

3D キャラクタアニメーション作成支援のための 人物動作推定システム

水野 克哉[†], 柏崎 礼生[†], 高井 昌彰^{††}, 高井 那美[‡]

[†] 北海道大学大学院情報科学研究科

^{††} 北海道大学情報基盤センター

[‡] 北海道情報大学

あらまし 近年、個人による 3DCG アニメーションの製作が盛んになってきているが、ユーザの前に立ちほだかる壁として、キャラクタのモーション作成が難しいという問題がある。本稿では、2 次元映像を入力とし、映像に映っている人物の動作を推定、モーションデータを出力するシステムを提案する。人物の動作推定は、人物の特徴点（関節位置）をトラッキングし、その結果と動画像から得られる情報を用いて 3 次元空間における特徴点の座標値を求め、得た座標値から各関節の回転角度を計算することで行う。本システムを用いることで、3DCG アニメーション作成の難しさを軽減することができると考えられる。

キーワード 動作推定, 単眼視, マーカレスモーションキャプチャ, オプティカルフロー, トラッキング

Human Motion Estimation System for 3D Character Animation

Katsuya MIZUNO[†] Hiroki KASHIWAZAKI[†] Yoshiaki TAKAI^{††} Nami TAKAI[‡]

[†] Graduate school of Information Science and Technology, Hokkaido University

^{††} Information Initiative Center, Hokkaido University

[‡] Hokkaido Information University

Abstract Recently it becomes much more active to make 3DCG animations. But there is a problem that it is not so easy to make motion of human characters. In this paper, we propose the human motion estimation system. The input of the system is markerless 2D monocular video data, and the output is 3D motion data with a standard format. The human motion estimation process is carried out by tracking feature points (joint positions) of the human body, getting 3D coordinates of the feature points using the results of tracking information of video images, and calculating angle of rotation at each joint using 3D coordinates of the feature points. This motion estimation system makes it easier to produce 3DCG animation.

Keywords motion estimation, monocular vision, markerless motion capture, optical flow, tracking

1 はじめに

近年、youtube[1] やニコニコ動画 [2] といった映像作品公開の場の拡充や、個人が所有するコンピュータの性能向上により、個人による様々な作品制作活動が盛んになってきている。その中でも、好きな曲に対して好きな 3D キャラクタを用いた Promotion video や、好きな 3D キャラクタに色々な動作をさせる動画、すなわち 3D キャラクタアニメーションを作成し公開することが流行している。

しかし、3D キャラクタアニメーションの作成過程において、キャラクタのモーションを作成するという作業は熟練を要し、3D キャラクタアニメーション作成の初級、中級者にとっては非常に困難であるという現実がある。このような敷居の高さは作品製作に対するモチベーションの低下を起し、潜在的なクリエイターの減少に繋がると考えられる。

そこで本研究では、3D キャラクタアニメーション作成の初級、中級者に対してモーション作成を簡易化することで、3D キャラクタアニメーションの作成支援を行うことを目的とする。すなわち、ユーザが撮影した 2 次元人物映像を入力として与え、映像に映っている人物動作の推定を行い、モーションデータを出力するシステムを構築する。本システムは 3D キャラクタアニメーション作成の初級、中級者の作品製作に対するモチベーションを高め、ひいては日本のコンテンツ産業を支える若者の増加に貢献するものとする。

2 システムの概要

2.1 入力映像の定義

本システムが対象とする入力映像は、単眼カメラで撮影された映像であり、以下の条件を満たすものとする。

- 最初のフレームは背景のみ。
- 人物は一人のみ。

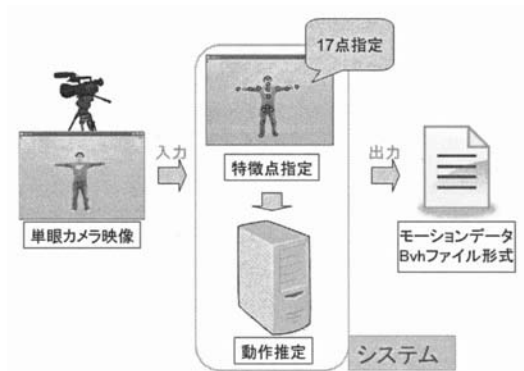


Fig.1 人物動作推定システム

- 人物は最初に初期姿勢をとる。
- 体全体が光軸方向に移動するような動作がない。

2.2 システムの流れ

本システムの流れは Fig.1 のようになる。

まずユーザは単眼カメラで撮影された映像を入力する。次にユーザは映像において初期姿勢をとっている人物に特徴点を 17 点指定する。この特徴点が人物の各関節位置となる。以上の入力をユーザが行った後、システムは人物の動作を推定し、モーションデータを出力する。

