームのスケルトンデータを、 N は前後フレームのマッチング範囲を表す。

3.2 スケルトンベースでの補間

連結すべき 2 フレームが選ばれたら、2つのスケルトンデータの補間軌跡を求める。

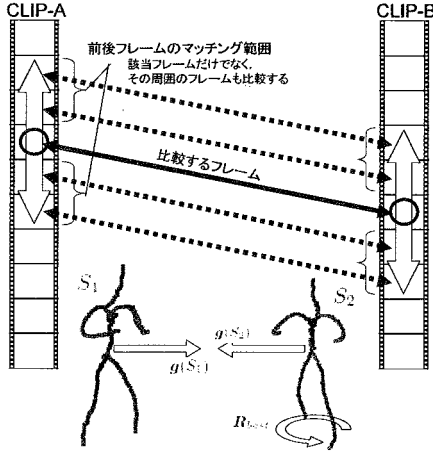


図2 連結可能フレームの探索

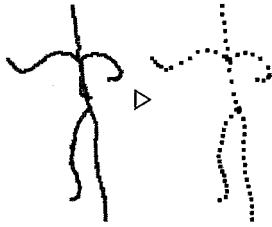


図3 スケルトンデータの間引き

2つのスケルトンは、前節のマッチングによって最適回転量が求められているので、向きを揃えることができる。また、スケルトンを構成している点（以下、スケルトン構成点と呼ぶ）の集合で1本の曲線として表現できる部分を「パーツ」とすると、パーツ同士の対応もスケルトン構成点の位置から求めることができる。ところが、スケルトンデータはボクセルデータを細線化して得られるので、スケルトン構成点の数はフレームによって違っている。そこで、図3に示すように、補間を行いやすくするためにスケルトン構成点の間引き、数を揃える。連結元となるスケルトンの各パーツにおいて、パーツの構成点数を n 、サンプリングレートを R （定数）とすると、そのパーツの間引き後の構成点数 n' は、

$$n' = \left\lceil \frac{n}{R} \right\rceil + 1 \quad (6)$$

と定義し、両スケルトンにおいてこの数に揃える。なお、間引き後もパーツの始端点と終端点は必ず残したいので、1を加えている。

続いて、各スケルトン構成点の補間を行う。ここで重要なのは、本研究で対象とする被写体は多関節モデルであり、それらの動きが補間過程で破綻してはならないことである。最も単純な補間は、スケルトン構成

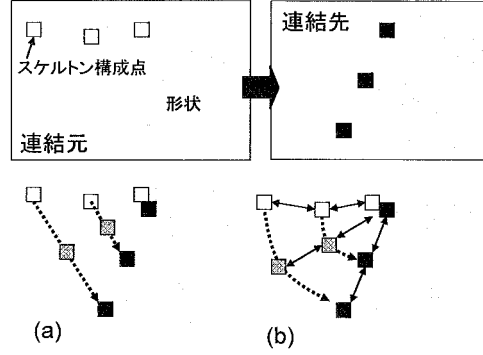


図4 スケルトンデータの補間

(a) は線形補間なのでスケルトン構成点間の距離が変化する場合がある。(b) は構成点間の距離を保持しており、こちらが行うべき補間である。

点単位での線形補間だが、この場合、図4(a)のように、補間過程で腕の長さが縮むなどの非現実的な変形が起こる可能性がある。そこで、ここでは図4(b)のように、隣接するスケルトン構成点との距離を保ちつつ補間を行う必要がある。

