Vol.2013-CG-153 No.24
Vol.2013-CVIM-189 No.24
2013/11/29

リミテッドアニメ風表現のための モーションタイミング調整法

北村 真紀[1,a)]　金森 由博[2]　三谷 純[2]　福井 幸男[2]　鶴野 玲治[1]

概要：リミテッドアニメーションとは，同じ絵を 2 コマあるいは 3 コマ連続して用いることによって，毎秒 24 コマのアニメーションを構成する表現手法である．本研究ではモーションキャプチャのデータを入力とし，このデータからフレームを間引くことで動きのタイミングを制御し，「中無し」と「コマ撮り」の効果を模倣する．具体的には，動きが特に速いフレームを削除することで中無しのように速さを強調し，動きの小さいフレームを更新しないことでコマ撮りのように微小な動きを低減する．適用結果の動画により，リミテッドアニメ風のモーションタイミング調整が実現できたことを示す．

キーワード：リミテッドアニメーション，モーションキャプチャ

Motion Frame Omission for Cartoon-like Effects

Abstract: Limited animation is a hand-drawn animation style that holds each drawing for two or three successive frames to make up 24 frames per second. In this paper, we propose a simple method for automatically converting motion capture data to imitate the unique expressions of limited animation. Especially, we focus on the following two features related to motion timing; one is that limited animation does not show subtle motion because it omits almost-static inbetween frames. The other is that it exaggerates motion speeds by omitting inbetweens. The accompanying movie with animating cartoon characters demonstrates that our method can imitate the expressions of limited animation.

Keywords: Limited animation, Motion capture

1. はじめに

リミテッドアニメーションとは，同じ絵を 2 コマあるいは 3 コマ連続して用いる「コマ撮り」によって，毎秒 24 コマのアニメーションを構成する表現手法である．元々はコスト削減のために開発されたが，意識的に絵や動きに制限を加えた表現としても発展し特有のメリハリのある表現を可能にした．リミテッドアニメーションは手描きアニメーションのための手法であるが，3DCG の分野でもリミテッドアニメーションの表現技法が注目されてきている．

手描きアニメーションの手法である形状変形による誇張表現やコマ撮りの手法は，カートゥン調のキャラクタの動きを表現するのに適している．一方で，カートゥン調のキャラクタにモーションキャプチャのデータを適用することは適していない．このことに関して，著名なアニメーション映画監督の John Lasseter は次のように述べている [1].

> *"Motion capture from human actors will always look realistic... for a human. But apply that motion to a chicken and it will look like a human in a chicken suit."*

しかし，カートゥーン調の 3D キャラクタを扱った最近のゲームでは，制作コストの問題でモーションキャプチャのデータを用いることが多い．そこで本研究ではモーションキャプチャのデータを入力として，リミテッドアニメーションで用いられている表現技法の効果を模倣する手法を提案する．

リミテッドアニメーションの特徴として，本研究では次の 2 つに着目する．一つはコマ撮りによって動きが少ない

1 九州大学
Kyushu University
2 筑波大学
University of Tsukuba
a) maki.terai@gmail.com

© 2013 Information Processing Society of Japan

フレームを描かないため，細かい動きがないことである．もう一つは連続している動きの中間の絵を描かない「中無し」[2] という技法を用いることで動きの速さの強調を行っていることである．本研究ではこれらの効果を模倣するために，モーションキャプチャのデータの細かい動きを低減し，動きの速さの強調を行うことを目的とし，動きが大きい部分のフレームを抜いたり，動きが小さい部分のフレームをコマ撮りにすることで，これらの効果を実現する．

2. 関連研究

コンピュータグラフィックスにおけるカートゥンアニメーションのための様々な技法が提案されている．カートゥンアニメーションのような動きを表現するために，オブジェクトの形状を変形させて動きを誇張する研究がある [3–5]．これらの手法は動きのタイミングについては考慮していない．

3D キャラクタアニメーションの分野で動きのタイミングを制御する研究は数多くある [6–8]．これらの手法はキーフレームあたりの時間を変化させることで動きのタイミングを調整しており，フレームを削除しているものではない．一方で，本手法ではフレームを削除することで，動きの速さを強調している．これはリミテッドアニメーションの制作手法に基づくものである．

