

3次元ビデオにおける人物動作の編集

小林亮介^{†1} 高井勇志^{†1} 松山隆司^{†1}

3次元ビデオの人物動作を自然さを保ったまま編集する方法を提案する。本論文では「自然さ」を、鑑賞者がその動作が現実起こりうるかと判断できるものと定義し、3次元ビデオの世界座標における位置の変化から感じられる運動学的特徴を制約条件として、その位置や再生速度の最適化を行った。本手法を用いることによって、個別に撮影された2つの3次元ビデオを、同時に再生する時に生じる違和感を減少させることが確認できた。

Motion Editing for 3D Video

RYOSUKE KOBAYASHI,^{†1} TAKESHI TAKAI^{†1} and TAKASHI MATSUYAMA^{†1}

3D videos are not only imaginary and artificial three dimensional animations made with CG but also three dimensional scene reconstruction from real video that capture the shape and appearance of the performer in full wardrobe. This paper proposes a method of editing human motion while preserving *the naturalness* of action in 3D videos, which is defined as the way how viewers perceive the motion which can be observed in the physical world. To accomplish this editing, we optimise positions and playback speed of 3D videos under constraint of the natural dynamics. Our proposed method enables us to reduce a mismatch which occurs when separately captured 3D videos are located in the same world coordinate system.

1. はじめに

3次元ビデオ^{*1}はCGによる仮想的、人工的な3次元アニメーションではなく、ダンスや

スポーツをする人間などの時間的変化を3次的にそのまま記録した実写立体映像であり、これまでは無形文化財のデジタルアーカイブ化のように編集を伴わない用途に用いられていたが、今後は映画のPreViz^{*2}のように本格的な編集を伴う応用が期待される。

そこで本研究では3次元ビデオの編集として、撮影された3次元ビデオの人物動作の自然さを保った範囲での編集を扱うこととする。ここでいう自然さとは、鑑賞者から見て、その動作が現実起こりうるかと判断できるというものである。したがって自然さを保った編集というのは、3次元ビデオのメッシュ形状と、位置の変化から鑑賞者が感じられる運動学的特徴と矛盾なく人物の運動を編集するということである。図1に例を示す。この編集が可能

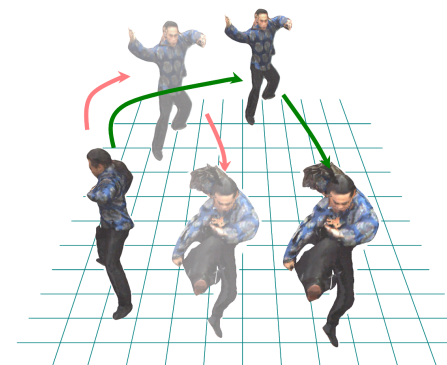


図1 人物動作の自然さを保った編集

Fig. 1 Editing human motion with preserving the nature of action.

になれば、人物の歩行経路を変更したり、別々に撮影された3次元ビデオのタイミングを調節したりすることが可能になると考えられる。3次元ビデオは実写立体映像であることから、鑑賞者が3次元ビデオが運動学的制約に基づいているかどうか、自然であると判断する指標であると考えることができる。そういった運動学的制約に基づかずに編集を行えば、

^{†1} 京都大学大学院情報学研究所

Graduate School of Informatics, Kyoto University

^{*1} 3次元ビデオは、各時刻毎に多視点画像から復元された一連の3次元形状及びテクスチャの集合である。各時刻において復元された3次元形状及びテクスチャをフレームと呼ぶ。また、3次元形状は頂点とその連結関係を記

録したメッシュ構造として表現されている。

^{*2} PreViz(Pre-Visualization)とは、映画の本番撮影前に想定シーンをあらかじめ可視化する技術のことで、従来の絵コンテでは表現できない激しいアクションや複雑なカメラワークなど、監督の持つ画作りのイメージを制作スタッフが共有することができ、本番撮影がよりスムーズなものとなる、というものである。

鑑賞者には大きな違和感を与えてしまう。編集前は、その3次元ビデオの特徴から、運動学的制約には完全に基づいているので、編集は元の運動学的特徴をどのように保存するかが問題となる。

そこで、本研究では3次元ビデオのメッシュに対応付けられた骨格構造を使用し、骨格構造の加速度、速度、位置そして再生速度から上記の制約をモデル化し、これらの制約の下での編集を行う手法を考案した。

1.1 関連研究と本研究の位置づけ

3次元ビデオに関連する研究として、3次元ビデオの撮影範囲の拡張を目指した研究、カメラのキャリブレーションに係る研究、多視点画像から精度のよい3次元形状を復元する研究⁵⁾などが進められている。また3次元ビデオで様々な単純な動作を撮影しておきライブラリを作成することで、インタラクティブに3次元ビデオの人物動作を変更することができるという研究も行われている³⁾。また一方で、復元された3次元ビデオから人体の動作を推定し、骨格構造を得る研究も進められている⁴⁾。本研究は対応付けられた骨格構造(スケルトンと呼ぶ^{4)*1})を編集することで、それに対応付けられたメッシュを動かし、3次元ビデオの動きを編集しようというものである。これはCGにおけるモーショキャプチャを用いたキャラクターアニメーションと似た発想であるが、大きく異なるものである。