間引き後の2つのスケルトンデータ S_1 、 S_2 の補間データ $S_{intp(t)}$ を求める (t は経過時間の割合)。構成点 ID の順番は、スケルトンの重心により近い座標にある側を始端、もう一方を終端とする。ここでは1つのパーツに着目し、スケルトン S_1 のそのパーツの第 p 構成点を $S_1(p)$ 、その座標を $s_1(p)$ と表す。補間スケルトン $S_{intp(t)}$ の各構成点 $s_{intp(t)}(p)$ は以下の式で求める。

$$s_{intp(t)}(p) = s_{intp(t)}(p-1) + \Delta s_{intp(t)}(p) \quad (7)$$

$$\Delta s_{intp(t)}(p) = (1-t)\Delta s_1(p) + t\Delta s_2(p) \quad (8)$$

$$\Delta s_1(p) = s_1(p) - s_1(p-1) \quad (9)$$

$$\Delta s_2(p) = s_2(p) - s_2(p-1) \quad (10)$$

各補間点は、隣(1つ前)の補間点からのベクトル $\Delta s_{intp(t)}(p)$ によって計算する。このベクトルは、スケルトン S_1 、 S_2 それぞれの対応するベクトルの線形補間により得られる。点ではなくベクトル(変化量)を補間することによって、スケルトン構成点間の距離を維持した形状で補間を行うことができる。

3.3 補間形状データの生成

補間メッシュ形状の頂点軌跡は、前節で求めたスケルトンの補間軌跡を追従する形で求められる。

まず、メッシュ形状の各頂点がどのスケルトン構成点を追従するかを決定する。頂点 V (座標: \mathbf{v}) が選ぶ追従スケルトン構成点 $S_{(V)}$ (座標: $\mathbf{s}_{(V)}$) は以下のように求める。

$$S_{(V)} = \arg \min (dist(\mathbf{v}, \mathbf{s})) \quad (11)$$

ただし以下の条件を満たすこと。

$$\cos^{-1}(\mathbf{n}_V \cdot (\mathbf{v} - \mathbf{s}_{(V)})) < \pi \quad (12)$$

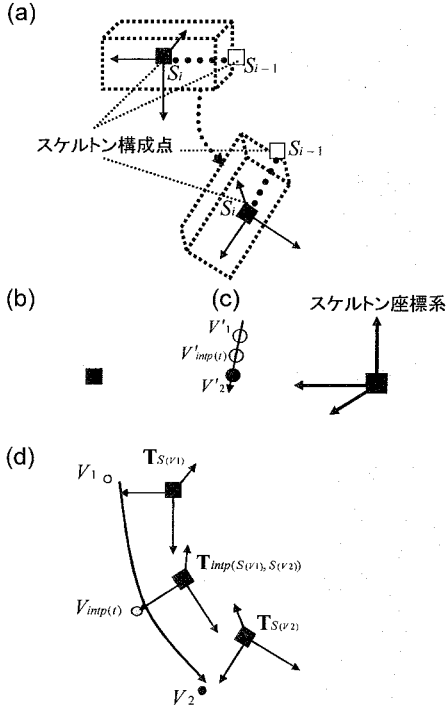


図5 メッシュ形状の補間

(a) 頂点のグループにおける座標系の設定 (□ から ■ へ向かうベクトルを X 軸とする), (b) 回転を考慮しない場合のマッチングの不具合, (c) スケルトン座標系内での頂点補間, (d) 補間頂点を世界座標系に戻す

ただし、 n_V は頂点 V の法線ベクトルである (頂点が含まれる全ての面の法線ベクトルの平均をとる)。

追従点が決定することで、メッシュ頂点群はスケルトン構成点の総数と同数のグループに分割される。図 5(a) に示すように、このグループ内の頂点同士で対応を見つけ、補間軌跡を生成するのが基本的な考えである。ただし、図 5(b) のように形状の向きが回転すると正しい対応関係が求められないので、被写体の形状をできるだけ保持するために、ソース形状とターゲット形状を適切に重ね合わせて対応関係を見つける必要がある。そのために、頂点 V に対して以下の座標変換を行う。

$$v' = T_{S(V)} v \quad (13)$$

あるパーツの第 p スケルトン構成点 $S(p)$ が作る座標系の変換行列 $T_{S(p)}$ は、以下の式で表す。

$$T_{S(p)} = \begin{bmatrix} r_x & r_y & r_z & s(p) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