リミテッドアニメーションの制作手法に着目した研究として，Morishima らは元のアニメーションより少ないキーポーズを選択するために，モーションカーブをキーポーズを頂点とする折れ線で近似し，折れ線の長さができるだけ大きくなるような最適化を行った [9]．しかしこの手法では連続して間引くのは 3 コマまで，といった局所的な制御に向いていない．本研究では，連続して間引くコマ数に制約を与えられる，コマ撮りの再現を目指した単純な手法を提案する．さらに，著しく動きの速いコマを抜くことで，中無しの効果を模倣する．

3. アルゴリズム

提案手法は，入力されたモーションキャプチャのデータに対し，動きの少ないコマを抜いて細かい動きを低減し，著しく速いコマを抜いて速さを強調する．本研究では入力となるデータは「投げる」，「振る」といった一つの動作で構成されていると仮定し，それらを組み合わせた複合的な動作の場合は手動で分割している．

3.1 速さの定義

本研究では，入力となるモーションデータはフレーム数が N 枚でジョイントの数は n 個であるとする．フレームレートは毎秒 24 フレーム（24FPS）とし，フレーム間の時間間隔はすべて等しいものとする．さらに，k フレーム目（$k = 1, 2, \ldots, N$）でのジョイント i（$i = 1, 2, \ldots, n$）における座標を $\mathbf{p}_i^k \in \mathbb{R}^3$ とする．k フレーム目と $(k-1)$ フレーム目の各々のジョイント位置の二乗距離の総和 (SSD) を計算し，これを k フレーム目での速さと定義する．

$$SSD(k) = \sum_{i=1}^{n} \| \mathbf{p}_i^k - \mathbf{p}_i^{k-1} \|^2 \qquad (1)$$

Algorithm 1 中無しによる速さの強調

$N_{io} \leftarrow$ user-specified number of inbetween omission
for $i \leftarrow 1$ to N_{io} **do**
  $k_{max} \leftarrow \mathrm{argmax}\, SSD(k)$
  delete k_{max}-th frame
  $k_{min} \leftarrow \mathrm{argmin}\, SSD(k)$ s.t. $\#\textsc{Duplicated}(k-1) < 3$
  duplicate $(k_{min}-1)$-th frame
end for

3.2 中無しによる速さの強調

中無しの効果を模倣するために，入力データの動きが速い部分の速さを強調する処理を行う (図 1)．まず，SSD が最大となるような k_{max} 番目のフレームを求め，このフレームを削除する．フレームを削除すると，モーション全体のフレーム長が変わってしまうため，フレーム長を一定に保つために，変化が少ない部分にフレームを追加する．k_{min} 番目の SSD が最小となる場合，$(k_{min}-1)$ 番目のフレームをコピーして挿入する．同じフレームが 3 フレームより多くコピーされると動きが静止して見えるため，すでに 3 フレームコピーされたフレーム間にはコピーせず，SSD が次に小さいフレーム間にフレームを追加する．本手法では著しくモーションのタイミングが変わることを防ぐために，フレーム長を一定に保つようにしているが，フレーム長が変わってよい場合は，この処理は不要である．フレームの削除と挿入の後に，再び SSD を再計算し，これらの処理を繰り返し行うことで中無しのように速さを強調する効果が得られる．この処理の疑似コードを Algorithm 1 に示す．$\#\textsc{Duplicated}(k)$ は k フレーム目を複製して新たに挿入したフレーム数を意味する．

3.3 コマ撮りによる細かい動きの低減

次にコマ撮りの処理を模倣することで，微細な動きの低減を行う (図 2)．SSD が最小となるような k_{min} 番目のフレームを求め，k_{min} 番目のフレームに $(k_{min}-1)$ 番目のフレームを上書きすることで実現する．この処理を貪欲的に繰り返すことで，動きが少ないところの姿勢行列に変化がなくなり，細かい動きが低減される．同じフレームが続くと動きが完全に止まってしまうため，中無しと同様に同じフレームが連続できる数は 3 フレームまでとする．この処理の疑似コードを Algorithm 2 に示す．$\#\textsc{Copied}(k)$ は k 番目のフレームを後続のフレームに上書きした数を意味している．