なお、出力するモーションデータのフォーマットには、モーションデータを格納するための代表的なファイルフォーマットである Bvh ファイル形式を用いる。

2.3 動作推定

動作推定の流れは以下になる。各処理は、Fig.2 のように人体領域を頭・胴体領域、腕領域、脚領域に分け、この順に行う。

1. ユーザが入力した特徴点から人物のボーンデータを作成する。人物のボーンデータは、Fig.3 のような腰ノードをルートノードとした木構造で表現されており、各ボーンの長さ、各フレームにおけるモーションデータ（各関節ノードの回転量、ルートノードの平行移動量）を保持する。
2. 映像の毎フレームについて、以下の処理を繰り返す。
 - (a) 背景差分により人物シルエットを作成し、その重心と慣性主軸を求める。
 - (b) 肌色領域を抽出しそれぞれ手、顔と対応付ける。
 - (c) $L_t - s$ 極大値を求め、それぞれ手、足先に対応付ける。
 - (d) 特徴点の 2 次元座標を推定する。
 - (e) 特徴点の 3 次元座標を推定し、モーションデータを計算する。

なお $L_t - s$ 極大値とは、人物シルエット画像の輪郭線上の点列 s について、重心との距離 g_t と頭頂部との距離

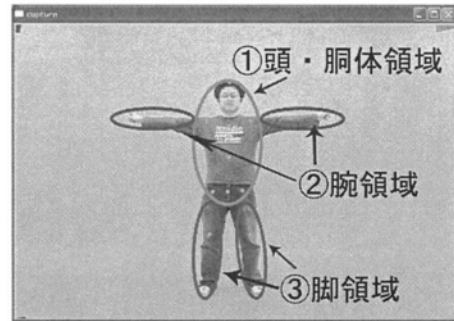


Fig.2 人体の領域分割

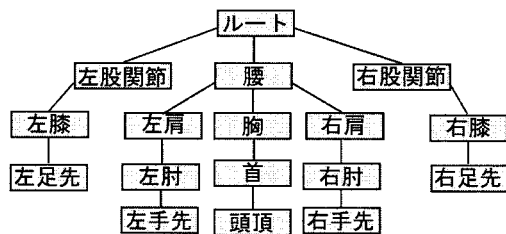


Fig.3 人物ボーンデータの階層構造

p_t による以下の計算式で求まる $L_t - s$ 曲線の極大値のことである。

$$L_t(s) = \sqrt{p_t^2 + g_t^2} \quad (1)$$

$L_t - s$ 極大値となる輪郭部分は、手、足先といった突出した部分と対応するという性質がある [3]。

2.3.1 特徴点の 2 次元座標推定

頭・胴体領域の特徴点は、Fig.4 のようにフレーム間の重心の移動量分平行移動させ、慣性主軸の回転量だけルートノードを中心に回転させる。

手、脚領域の特徴点は、手、足先特徴点、肘特徴点、膝特徴点それぞれ以下に示す方法で 2 次元座標推定を行う。手、足先特徴点は、オプティカルフロートラッキングを行って得た特徴点を、肌色領域、 $L_t - s$ 極大値を用いて Fig.5 のように修正する。肘特徴点は、まずオプティカルフロートラッキングを行って得た特徴点について、肩特徴点、および手先特徴点とでつくる線分が人物領域外を通っているかどうかを確認する。Fig.6 のように人物領域外を通っている場合は、肘特徴点のオプティカルフロートラッキングは失敗したと判断し、以下のような修正を行う。その後、肘特徴点と肩特徴点、手先特徴点との長さが、それぞれ上腕ボーンおよび下腕ボーンより長くないように修正する。