座標軸ベクトル r_x, r_y, r_z は以下の 2 つのベクトルを正規直交化して得られる。

$$r_{x0} = s(p) - s(p-1) \quad (15)$$

$$r_{y0} = \begin{cases} s_{intp(1)}(p) - s(p) & (S_1, \text{連結元}) \\ s(p) - s_{intp(f)}(p) & (S_2, \text{連結先}) \end{cases} \quad (16)$$

なお、 f は補間フレームの数を表す。この座標系は、隣接するスケルトンの方向ベクトルと、スケルトンの補間軌跡の動く方向ベクトルを基準として設定している。

スケルトン座標系内でユークリッド距離の最も近い対応点を探した結果、頂点 V_1 と V_2 の対応が判明したとする (図 5(c))。その頂点に対応する補間頂点 $V_{intp(t)}$ (t は経過時間の割合) の座標 $v_{intp(t)}$ を求めたい。まずは、スケルトン座標系内におけるローカルな補間頂点座標 $v'_{intp(t)}$ を求める。

$$v'_{intp(t)} = (1-t)v'_1 + tv'_2 \quad (17)$$

この座標を元の座標系に戻さなくてはならないが、ここでは「座標系の補間」を行い、補間した変換行列の逆変換を使用する (図 5(d))。

$$v_{intp(t)} = T_{intp(S(V_1), S(V_2))}^{-1} v'_{intp(t)} \quad (18)$$

補間座標変換行列 $T_{intp(S(V_1), S(V_2))}$ は、世界座標系から補間スケルトン座標系への変換を表す。導出法は基本的に式 (14) と同様だが、式 (15)(16) を以下のものに置き換えて求める。

$$r_{x0} = s_{intp(t)}(p) - s_{intp(t)}(p-1) \quad (19)$$

$$r_{y0} = s_{intp(t+1)}(p) - s_{intp(t)}(p) \quad (20)$$

最後に、各メッシュデータの持つ色情報の補間に関して述べる。3次元メッシュデータは頂点ごとに色情報を持っており、各頂点ごとに正しい対応が取れていれば、2次元画像のモーフィングという所のワーピング元・ワーピング先の座標情報は既に得られている。そこで、対応する頂点の持つ RGB 値の線形補間によって色の補間を行うことができる。

3.4 連結クリップの生成

(1) クリップ A の初め～連結フレーム, (2) 前節で生成した補間クリップ, (3) クリップ B の連結フレーム～最後、を順に繋ぐと連結クリップを生成することができる。しかし、クリップ A と B の連結フレームにおいては、一般には形状の位置・向きとも異なるので、これらを揃える必要がある。

位置に関しては、クリップ A・B の全フレームのデータを、それらの各連結フレームにおけるスケルトンの重心座標が原点になるよう平行移動させる。向きについては、連結可能フレーム探索の際に最適回転量 R_{best} を求めているので、クリップ B の全フレームのデータをその分回転させる。位置と向きを揃えれば、オリジナルフレームと補間フレームとスムーズに繋がる。

なお、回転量を求めることにより、被写体が回転していてもメッシュを補間することができるが、被写体によっては回転軸を制限する必要がある。例えば人体のスケルトンが対象の場合は、鉛直方向の軸を中心とした回転以外は考慮しない。それ以外の回転も考慮し



図 6 データ撮影のための実時間キャプチャスタジオ

てしまうと、傾いた姿勢のスケルトンと直立のスケルトンがマッチする可能性がある。こうなると、クリップ A, B どちらかに姿勢を揃えなければならないため、いずれかのフレームの姿勢が常に傾くなど、好ましくない結果が出る場合がある。

4. 新規 3 次元ビデオの生成

時系列 3 次元データ連結アルゴリズムを使用して、無限再生可能な 3 次元ビデオを生成する。これは、複数のクリップを次々に連結していき、無限に再生できる映像を表現したものである。本項では、この過程と結果について述べる。