まず、CGにおけるキャラクターアニメーションは、一般にあるキャラクターの1つのメッシュとそれに対応付けられたフレームごとのスケルトンの動きによって行われる。つまりキャラクターのメッシュは全フレーム同じものを使用している。一方、3次元ビデオはフレームごとに別々のメッシュを復元している。3次元ビデオではメッシュとスケルトンの対応付けを毎フレーム行う必要が出てくるということである。つまり動作にそった自然な洋服の皺、髪の毛の動きなど、微細な視覚情報を復元することが可能な一方、スケルトンによって動作を編集しようとすると、それに適した洋服の皺や髪の毛の動きになるようにメッシュも変形するわけではないので、CGに比べ違和感が発生しやすくなってしまふ。

*1 腰の位置をルート(根)ノードとして持っており、ルートノードは子ノードとして腹ノードと左大腿ノードと右大腿ノードを持っている。このノード間をボーンと言う。また腹ノードは子ノードとして胸ノードを、胸ノードは首ノード、左肩ノード、右肩ノードを子ノードとして持っている。以下同様に人間と似た骨格構造を階層構造として持っている。子ノードには、親ノードまでの距離が記述されており、ルートノードにはワールド座標系での位置が記述されている。親ノードが動くとき以降の子ノードが運動する。つまりルートノードが移動、回転をすると、体全体が動き、腹のノードに回転情報を与えるとそこから下の階層である上半身が回転する。スケルトンは、以上の各ノードの階層構造を持ったヒエラルキー部と、各時刻ごとの各ノードの親から見たときの初期位置からの回転情報が記述されているモーション部と再生速度とをデータとして持っている。

また、一般にCGにおいて、キーフレーム間をつなぐ手法としては、スプライン補間が用いられる。これはキーフレームにおける速度、加速度を計算することで、その間をなめらかにつなぐというものである¹⁾。これはセグメント内の人物の動き、ルートノードの位置を考慮しているものではない。したがって、今回キーフレームのルートノードにスプライン補間を適用しても、元のセグメントの経路は一般に保存されないため、元の3次元ビデオの運動学的特性を保存するという本研究の目的にはそぐわない。

以上の2点が本論文でのCGにおけるスケルトンを用いた編集と3次元ビデオにおけるスケルトンを用いた編集の差異である。

2. 提案手法

3次元ビデオの運動学的特性に基づいた編集を行うことを目標としている。

2.1 問題設定

提案手法の具体的な応用として、本論文では以下の問題に着目して研究を行う。現在、複数人が出演している3次元ビデオを作ろうとしたとき、スタジオの大きさの制限、復元精度の観点から、一度に多人数を同じスタジオで撮影することが出来ないため、別々に撮影する必要が出てくる。3次元ビデオにおける2者のアクションシーンはプロの役者によって撮影されているが、各々目の前にいない役者を想定して撮影されたものである。それを鑑賞する時はその2つの3次元ビデオを1つのワールド座標系に設置している(図2)。このとき、各3次元ビデオのローカル座標に移動ベクトル、回転ベクトルを与え、あるフレームで2者の位置関係、タイミングが合うように調整されている。しかし、このような設定であると、別のフレームでは両者の位置関係がおかしかったり、殴る、殴られるといった2者間で特に働きかけをしているフレームでタイミングがずれてしまっているものが見られる(図3)。これは鑑賞者に大きな違和感をもたらす。この違和感は2つの3次元ビデオを一つのワールド座標系に配置したときに起こる相対的なものであり、各々を別々に鑑賞したときには感じるものでないため、本研究では3次元ビデオの運動学的特性に基づくことで自然な範囲で2者の3次元ビデオの動作を編集し、それを消すことを問題として扱う。

また、2者のアクションシーン、特にここでは格闘シーンを扱うが、これを想定する場合、お互いが常に向き合っている姿勢であることが考えられるので、ここでは回転に関しては扱わない。つまり、最初に2つの3次元ビデオを一つのワールド座標系に配置したとき以降、回転に関しては制御しない。

本研究では、目標を達成するに当たって、推定されたスケルトン自体の編集はできるだ

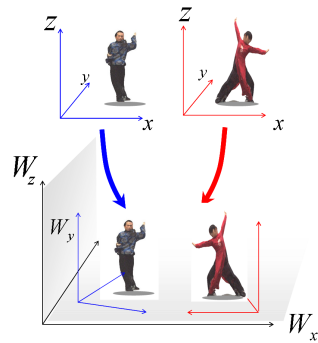


図 2 複数の 3 次元ビデオの配置
 Fig. 2 Locating multiple 3D videos.