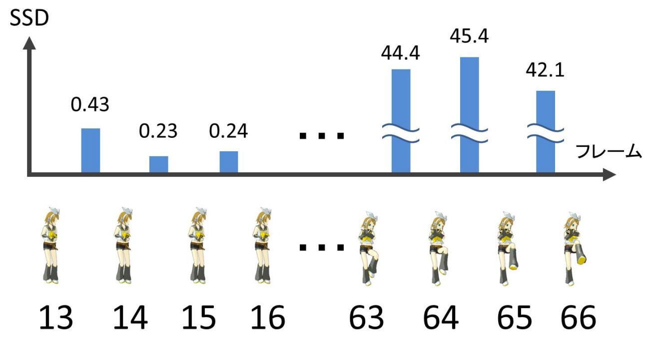

図 1 各フレーム間の SSD のグラフと姿勢．縦軸はフレーム間のジョイントの座標の二乗距離の総和 (SSD) を示す．横軸はフレームを示す．64 フレーム目と 65 フレーム目の間の移動量が最も大きいので 65 フレーム目が中無しによって抜かれる．また，14 フレーム目と 15 フレーム目の移動量が小さいので，このフレーム間に 14 フレーム目のデータがコピーされる．

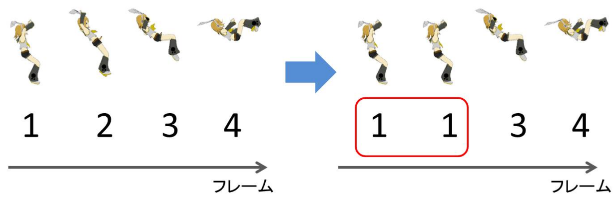

図 2 コマ撮り．SSD(1) が小さいため，2 フレーム目に 1 フレーム目をコピーする．こうすることで，細かい動きが低減される．

Algorithm 2 コマ撮りによる細かい動きの低減

$N_{hf} \leftarrow$ user-specified number of holding frames
for $i \leftarrow 1$ to N_{hf} **do**
  $k_{min} \leftarrow \mathrm{argmin}\, SSD(k)$ s.t. $\#\textsc{Copied}(k-1) < 3$
  copy $(k_{min}-1)$-th frame to k_{min}-th frame
end for

3.4 代替手法についての検討

中無しを行う際，本手法ではフレームを抜いた後に SSD を再計算し，再びフレームを抜いている．別の方法として，全フレームに対し SSD を計算して SSD が大きい上位のフレームを一度に抜くという手法も考えられるが，この手法では本手法に比べ元のモーションからの変化が少ない (図 3)．本手法では抜かれるフレームが連続するため，同じ箇所からフレームが抜かれ動きの速さが強調される．一方で，一度に抜く手法では抜かれるフレームが連続せずに様々な箇所から抜かれ，元のモーションとの変化が少なく，動きの速さが強調されにくい．

中無しとコマ撮りのどちらを先に行うかという順番に関して，本手法では中無しを行った後にコマ撮りを行っているが，この順番を逆にして実験を行ったところ大差が見られなかった．理由は，コマ撮りで抜かれるフレームは動きが小さい部分であり，中無しで抜かれる可能性があるフレームには互いに影響を及ぼさないからである．

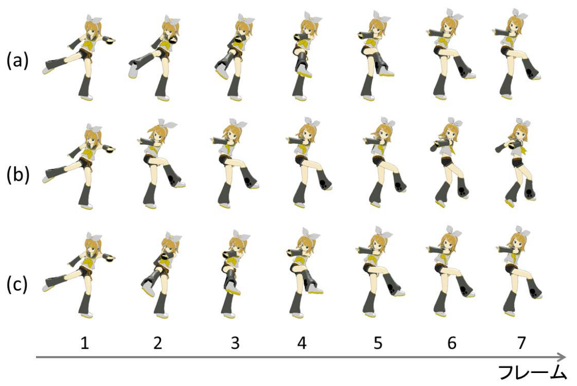

図 3 本手法 (b) とフレームを一度に抜く方法 (c) の比較．(a) は入力モーションデータ．本手法では 2 フレーム目から 5 フレーム目までが抜かれるが，フレームを一度に抜く方法では 2 フレーム目しか抜かれないため，繰り返し抜く方が動きの速さが強調されやすいことがわかる．