入力データとなる時系列 3 次元データは、図 6 の実時間キャプチャスタジオで撮影・取得した。このスタジオでは、天井に固定された 8 台のカメラが被写体を周囲から見下ろす環境を設定している。各カメラにおいては、平面パターンを用いてレンズの非線形歪みを校正し、既知の大きさのキャリブレーションボックスを用いてカメラパラメータを求めておく²²⁾²³⁾。入力画像から色相を考慮した背景差分を用いることで対象の領域を抜き出し、カメラパラメータと視体積交差法²⁴⁾を用いて被写体のボクセルデータを復元する。さらに、このデータをマーチン・キューブ法により表面形状のみを持つ 3 次元メッシュデータに変換する。

また、カメラパラメータを用いることで、メッシュデータの各頂点座標に対応する撮影画像上の座標が取得できるので、メッシュ頂点に画素の色情報載せることでカラーメッシュデータが得られる。ただし、撮影画像はカメラの台数分存在するので、各頂点の法線ベクトルと、頂点からカメラ位置に向かう方向ベクトルを比較し、その角度差が最も小さいカメラを最適カメラとして選択し、そのカメラで撮影された画像の該当画素の色を使用した。

今回は、ソースクリップとして 50 フレームの 3 次元ビデオクリップを 3 種類作成した (図 7(a))。これらを相互に連結するためには補間クリップが 9 種類必要になる (自分自身のフレーム冒頭に繋がるケースも考慮)。各クリップにおいて連結フレームを探索した。各クリップにおいて、パーツ数が一致し、かつ式 (5) で計算される評価値が 2000 以下のものを連結可能で

表 1 別クリップへ遷移可能なフレームの割合

連結元	連結先	遷移可能フレーム数	(割合)
A	B	10 / 50	(20%)
A	C	18 / 50	(36%)
B	A	6 / 50	(12%)
B	C	7 / 50	(14%)
C	A	18 / 50	(36%)
C	B	14 / 50	(28%)
平均		12.2 / 50	(24.3%)

あるとし、他のクリップに遷移可能であるフレームの数を表 1 に示す。対象クリップによってやや差が見られるが、平均 24.3% のフレームで別のフレームへの遷移が可能であり、ほぼ 4 フレームに 1 フレームの割合で遷移が可能になることが分かる。任意タイミングでのクリップ切り替えを想定した場合、ユーザの指定したタイミングから大きな遅延が起らず遷移が可能であると考えられる。

これらの補間クリップを使用して、ランダムな順番で 3 種類の動作を繰り返す無限再生映像を生成した。連結可能フレームの探索結果の例として、図 7(b) にクリップ A とクリップ B の主なスケルトンデータと、最も評価値の小さな類似フレームを示す。なるべく動きの小さい、補間の行きやすいフレームが選ばれていることがわかる。また、補間部分の結果の例として、図 7(c) にクリップ A からクリップ B へ向かう補間クリップのスケルトンデータ、メッシュデータ、カラーメッシュデータ、およびメッシュデータの右腕部分のみの補間の拡大図を示す。スケルトンの補間は問題なく行っているが、メッシュデータや色の補間についてはやや崩れが見られ、改善の余地がある。

この無限再生ビデオを VR 環境に応用すれば、閲覧者がスタイラスやサイバークロップなどの空間入力装置で被写体に「触れる」と被写体の動きが変化するなど、双方向性を持つアプリケーションを制作することができる。

5. おわりに

本稿では、インタラクティブな 3 次元ビデオの実現のための提案として、複数の 3 次元ビデオクリップをソースとした無限再生可能な自由視点映像を紹介した。ここで使用したのは、2 つの 3 次元ビデオクリップを自動的かつスムーズに連結するアルゴリズムであり、アルゴリズムにおける重要な特徴は、3 次元形状を細線化してスケルトンデータを使用することである。スケルトンデータを用いることで、連結可能フレームの探索、2 つの 3 次元データの正確な対応関係の決定、自然な頂点の補間軌跡の推定を行うことができた。