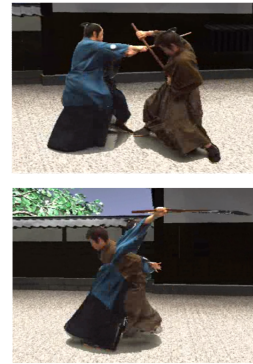


図 3 複数の 3 次元ビデオの同時再生
 Fig. 3 Playing multiple 3D videos.

け行わないという方針をとる．したがって人物の動作としては 3 次元ビデオの人物の位置とその再生速度を扱う．なぜなら，3 次元ビデオは 1.1 節で述べたとおり，毎フレーム別のメッシュを使用している．そのため，あるフレームから検出されたスケルトンの足や手を動かし，それをまたメッシュに埋め込み対応付けするという作業（スキニング）をしたとき，洋服が重なっている部分など，元のメッシュデータにはない場所での色や形状情報が必要になってくる可能性があるからである．そういった理由で，検出されたスケルトンにはできる限り手を加えないものとし，元の視覚情報を保存しつつ，運動学的制約を保つためにスケルトンにおける編集は，そのルートノードに限定する．

2.2 語句の定義とアルゴリズムの概要

2.2.1 語句の定義

2.2.1.1 キーフレーム

位置，タイミングともに 2 つの 3 次元ビデオ間で合っていて欲しいフレームを指す．このフレームにおける 2 つの 3 次元ビデオの相対的な位置，再生時刻が演者の意図と異なると，鑑賞者は違和感を感じると考えられるフレームである．具体的には一方がもう一方になんらかの働きかけをしているフレームとそれに対応するもう一方の 3 次元ビデオのフレームがキーフレームとなる．3 次元ビデオにおける運動学的特徴の特異点として，編集の軸として考える．また，上記とは別に，各 3 次元ビデオ映像のシーケンス全体（すべてのフレームの集合）の最初のフレームもキーフレームとする． k 番目のキーフレームをキーフレーム k と

定義する．

2.2.1.2 セグメント

キーフレーム自身を含むキーフレーム間のフレームの集合である．すなわちキーフレーム k は $k-1$ 番目のセグメントとは k 番目のセグメントのどちらにも含まれる（ただし，最後のセグメントは除く）． k 番目と $k+1$ 番目のキーフレーム間のセグメントをセグメント k と定義する．

2.2.2 アルゴリズムの概要

アルゴリズムの概要は以下の通りである．3 次元ビデオにおける人物の運動学的特性を考慮しスケルトンのルートノードの位置と再生速度を変更する．

- (1) 3 次元ビデオからのスケルトンの推定
- (2) キーフレームの選択
- (3) キーフレームの時刻調整
- (4) キーフレームの位置調整
- (5) 各セグメントでの経路の最適化
- (6) 調整したスケルトンを各フレームに対応する 3 次元ビデオメッシュにスキニング 1 に関しては文献 4) の方法を用い，6 のスキニングについても従来の方法で行うので，本論文では 2 から 5 について主に論じていくこととする．

2.3 キーフレームの調整

2.3.1 キーフレームの選択

キーフレームの定義（2.2.1.1 節）にしたがって各シーケンスを確認し，手作業で選択する．3 次元ビデオ A でキーフレームが n 個選択された場合，もう一つの 3 次元ビデオ B でもそれに対応したキーフレームを n 個選択する．

2.3.2 キーフレームの時刻調整

キーフレームは時刻，位置，ともに合っていなければいけないフレームである．まず，その再生時刻に関して合わせることを考える．運動学的特徴の保持するという観点から，元の人物の加速度と重力加速度をなるべく維持する手法で時間調節する必要がある．3 次元ビデオにおけるキーフレームを除いたセグメント k におけるフレーム数を $S_{[k]}$ ，再生速度を $x_{[k]}$ とすれば，セグメントの再生時間は

$$t_{[k]} = \frac{S_{[k]}}{x_{[k]}} \quad [\text{s}] \quad (1)$$

と表すことができる．セグメント内で地面に足がついている場合，3 次元ビデオ A,B の再

表 1 キーフレーム時刻調整
Table 1 Editing playback time of keyframe

t_k	A	\bar{A}
B	$\min_{t_k} \left\{ \frac{ t_{A[k]} - t_k }{t_{A[k]}} + \frac{ t_{B[k]} - t_k }{t_{B[k]}} \right\}$	$t_k = t_{B[k]}$
\bar{B}	$t_k = t_{A[k]}$	$\min_{t_k} \left\{ \frac{ t_{A[k]} - t_k }{t_{A[k]}} + \frac{ t_{B[k]} - t_k }{t_{B[k]}} \right\}$

