

## 多様な 3D キャラクタのための特徴関節による効果的な表現入力法

万谷 勇輝<sup>†</sup> 松田 浩一<sup>†</sup><sup>†</sup>岩手県立大学 ソフトウェア情報学部

## 1. はじめに

3D キャラクタを用いたアニメーションの動きの指定において、モーションキャプチャ(Mo-Cap)によって作成された動作のデータベースが存在するが、それを用いるにはキャラクタに合わせるための編集が必要になる。

今間らは Mo-Cap データにアニメーション独特の誇張表現と加速度に注目して編集する手法を提案している[1]。しかし、誇張の適用度合いをパラメータ設定する必要がある。

本研究では、誇張表現を(1)タイミング、(2)部位強調、(3)カメラワーク、による効果と定義し、これらのパラメータ設定をカメラからの実動作入力により自動決定する手法を提案する。本手法では複数の腕が複数あるなど多様なキャラクタでも一度の入力で制御可能とする。

## 2. 特徴関節による効果的な表現入力法の提案

アニメーションの表現[2]を分類した結果、以下の要素が誇張に関係すると考えた。

- (1) タイミング：キャラクタの印象が変わる。
- (2) 部位強調：動作の際に強調したい部位へ注意を呼び起こす。
- (3) カメラワーク：より効果的に動きを伝えるためにどこが動いているかを伝える。

ここで、全ての関節を誇張する意味はなく、動作の特徴を表す関節に絞り制御を行う。例えば、ボクシングなら両腕、サッカーなら両足が対応する。

本手法では、これらの特徴関節に対して、タイミングおよび部位強調の制御を行い、カメラワークによって、その誇張を効果的に伝える。

また、腕が複数あるなど人型とは違う場合の入力も、特徴関節の設定により対応可能となる。

An Effective Expression Input Method to Characteristic Joint for Various 3D Character.

Yuki MANYA<sup>†</sup>, Koichi MATSUDA<sup>†</sup>, <sup>†</sup>Faculty of Software and Information Science, Iwate Prefectural University

## 3. 提案手法のアルゴリズム

## 3.1. 初期設定

関節の構造部と各関節の角度情報部に分かれた BVH 形式をサポートし、あらかじめ 1 動作ごとに分けたデータを一覧として表示し、ユーザが選択したデータを対象データとする。

また、データの特徴関節と色マークをつけた人形との対応付けを図 1 のように設定しておく必要がある(色マークをつけるのは人形でなくても良い)。この時、特徴関節と連動して制御したい関節についても任意に設定可能である。

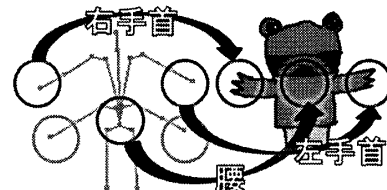


図 1：対応付け例

(黒丸：特徴関節，灰色の丸：連動する関節)

## 3.2. 実動作入力と対象データから情報の取得

特徴関節と対応付けた色マークによる実動作入力から、制御情報を取得する。

OpenCV を用いて、カメラ画像を取得し、色マークを追跡する(図 2)。取得した色マークの座標から、上下左右の移動量を求め、すべての色マークにおける最大移動量との比率(0~1)を入力情報として求める(図 3)。また、最大移動量がある色マークに対応する関節を強調関節とする。