しかし、補間クリップの細かい部分に注目すると、形状や色の補間にはやや崩れている部分も見られた。

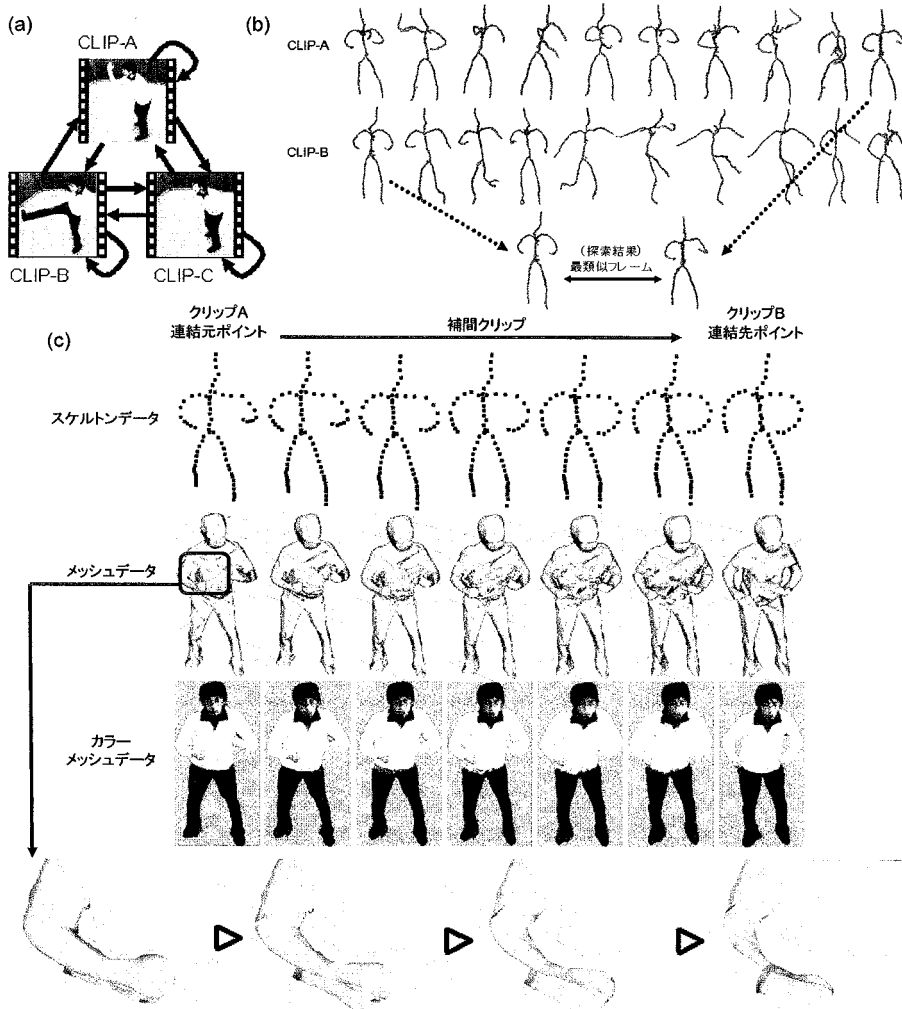


図7 新規3次元ビデオの生成

(a) 取得した3種類のソースクリップ, (b) スケルトンデータの比較によるクリップAとクリップBにおける連結可能フレームの検索, (c) クリップAからクリップBへの連結の際の補間結果

これを解決するために、メッシュ形状やテクスチャがさらに安定するような補間手法や、より適切な連結フレームを見つけるための探索手法を考える必要がある。

参考文献

- 1) Rashid, R.: Towards a System for the Interpretation of Moving Light Display, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.2, No.1, pp.574-581 (1980).
- 2) Cheung, K. M., Kanade, T., Bouguet, J.-Y. and Holler, M.: A real time system for or-

bust 3D voxel reconstruction of human motions, *Proceedings of the 2000 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, Vol.2, pp.714-720 (2000).