A: $skeletonA_k[i].r_{toe} > height$ かつ $skeletonA_k[i].l_{toe} > height$ となるフレームが3次元ビデオ A のセグメント k の中にある． B: $skeletonB_k[h].r_{toe} > height$ かつ $skeletonB_k[h].l_{toe} > height$ となるフレームが3次元ビデオ B のセグメント k の中にある． $skeletonA_k[i].r_{toe}$ はセグメント k の i 番目のフレームの3次元ビデオ A から検出されたスケルトンの右足の位置の世界座標系における z 座標を示す． $height$ はそれ以上の高さで空中に浮いていると考えられる地面からの距離を表す．

A: a case when segment k of 3D video A has frames which include $skeletonA_k[i].r_{toe} > height$ and $skeletonA_k[i].l_{toe} > height$. B: a case when segment k of 3D video B has frames which include $skeletonB_k[i].r_{toe} > height$ and $skeletonB_k[i].l_{toe} > height$. $skeletonA_k[i].r_{toe}$ denotes the position of right foot of skeleton on the world coordinate which is estimated from 3D video A in frame number i of segment k . $height$ denotes the minimum distance between a 3D video object and the floor ($z = 0$) that viewers can perceive the object is in the air.

生速度はなるべく維持しつつ、キーフレームの時刻を合わせるため、

$$\min_{t_k} \left\{ \frac{|t_{A[k]} - t_k|}{t_{A[k]}} + \frac{|t_{B[k]} - t_k|}{t_{B[k]}} \right\} \quad (2)$$

より t_k を求める．一方、ある一定以上の高さにスケルトンの足が位置するフレームがある場合、そのセグメントは重力加速度の影響を強く鑑賞者に与えていると考えられるので、該当する3次元ビデオのセグメントに関しては再生速度を変更しない．もう片方の3次元ビデオの再生速度でキーフレームの時刻合わせをする．動作の変曲点、キーフレーム以外の場所で再生速度が変わることで、元の演者の運動学的特性が失われると考えられるため、セグメント内では一定の再生速度とする．例えば、3次元ビデオ A のセグメント k で空中にいると判定されるフレームがあれば、

$$t_k = t_{A[k]} \quad (3)$$

t_k を求める．それをキーフレーム k が再生されてからキーフレーム $k+1$ が再生されるまでの時間とする．以上のように求めた t_k から再生速度 $\dot{x}_{A[k]}, \dot{x}_{B[k]}$ を計算する．以上をまとめたものを表 1 に示す．

2.3.3 キーフレームの位置調整

キーフレームは、最初のキーフレーム、つまり3次元ビデオがワールド座標系のどこを基準にして再生するかを決めるフレームと、それ以外の演者の位置、タイミングがあって欲しいフレームとに分類される．各3次元ビデオ A, B のキーフレームのルートノ

ドをそれぞれ、 $A_1, A_2, \dots, A_{n-1}, A_n, B_1, B_2, \dots, B_{n-1}, B_n$ とすると、最初のキーフレームは A_1, B_1 である．方針として、 $\vec{A_k A_{k+1}}, \vec{B_k B_{k+1}}$ ベクトルを保つことを考える．もしこのベクトルの保存を考慮に入れず、キーフレームの位置を設定すると、人物が前に歩いてる姿勢であるのに横に移動するという現象が見られると考えられる．鑑賞者は3次元ビデオの運動学的特性をメッシュの形状から無意識に頭の中で推定しているはずで、それからずれたワールド座標の位置の変化は、違和感として大きく感じられる． A_1, B_1 は他の接触を想定したキーフレームと違い、 A_1 と B_1 の位置関係を指定することは難しい．なぜなら、格闘シーンを想定したとき、それはお互いに距離をとって離れているフレームであり、動作が行われないと違和感を感じる要因にはならないからである． A_1 と B_1 の位置関係を決定するためにも、 A_2, B_2 の位置関係について考え、その後すべてのキーフレームの位置を設定するアルゴリズムについて以下に記す．

(1) 演者が意図すると想定される互いの位置関係になるように、 A_2 に与える平行移動ベクトル l_{A_2} を手作業で求める (図 4 左)．

(2) この平行移動ベクトルをすべての3次元ビデオ A のフレームのルートノードに与える．キーフレームに着目すると、

$$A_k = A_k + l_{A_2} \quad (k = 1, 2, \dots, n-1, n) \quad (4)$$

である．最終的な各キーフレームのルートノードの位置を $\bar{A}_k, \bar{B}_k (k = 1, 2, \dots, n-1, n)$ とすると、

$$\bar{A}_1 = A_1, \bar{A}_2 = A_2, \bar{B}_1 = B_1, \bar{B}_2 = B_2 \quad (5)$$

となる．

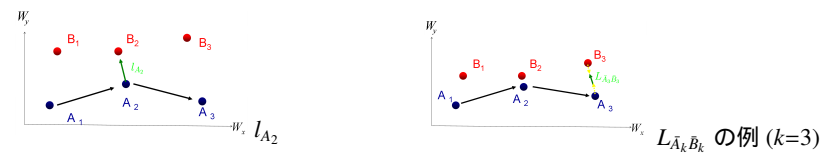


図 4 キーフレームの位置調整

青丸、赤丸はそれぞれスケルトンのルートノードの位置を表す

Fig. 4 Editing the position of keyframe.