1. 肩特徴点と手先特徴点を結ぶ線分 L を、上腕ボーンの長さ：下腕ボーンの長さ：内分する点 A を求

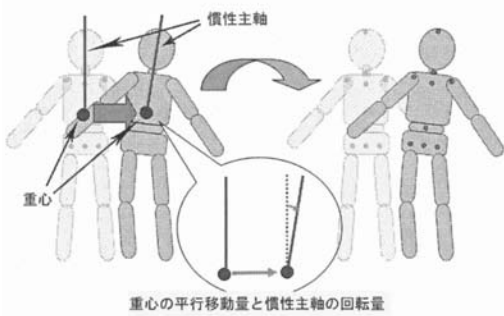


Fig.4 頭・胴体領域の特徴点の2次元座標推定

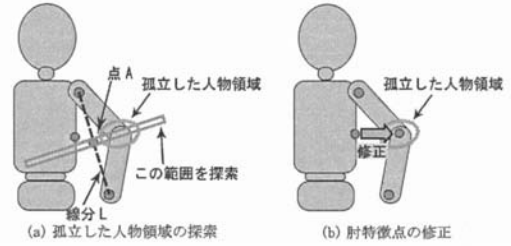


Fig.7 肘特徴点の修正

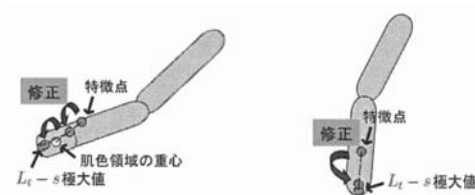


Fig.5 手先, 足先特徴点の修正 (左:手先, 右:足先)

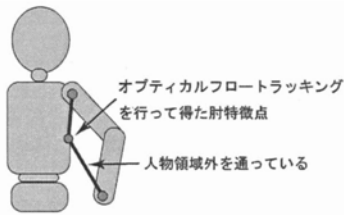


Fig.6 肘特徴点のオプティカルフロートラッキング失敗例

める。

2. Fig.7(a)のように, 点Aから線分Lと垂直な2方向へそれぞれ上腕ボーンの長さ分だけ探索し, 孤立した人物領域を見つける。孤立した人物領域とは, 探索範囲において, 人物領域外に囲まれた人物領域のことである。
3. 孤立した人物領域が見つかった場合, Fig.7(b)のように肘特徴点を修正する。

膝特徴点は, オプティカルフロートラッキングを行って得た特徴点について, 股関節特徴点, 足先特徴点との長さが, それぞれ上腿ボーンおよび下腿ボーンより長くないように修正する。

2.3.2 特徴点の3次元座標推定

頭・胴体領域の特徴点は, 映像の人物が光軸方向に移動しないという制約があるため, 奥行き方向の座標値を0とする。腕, 脚領域の特徴点は, 映像が平行投影であるとみなし, 各ボーンの長さと同様にボーンの間となる推定特徴点間の距離から三平方の定理を用いて計算する。

2.3.3 モーションデータ計算

頭・胴体領域において, ルートノードの平行移動量, 及び回転量を, 2.3.1での手法と同様に人物シルエットの重心, 慣性主軸の傾きから計算する。

腕, 脚領域については, 逆運動学の手法の一つであるLimbIK[4]を用いて各特徴点の回転角度を計算する。LimbIKはFig.8のようにR1関節とR2関節の座標からRy関節の回転量を計算し, swivel角を変化させたときにRy関節が描く軌跡を求める。swivel角を与えることでR1関節の回転量を計算することができる。本システムでは, swivel角を肘(膝)特徴点の位置から以下のように $-90^{\circ} \sim 90^{\circ}$ の範囲で求め, 与えている。

1. 肩から手先へのベクトルと垂直で, 肘特徴点へ向かうベクトルVを求める。
2. ベクトルVと, LimbIKにおける $swivel=0^{\circ}$ のときの肘方向ベクトルの成す角度 θ を求める。
3. θ が -90° 以下のときは $\theta = -180^{\circ} - \theta$, 90° 以上のときは $\theta = 180^{\circ} - \theta$ とすることで, θ を $-90^{\circ} \sim 90^{\circ}$ の範囲に収める。
4. θ をswivel角に設定する。

3 システムの実装と実行結果

システムを評価するために, 背伸びの運動(ラジオ体操第一より), 足踏み, コマネチ, 腕が大きく動く動作, ダンスのような動作の映像を本システムに入力し, モーションデータを出力した。そして, システムが出力したモーションデータをRokDeBone2[5]を用いて3Dキャラクターに適用し, 3Dキャラクターアニメーションを作成した。動作推定の結果をFig.9に示す。

Fig.9から, 本システムにより概ね正確に動作推定が行われたことがわかる。しかし, 動作D(Fig.9(d))のように動作の途中で人物の肌色領域(手, 顔)が重なって