対象データにおける特徴関節も同様に、各フレームにおける上下左右の移動量を求め、すべての特徴関節における最大移動量との比率を動作情報として求める。

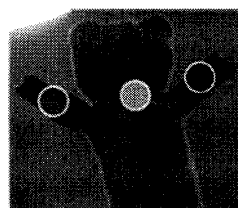


図 2：色マーク抽出

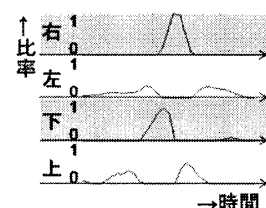


図 3：入力情報の例

### 3.3. アニメーションの表現を踏まえた制御

入力情報と動作情報を用いて、(1)タイミング、(2)部位強調、(3)カメラワーク、の 3 つの表現について制御を行う。

#### 3.3.1. タイミング

入力情報と動作情報の最大の位置をキーとし、入力情報のキー位置に合わせて対象データの角度情報を時間軸方向に伸縮させる(図 4)。

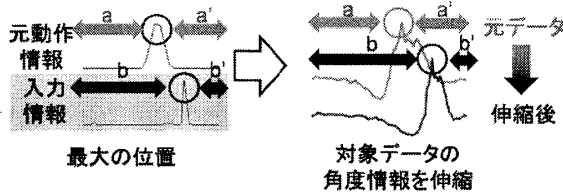


図 4: タイミングの制御

#### 3.3.2. 部位強調

関節の角度を大きくするとあり得ない方向に曲がる可能性があるため、逆に強調関節以外の角度を小さくする事で強調関節を際立たせる。

特徴関節のうち強調関節でない関節に対し、強調関節に対応する入力情報の最大と各関節における入力情報の最大との比率  $R$  と初期フレームの角度情報  $B$  から、各フレームにおける対象データの角度情報  $F_i$  を以下の式(1)で更新する。

$$F_i = B + (F_i - B)R \quad (1)$$

#### 3.3.3. カメラワーク

Mo-Cap データにはカメラの設定がないため、キャラクターの位置を動かすことによる擬似的なカメラワークによって実現する。

拡大で引きつけ、動作を見せるために縮小する 2 段階の制御を行う。切り替わり地点には強調関節が一番手前に来た時を用いる。

また、画面中央に強調関節が来るように、キャラクターの位置を設定する。

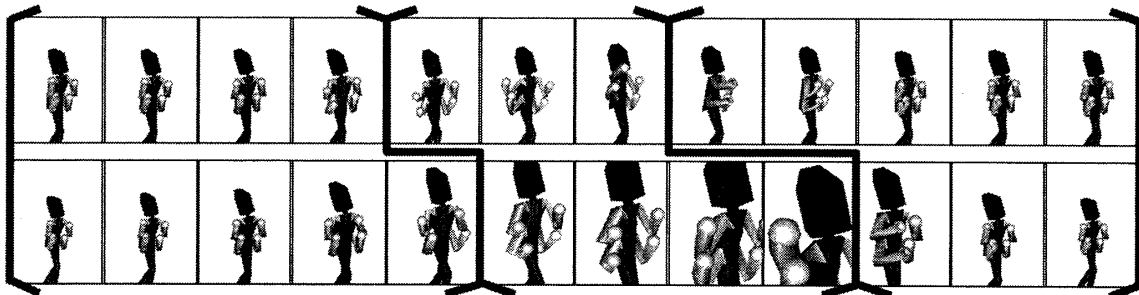


図 5: 作成したアニメーションから 20 フレームごと抽出  
(上: 元の動作, 下: 編集後の動作)

## 4. 実験

元の動作と作成した動作をキャラクターモデルに当てはめた結果を図 5 に示す。

キャラクターモデルには自作のアリ(腕 4 本、足 2 本)を用い、Mo-Cap データベース[3]からボクシングの 1 動作「右フック」を使用した。

入力には手に付け動かせるパペットを用い、図 1 のように対応付けた。主に右手の入力を行い、最初はゆっくりと、最後にかけて早くした。

その結果、元のデータと比べ、タイミングの変化によってタメや勢いができたことや、部位強調とカメラワークの変化によってどこが動いているかが伝わりやすくなったことから、効果的な編集が行えていることが確認できた。

## 5. おわりに

本研究によって、特徴関節のみの実動作入力から、アニメーション表現を考慮した制御により Mo-Cap データをパラメータ設定なしで誇張することが可能になった。また、複数の腕を持つようなキャラクターでも一度の入力で制御出来た。

## 参考文献

- [1] 今間, 近藤, 栗山, 古家, “モーションキャプチャデータを用いた加速度制御手法によるメンタルモーション生成”, 図学研究 第 39 巻 2 号(通巻 108 号), pp.3-10, 2005.
- [2] ハロルド・ウィテカー, ジョン・ハラス, 青木義朗: アニメーションのタイミング技法, ダヴィッド社, 1983.
- [3] CMU Graphics Lab Motion Capture Database: <http://mocap.cs.cmu.edu/>.