Red and blue points denote the position of root node.

(3) セグメント 2 以降の、キーフレームの位置について考える．各 $k (k = 2, 3, \dots, n-1, n)$

にそれぞれについて理想的な 2 者の位置関係を示すベクトル $L_{\vec{A}_k \vec{B}_k}$ を手作業で求める (図 4 右) .

$$L_{\vec{A}_k \vec{B}_k} = \overrightarrow{\vec{A}_k \vec{B}_k} \quad (6)$$

(4) $L_{\vec{A}_k \vec{B}_k}$ を条件として,

$$\min_{\vec{A}_k, \vec{B}_k} f(\vec{A}_k, \vec{B}_k) = \left| \overrightarrow{A_k A_{k+1}} - \overrightarrow{\vec{A}_k \vec{A}_{k+1}} \right|^2 + \left| \overrightarrow{B_k B_{k+1}} - \overrightarrow{\vec{B}_k \vec{B}_{k+1}} \right|^2 \quad (7)$$

より, $\vec{A}_k \vec{B}_k (k = 2, 3, \dots, n-2, n-1)$ を順次求める .

2.4 セグメントの経路最適化

キーフレームの位置がそれぞれの 3 次元ビデオにおいて求めたので, セグメントにおける両者の経路の最適化を行う . CG で行われているキーフレーム間の補間方法はセグメント内の経路を保存したものではなく, もとの 3 次元ビデオの運動学的特性を考慮したものではない . そこでルートノードの位置に関してこれを考慮した目的関数を設定し, それを最小化することで, セグメントの経路を最適化する方法を考案した .

2.4.1 目的関数の設定

あるセグメントにおける各フレームのルートノードの位置を $C_i (i = 1, 2, \dots, m-2, m-1)$ とし, 経路の最適化後の各フレームのルートノードの位置を $\vec{C}_i (i = 1, 2, \dots, m-2, m-1)$ とする . またこのセグメントでのキーフレームのルートノードの位置は $C_0, C_m, \vec{C}_0, \vec{C}_m$ であり, \vec{C}_0, \vec{C}_m はキーフレームの位置調整 (2.3.3 節より $\vec{C}_0, \vec{C}_m = \vec{A}_k, \vec{B}_k (k = 1, 2, \dots, n-1, n)$) によって求まっている . 目的関数として次を設定する .

$$f = \sum_{i=1}^{m-1} \{ \gamma f_a(\vec{C}_i) + (1-\gamma) w_i f_v(\vec{C}_i) \} \quad (8)$$

鑑賞者から見た運動学的特徴の保存は加速度を保つことで満たされると考え, 目的関数中の $f_a(\vec{C}_i)$ を以下のように定義する .

$$f_a(\vec{C}_i) = |(C_{i-1} - 2C_i + C_{i+1}) - (\vec{C}_{i-1} - 2\vec{C}_i + \vec{C}_{i+1})|^2 \quad (9)$$

加速度の保存だけを関数に設定すると, 運動学的な制約を保持する最適な解が求まるが, あらたな歪みが出てくると考えられる . 加速度が 0 であるということは, 静止しているか, 等速直線運動をしているということであるが, この関数だけでは, ルートノードの速さが一定なのか 0 なのかの判別が出来ないため, あるフレームで元は静止しているのに, 最適化後, 加速度を維持しようという関数の作用で動いてしまうということがありうる . それを防ぐために, 目的関数中の $f_v(\vec{C}_i)$ は以下のように定義する .

$$f_v(\vec{C}_i) = \left| \frac{C_{i-1} - C_{i+1}}{2} - \frac{\vec{C}_{i-1} - \vec{C}_{i+1}}{2} \right|^2 \quad (10)$$

$$w_i = \frac{1}{1 + \exp\{\alpha(v_i - v_k)\}} \quad (11)$$

ここで

$$v_i = \frac{|C_{i-1} - C_i| + |C_i - C_{i+1}|}{2} \quad (12)$$

と定める . この項によって, 最小化する際, 速度が小さければ小さいほど, 最適化前と後で速度の変化を小さくするよう力が働く .