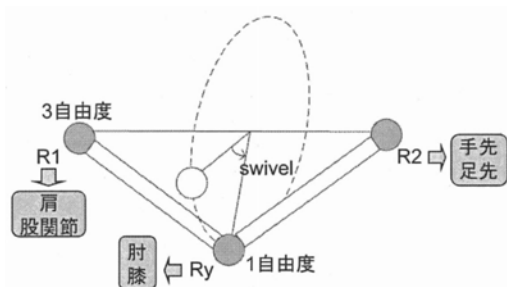


Fig.8 LimbIK

いる場合や、動作 E(Fig.9(e)) のように動作の途中で脚が交差している場合には動作推定を失敗している。これは、手、顔と肌色領域の対応付けの失敗や自己遮蔽対策が不十分であることが原因であると考えられる。

なお本システムの実装は、開発言語に C++ 言語を、画像、動画処理に OpenCV[6]、DirectX9 を使用した。また、CPU: Intel Core2 Quad CPU 2.66Ghz、メモリ: 2GB、GPU: NVIDIA Quadro FX 4500 の PC 上で本システムを実行したところ、動作速度は約 5fps となった。すなわち、30fps で 10 秒間の映像を処理するのに 1 分を要している。

4 まとめと今後の課題

本稿では単眼カメラで撮影された映像から、映像に映っている人物の動作を推定し、3D モーションデータを出力するシステムについて述べた。

今後、対応する動作の増加や、各特徴点の推定精度の向上により動作推定の完成度を高め、またユーザがシステムの推定ミスを手動に修正できるようなインターフェースを用意することにより、実用性の高いシステムを目指す。

参考文献

- [1] YouTube,
<http://jp.youtube.com/>
- [2] ニコニコ動画(夏),
<http://www.nicovideo.jp/>
- [3] 高橋和彦, 坂口竜己, 大谷 淳: "実時間非接触非装着型 3 次元人物姿勢推定法に関する一考察", 電子情報通信学会論文誌 D-II, Vol.J83-D2, No.5, pp.1305-1314 (2000).
- [4] D. Tolani, A. Goswami, and N. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs", Graphical Models 62(5), pp. 353-388, 2000.
- [5] RokDeBone2,
<http://www5d.biglobe.ne.jp/~ochikko/rokdebone.htm>
- [6] 奈良先端科学技術大学院大学 OpenCV プログラミングブック製作チーム; OpenCV プログラミングブッ

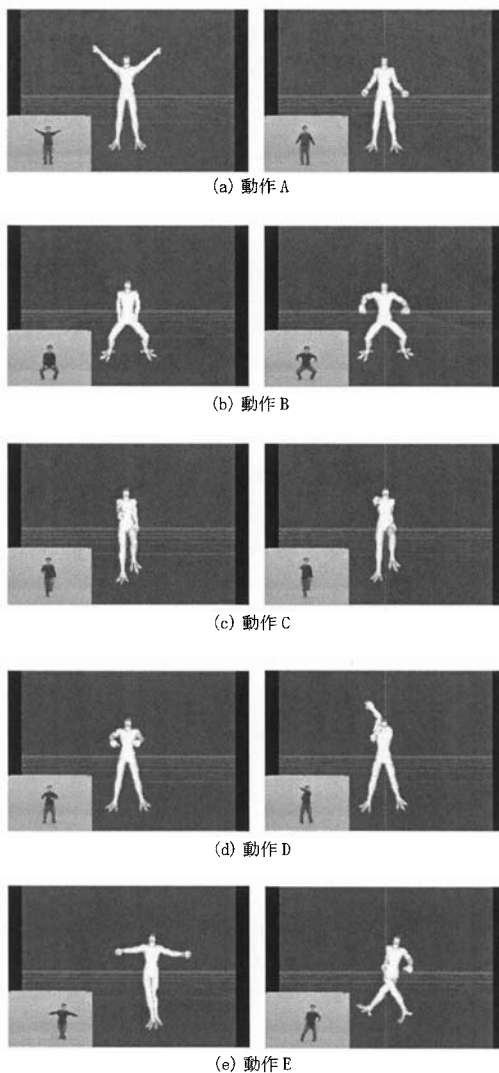


Fig.9 動作推定結果

ク, 毎日コミュニケーションズ, 2007