4. 結果

本手法を C++と Qt Library を用いて実装した．実行環境は Intel Core i7-2760QM 2.40GHz CPU の PC を用いて実行した．モーションキャプチャのデータは Mocapdata.com [10] と CMU Graphics Lab Motion Capture Database [11] で公開されているものを用いた．3D モデルは MikuMikuDance [12] に含まれているものを用いた [*1]．本手法を用いて得られた結果を図 4, 5, 6, 7 に示す．これらの図の上段は入力に用いたモーションキャプチャのデータであり，下段は本手法で得られた結果である．いずれも 24FPS で出力したものの一部を表示している．本手法の結果は動きの速さを強調するものであり，スナップショットではわかりにくいため，動画ファイルを参照されたい [*2]．図 4 では，上段の 3 フレーム目から 6 フレーム目が中無しによって削除されているため，キャラクタの右腕の動きの速さがはっきりと強調されている．同様に，図 5 は蹴る動作に本手法を適用したものである．上段の 2 フレーム目からフレームまでが削除され，右足の動作のスピード感が増していることが確認できた．図 6 はテニスのスウィングの動きに適用したものである．2 フレーム目から 5 フレーム目の動きが削除されていることで，振りの速さの強調が確認できた．図 7 は野球のスウィングの動きである．2 フレーム目から 7 フレーム目の動きが削除され，速さが強調されている．

図 8 は，図 4 と同じモーションの一部の区間のキーフレームを表示したものである．青い棒があるフレームは

[*1] 鏡音リンは，クリプトン・フューチャー・メディア株式会社の登録商標である．
[*2] 動画は下記の URL にある．
`http://vacation.aid.design.kyushu-u.ac.jp/~maki/projects/limited_animation/limited_animation_ja.html`

© 2013 Information Processing Society of Japan

Vol.2013-CG-153 No.24
Vol.2013-CVIM-189 No.24
2013/11/29

表 1 実験結果

ファイル名	フレーム長	N_{io}	N_{hf}	処理時間 [ms]
Pitch	129	4	33	69
Kick	92	4	30	28
Tennis	88	4	29	27
Baseball swing	120	6	40	63

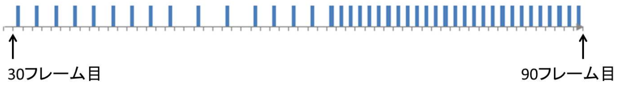

図 8 図 4 のモーションの一部の区間のキーフレーム.

キーフレームであり，棒がないフレームはコマ撮りによって削除され，前フレームをコピーして挿入したフレームであることを示している．動きが大きいところはキーフレームが多く，少ないところはキーフレームが少なくなっている．これを動画としてみると微小な動きが低減されたことが確認できた．

表 1 は，それぞれのモーションのフレーム長，中無しで抜いた総フレーム数 N_{io}，コマ撮りでコピーした総フレーム数 N_{hf}，処理時間をまとめたものである．中無しの数 N_{io} とコマ撮りの数 N_{hf} に関しては，何枚抜けば適切かという枚数はアーティストの感覚に依存するため一意に決めることは難しく，本手法ではユーザが枚数を設定できるようにしている．我々が行った実験では，中無しの数 N_{io} は 4 フレームから 6 フレームくらいが適当であり，それ以上抜くと動きがとびすぎて一連の動作に見えなくなることがわかった．またコマ撮りによってコピーされるフレームの数 N_{hf} はフレーム長の 3 分の 1 くらいが適当であった．