2.4.2 初期位置

無制約最適化問題として,

$$\min_{C_i (i=1,2,\dots,m-2,m-1)} f = \sum_{i=1}^{m-1} \{ \gamma f_a(\vec{C}_i) + (1-\gamma) w_i f_v(\vec{C}_i) \} \quad (13)$$

を解くが, これを解く過程で, $\vec{C}_i (i = 1, 2, \dots, m-2, m-1)$ に与えられる値によって求められるのは局所解のいずれかであって, それが最適解である保証はない . したがって, 最適の解に近い値を初期位置に与える必要がある . 一番すべてのフレームに関して最適解に近い初期位置を与えていると考えられる式 (14) を用いる .

$$\vec{C}_i = C_i + \left(\frac{m-i}{m} \right) \overrightarrow{C_0 \vec{C}_0} + \left(\frac{i}{m} \right) \overrightarrow{C_m \vec{C}_m} \quad (14)$$

2.5 評価方法

以上のアルゴリズムより時間, 位置ともに最適化されたルートノードが, 修正前と比べてどの程度演者の意図にそったものかを評価する方法について説明する . 格闘シーンを 3 次元ビデオで撮影した後, 同じ動きを今度は 2 者同時にモーションキャプチャの技術を使い撮影した . つまり, モーションキャプチャを用いて得られたデータは, 演者の意図にそった 3 次元ビデオにとっての理想的な編集結果と位置付けることができる . モーションキャプチャもそれぞれ, モーションキャプチャ A とモーションキャプチャ B とする . 片方の 3 次元ビデオ A を基準とすると, 各フレームの再生時刻が求まり, スケルトン A に対しても対応するフレームを算出することができる . また, スケルトン A の各フレームをモーションキャプチャ A の各フレームを対応させることで, 同時刻に再生されるフレームを算出することができる . こうすることで編集後のスケルトン A とスケルトン B の相対的な位置関係が, モーションキャプチャ A とモーションキャプチャ B の関係に近いと比較することができる . 正解

のモーションキャプチャの各時刻でのルートノードをそれぞれ $M_A(t)$, $M_B(t)$ としてセグメント k における 3 次元ビデオのキーフレームを除いたフレーム数を m , 修正後の 3 次元ビデオの再生速度を $\dot{x}_{A[k]}$ とし, 3 次元ビデオ B を基準とすると,

$$\dot{t} = t\dot{x}_{A[k]} \quad (15)$$

とすることで

$$f_{\text{before}}(k) = \frac{1}{m} \sum_{t=1}^m \{|M_A(t) - M_B(t)\} - \{C_A(t) - C_B(t)\} \quad (16)$$

$$f_{\text{after}}(k) = \frac{1}{m} \sum_{t=1}^m \{|M_A(\dot{t}) - M_B(\dot{t})\} - \{\bar{C}_A(\dot{t}) - \bar{C}_B(\dot{t})\} \quad (17)$$

として定義する. この 2 つの式を比較することによって, どれだけ全体として意図された位置関係に同時刻にいるかを比較することができる. 関数の値が小さければ小さいほど, そのセグメントにおいて毎時刻正解に近い位置関係であることが言える.

3. 実験結果

3.1 実験条件

実験データとして別々に撮影された男女のカンフーの格闘シーンをを用いる. どちらも再生時間 15 秒ほどの長さのものである. 3 次元ビデオ A を男性, 3 次元ビデオ B を女性とする. また, 本来はこの 2 つの 3 次元ビデオから推定されたスケルトンを使用すべきであるが, 現在はその精度に問題があるため, ここでは 3 次元ビデオとはべつに撮影されたモーションキャプチャのデータを 3 次元ビデオ A から推定されたスケルトンとし, スケルトン A とする. 3 次元ビデオ B でも同様にモーションキャプチャのデータを使うべきではあるが, 互いのモーションキャプチャのデータは同時に撮影されたものであるため, 位置, タイミングとも完全に一致してしまっている. よって使用することが出来ない. 本研究では 3 次元ビデオから推定されたスケルトンにおいて, 現段階ではルートノードの位置を制御することだけを行った. したがって, 3 次元ビデオ B においては時刻調整における空中にいるかどうかの判定時にはバウンディングボックス^{*1}を使用し, ルートノードのかわりにメッシュの全頂点の重心 (センターと呼ぶ.) を扱うことにする.

*1 バウンディングボックスとは 3 次元ビデオの各フレームの全頂点をすっぽり包み込む, 最小の直方体のことである. 今回はジャンプの判定に使用するだけであるので, バウンディングボックスで十分であると考えられる. また, 頂点の重心をルートノードとして扱うが, 厳密にはこれはルートノードに対応するものではない.

3.2 キーフレームの調整結果

3.3 キーフレームの選択

キーフレームの選択は, 2.3.1 節の定義にしたがって, 各 3 次元ビデオにおいて 12 個のキーフレームを選択した (図 5).



図 5 選択したキーフレーム
Fig. 5 Selected keyframes.