- 3) Laurentini, A.: The Visual Hull Concept for Silhouette-Based Image Understanding, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 16, No.2, pp.150-162 (1994).
- 4) Matsuyama, T. and Takai, T.: Generation, visualization, and editing of 3D video, *3D Data Processing Visualization and Transmis-*

- sion, 2002. *Proceedings. First International Symposium on*, pp.234-245 (2002).
- 5) Vedula, S., Baker, S. and Kanade, T.: Spatio-temporal view interpolation, *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*, pp.65-76 (2002).
 - 6) Witkin, A. and Popovic, Z.: Motion warping, *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp.105-108 (1995).
 - 7) Pullen, K. and Bregler, C.: Animating by Multi-Level Sampling, *CA '00: Proceedings of the Computer Animation*, p.36 (2000).
 - 8) Perlin, K.: Real Time Responsive Animation with Personality, *IEEE Transactions on Visualization and Computer Graphics*, Vol.1, No.1, pp.5-15 (1995).
 - 9) Kovar, L., Gleicher, M. and Pighin, F.: Motion graphs, *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pp.473-482 (2002).
 - 10) Lee, J.: A Hierarchical Approach to Motion Analysis and Synthesis for Articulated Figures, *Journal of Parallel and Distributed Computing* (2000).
 - 11) Rose, C., Guenter, B., Bodenheimer, B. and Cohen, M. F.: Efficient generation of motion transitions using spacetime constraints, *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pp.147-154 (1996).
 - 12) Alexa, M.: Recent advances in mesh morphing, *Computer Graphics forum*, Vol.21, 2, pp.173-197 (2002).
 - 13) Alexa, M.: Merging polyhedral shapes with scattered features, *The Visual Computer*, Vol.16, No.1, pp.26-37 (2000).
 - 14) Kanai, T., Suzuki, H. and Kimura, F.: Metamorphosis of Arbitrary Triangular Meshes, *IEEE Comput. Graph. Appl.*, Vol.20, No.2, pp.62-75 (2000).
 - 15) Alexa, M., Cohen-Or, D. and Levin, D.: As-rigid-as-possible shape interpolation, *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp.157-164 (2000).
 - 16) 道川隆士, 金井 崇, 鈴木宏正: 多重解像度補間メッシュにおける近似剛体補間, *Visual Computing / グラフィクスと CAD 合同シンポジウム* (2004).
 - 17) Blanding, R. L., Turkiyyah, G. M., Storti, D. W. and Ganter, M. A.: Skeleton-based three-dimensional geometric morphing, *Comput. Geom. Theory Appl.*, Vol.15, No.1-3, pp.129-148 (2000).
 - 18) Smithe, D.B.: A Two-pass Mesh Warping Algorithm for Object Transformation and Image Interpolation, Technical memo #1030, Industrial Light and Magic (1990).
 - 19) Beier, T. and Neely, S.: Feature-based image metamorphosis, *Seminal graphics: pioneering efforts that shaped the field*, pp.373-380 (1998).
 - 20) Sederberg, T.W. and Greenwood, E.: A physically based approach to 2-D shape blending, *Computer Graphics (SIGGRAPH '92 Proceedings)* (Catmull, E. E., ed.), Vol. 26, No. 2, pp. 25-34 (1992).
 - 21) 齋藤豊文, 森 健策, 鳥脇純一郎: ユークリッド距離変換を用いた 3 次元デジタル画像の薄面化および細線化の逐次型アルゴリズムとその諸性質, *電子情報通信学会論文誌. D-II, 情報・システム, II-情報処理*, Vol.79, No.10, pp.1675-1685 (1996).
 - 22) 加藤博一, M.Billinghurst, 浅野浩一, 橘啓八郎: マーカー追跡に基づく拡張現実感システムとそのキャリブレーション, *日本バーチャルリアリティ学会論文誌*, Vol.4, No.4, pp.607-616 (1999).
 - 23) 井口征士, 佐藤宏介: 三次元画像計測 (1990).
 - 24) Wada, T., Wu, X., Tokai, S. and Matsuyama, T.: Homography Based Parallel Volume Intersection: Toward Real-Time Volume Reconstruction using Active Cameras, *CAMP '00: Proceedings of the Fifth IEEE International Workshop on Computer Architectures for Machine Perception*, p.331 (2000).