5. 結論

本研究ではリミテッドアニメーションの表現技法のコマ撮りと中無しに着目し，モーションキャプチャのデータを入力としてそれらの表現の模倣を試みた．SSD が大きいフレームを削除することで中無しの効果が再現でき，また，SSD が小さいところをコマ撮りにすることで，微小な動きが低減されたことを確認した．

本手法では SSD を計算する際，ワールド座標系でのジョイントの座標値で計算している．このためワールド座標系では SSD が大きくても，カメラとオブジェクトが離れているときはスクリーン座標系での SSD は小さいというように，ワールド座標系での SSD とスクリーン座標系での SSD が一致しない場合もありうる．これに対応するためにスクリーン座標系での移動量も考慮する必要がある．

本手法では誇張のためのモデルの変形は行っていないが，リミテッドアニメーションの表現では変形も重要な要素である．本手法とカートゥンアニメーションフィルタ [4] のような誇張表現を組み合わせることで，よりリミテッドアニメーションに近い表現ができると考えられる．

謝辞 本研究に際して，様々なご指導を頂いた豊橋技術科学大学の栗山繁教授及び，株式会社スクウェア・エニックスの向井智彦氏に感謝申し上げる．また，本研究は日本学術振興会の特別研究員奨励費（課題番号 25・5451）の支援により実施された．

参考文献

[1] Lasseter, J.: Tricks to animating characters with a computer, *SIGGRAPH Comput. Graph.*, Vol. 35, No. 2, pp. 45–47 (2001).
[2] 尾沢直志：アニメ作画の仕組み，ワークスコーポレーション (2004).
[3] Chenney, S., Pingel, M., Iverson, R. and Szymanski, M.: Simulating cartoon style animation, *Proceedings of the 2nd international symposium on Non-Photorealistic Animation and Rendering (NPAR)*, pp. 133–138 (2002).
[4] Wang, J., Drucker, S. M., Agrawala, M. and Cohen, M. F.: The cartoon animation filter, *ACM Trans. Graph.*, Vol. 25, No. 3, pp. 1169–1173 (2006).
[5] Li, Y., Gleicher, M., Xu, Y.-Q. and Shum, H.-Y.: Stylizing motion with drawings, *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, pp. 309–319 (2003).
[6] Hsu, E., da Silva, M. and Popović, J.: Guided time warping for motion editing, *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, pp. 45–52 (2007).
[7] McCann, J., Pollard, N. S. and Srinivasa, S.: Physics-based motion retiming, *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, pp. 205–214 (2006).
[8] Kwon, J.-y. and Lee, I.-K.: An animation bilateral filter for slow-in and slow-out effects, *Graph. Models*, Vol. 73, No. 5, pp. 141–150 (2011).
[9] Morishima, S., Kuriyama, S., Kawamoto, S., Suzuki, T., Taira, M., Yotsukura, T. and Nakamura, S.: Data-driven efficient production of cartoon character animation, *SIGGRAPH 2007 sketches* (2007).
[10] *Mocapdata.com*, http://www.mocapdata.com/.
[11] *CMU Graphics Lab Motion Capture Database*, http://mocap.cs.cmu.edu/.
[12] *MikuMikuDance*, http://www.geocities.jp/higuchuu4/.

ⓒ 2013 Information Processing Society of Japan

図 4 Pitch．入力となるモーションデータ（上段）と本手法で得られた結果（下段）の一部のシーケンスを示す．フレーム長が 129 フレーム，中無しを 4 フレーム，コマ撮りを 69 フレーム適用．右腕の動きの速さが強調されている．

図 5 Kick．フレーム長が 92 フレーム，中無しを 4 フレーム，コマ撮りを 30 フレーム適用．足の動きの速さが強調されている．

Vol.2013-CG-153 No.24
Vol.2013-CVIM-189 No.24
2013/11/29

図 6 Tennis．フレーム長が 88 フレーム，中無しを 4 フレーム，コマ撮りを 29 フレーム適用．右腕の動きの速さが強調されている．

図 7 Baseball swing．フレーム長が 120 フレーム，中無しを 6 フレーム，コマ撮りを 40 フレーム適用．両腕の振りの速さが強調されている．

ⓒ 2013 Information Processing Society of Japan