3.4 キーフレームの時刻編集結果

再生時刻を 2.3.2 節のアルゴリズムに基づいて決定した. 元の再生速度はスケルトン A は $60fps$ であり, 3 次元ビデオ B は $25fps$ である. 再生速度には, 整数しか与えることが出来ない. 再生速度を求めて, それを四捨五入して再生速度 $\dot{x}_{A[k]}$, $\dot{x}_{B[k]}$ として与えた. 結果を

表 2 時刻編集結果
Table 2 Result of editing the playback speed.

セグメント (k)	$\dot{x}_{A[k]}$ [fps]	$\dot{x}_{B[k]}$ [fps]	セグメント (k)	$\dot{x}_{A[k]}$ [fps]	$\dot{x}_{B[k]}$ [fps]
1	59	25	8	60	19
2	60	24	9	60	23
3	60	25	10	44	25
4	60	12	11	55	25
5	60	31	12	60	25
6	52	25	13	60	25
7	60	23			

表 2 に示す。

3.4.1 キーフレームの位置調整結果

手作業で l_{A_1} を求め、全フレームのスケルトン A のルートノードに対してそれを与えた。また、 $L_{A_k B_k}$ は、スケルトン A のルートノードと、3次元ビデオ B のセンターとの関係によって求めた (2.3.3 節)。ここではセグメント 1 において、2つの3次元ビデオはお互いに並進運動を仮定していなかったが、キーフレーム 1 の位置に違和感を感じることはなかった。並進運動に類似した動作があった場合、2.3.3 節のアルゴリズムを用いてキーフレーム 2 に合わせて 2 者のキーフレーム 1 の位置を決定しても、その位置に違和感を感じる可能性がある。これに関しては、スケルトンのルートノードの角度を調節するアルゴリズムが必要であると考えられる。

3.5 セグメントの経路最適化結果

各 3次元ビデオの各セグメントにおいて、経路の最適化を行った。ここでは、男性スケルトン A のセグメント 5 における経路最適化について考察する。セグメント 5 は、相手を肘うちして、一度後ろに下がり、静止し、飛び蹴りをするというもので、全セグメントの中でもっとも目的関数の有用性を確認できると考えられるセグメントである。

実験は $\alpha = 3.0$, $v_k = 1.0$, $\gamma = 0.5$ として行った。編集結果を図 6 に示す。ルートノードの位置は修正前はキーフレーム 5 に対してのキーフレーム 6 が y 軸マイナス方向であったのに対して、編集後は y 軸プラス方向であり、鑑賞者に大きな違和感として影響があると考えられる。しかし、加速度の変化は x 軸方向には見られず、また y 軸方向に関して、元のルートノードと編集後のルートノードでは差があるものの、凹凸の形は保存されており、編集前の加速度がプラスのところでは編集後の加速度もプラス、編集前の加速度がゼロもしくはマイナスのところでは編集後の加速度がマイナスである。これはジャンプしているフレー

ムを含め人物にかかる力の方向を保存していること、つまり鑑賞者から見た運動学的特徴を保存していると考えられるため、ルートノードの位置としては大きな変更があっても違和感を感じさせることはなかった。次に、足の滑りについて考察を進める。このグラフでは、編集前と編集後の差があるところが鑑賞者には滑っていると感じられるフレームを表す。ほとんどのフレームで足がすべっていないことが確認できるが、右足の $40 < i < 60$ の範囲では、足が滑っているのが確認できる。このフレーム群では演者は右足を軸にして左足を前方から後方に下げているのだが、後ろに左足を下げている時、右足が滑ってしまっている。ルートノードは速さを持って動いているので、セグメントにおいてこのフレーム群は w_i が小さく、速さを保持する力が強く働いてなかったこと、セグメントの最初のキーフレームと最後のキーフレームの位置関係が修正前と修正後では大きく異なっていたこと、この 2つが直接的な原因であるが、そもそもルートノードの制御のみで運動学的特徴を保存できていることに無理があると考えられる。

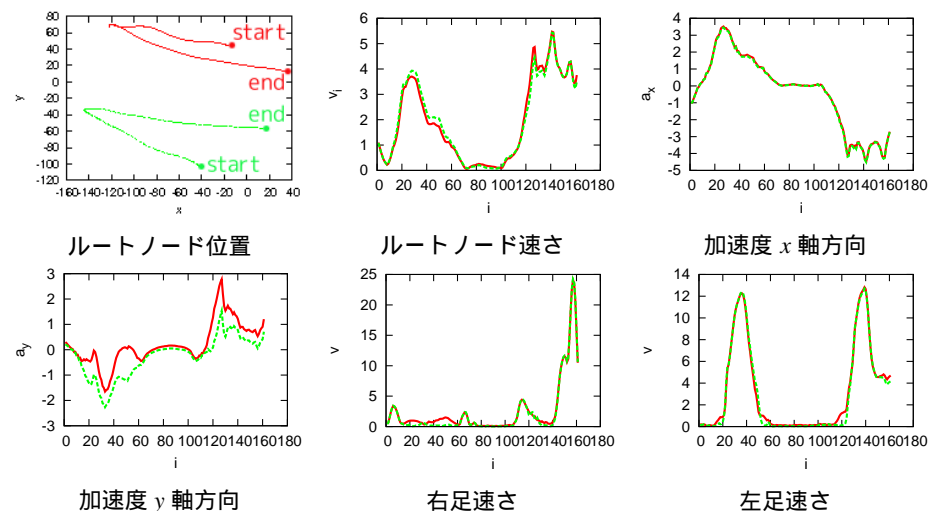


図 6 セグメント 5 における実験結果 (赤:編集前 緑:編集後)
Fig. 6 Experimental result of segment 5 (red: before green: after).

3.6 評価

$\alpha = 3.0$, $v_k = 1.0$, $\gamma = 0.5$ として行った最適化によって得られた各セグメントのルート

表 3 3次元ビデオ B を基準とした毎時刻における相対的な位置関係の差
Table 3 Difference of relative positions between video A and video B based on 3D video B.

セグメント (k)	$f_{\text{before}}(k)$ [cm]	$f_{\text{after}}(k)$ [cm]	セグメント (k)	$f_{\text{before}}(k)$ [cm]	$f_{\text{after}}(k)$ [cm]
1	79.37	38.48	8	234.9	157.6
2	165.8	89.73	9	397.3	281.4
3	277.6	160.4	10	293.3	169.8
4	345.0	188.6	11	521.6	394.6
5	401.3	289.3	12	494.8	367.8
6	182.1	104.8	13	417.6	234.5
7	223.4	119.5			

ノードを after とし、評価関数にしたがって評価した。ここでは before として、時間方向には調節を行わず、キーフレーム 2 において位置があうように、ルートノードに平行移動ベクトルを与えたものを使用した。式 (16) と式 (17) の t については 3 次元ビデオ B のフレームを基準とした。結果を表 3 に記す。ここでセグメントという表現は 3 次元ビデオ B を基準としたものである。before では 3 次元ビデオ B のキーフレームの再生時刻と、スケルトン A の再生時刻は必ずしも一致しない。

表 3 よりすべてのセグメントにおいて正解の位置関係、タイミングに編集後は近づけたと言える。つまり、元の運動学的特徴の変化を最小限にした 3 次元ビデオの編集を行えたと言える。セグメント 1 に関しては、before と after の違いはスケルトン A の再生速度を 60fps から 59fps に変更したのみであるが、大きな差が出た。これは、静止している 3 次元ビデオ B にスケルトン A が 3 次元ビデオ B にすばやく向かっていく動作があったためと考えられる。後半のセグメントになるほど、全体的に高い数値になっているが、これは、相手を想定して撮影を行う場合、互いの動作にずれが生まれるためであると考えられる。

4. 結 論

本研究では人物動作としての自然さを保った範囲での 3 次元ビデオの編集を行った。自然さは、運動学的特性を考慮することで保存することができると考え、メッシュに対応付けられた骨格構造を用いて、その特性に基づいたモデル化を考案し、それによる編集の手法を示した。

実験結果から、ほとんどの時刻において鑑賞者に違和感を感じさせない自然な 2 つの 3 次元ビデオの同時再生を可能にする人物動作の編集方法を示すことができた。その一方で、運動学的特性は、ルートノードを制御することが出来るという仮定に基づいているため、ル

ートノードの位置の変化が人物の運動学的特徴と関係が薄いときには、足が滑るなどの問題が見られた。本研究で制御をルートノードのみとしたのは多視点画像から得られた本来の形状の変形を極力避けたいと考えたからであるが、今後は 3 次元ビデオのスケルトン自身を編集すれば、それに付随する形で 3 次元ビデオのメッシュも違和感なく変形させることが出来る編集方法について考えていきたい。

謝辞 本研究は、独立行政法人科学技術振興機構 (JST) チーム型研究 CREST 「デジタルメディア作品の制作を支援する基盤技術」の支援を受けて行った。

参 考 文 献

- 1) MARK DELOURA. *game programming gems*. ボーンデジタル, 2002.
- 2) Lucas Kovar, Michael Gleicher, and John Schreiner. Footskate cleanup for motion capture editing. In *ACM SIGGRAPH Symposium on Computer Animation*, pp. 97–104, 2002.
- 3) Jonathan Starck and Adrian Hilton. Surface capture for performance-based animation. *IEEE Computer Graphics and Applications*, Vol.27, No.3, pp. 21–31, 2007.
- 4) 宮本新. 3次元モデルを用いた重度の接触を含む複雑な人体動作の推定. Master's thesis, kyoto, 2008.
- 5) 延原章平, 和田俊和, 松山隆司. 弾性メッシュモデルを用いた多視点画像からの高精度 3次元形状復元. 情報処理学会 CVIM 研究会論文誌, SIG11(CVIM5), Vol.43, pp. 53–63, 12 2002.