

# コンピュータアニメーションの中割りアルゴリズムに関する一考察 II

坂本 真人<sup>a)</sup>・黒木 脩平<sup>b)</sup>

## A Study on Inbetweening Algorithm for Computer Animation II

Makoto SAKAMOTO, Shuhei KUROKI

### Abstract

Computer animation is now applied widely in the entertainment, and is also being used in education, industry, scientific research, and so on. By the way, it is important for animation to think about shape, color, position, structure, quality, and so on. Especially, it is of very importance to outline an object. However, the animator must draw a lot of frames of the object and spend much time and labor, so computerized inbetweening methods for animation have been proposed. Above all, the Miura inbetweening algorithm is very famous, but this algorithm has a problem which appropriate frames are not produced from two keyframes in the case of reducing data. In this paper, we propose a revised algorithm in order to improve such a weak point, especially rotation part. The algorithm is implemented in Java on a personal computer.

**Keywords:** algorithm, computer animation, inbetweening, Java, keyframe, moving point

### 1. はじめに

近年における日本のアニメ市場規模は年々大きくなっており、2015年には初めてアニメ市場全体で約2兆円に達している<sup>1)</sup>。主な要因として深夜アニメ数の増加や劇場アニメの好調、アニメのグッズや関連商品の売り上げ、海外売り上げが大きく伸長したことが挙げられる。中でも深夜アニメは地方を舞台とする作品も多く、「聖地巡礼」と呼ばれるアニメのファンがアニメの舞台になった場所に訪れることにより、地方の観光や経済にも良い影響を与えている<sup>2)</sup>。

しかし、アニメ市場が好調で成長を続けているのに対し、実際にアニメを制作するアニメーション制作者の実態はあまり良いとは言えない。アニメーション制作者の1日あたりの平均作業時間は約11時間、年収は約333万円となっていて、労働時間は長く、年収も日本の民間事業所に勤務している給与所得者の平均を下回っている<sup>3)</sup>。原因として、劇場アニメに比べて深夜アニメなどに割り当てられる資金は少ないこと、またアニメ業界の競争が年々大きくなっており、アニメ制作会社は少ない資金で多くの数のアニメを作らなければならないことが挙げられる。さらにはアニメの作画のクオリティも求められているので、作業時間も長くなってしまふ。

そこで、アニメ制作において最も時間のかかる中割りの工程をコンピュータにより支援する中割り法の研究を行った。今回の研究では代表的な中割りのアルゴリズムであるMiuraのアルゴリズムをもとに作成されたアルゴリズムを修正、改良した。修正前と修正後のアルゴリズムに対し様々な画像データの中割りを行い、結果を比較して修正後の中割りアルゴリズムの有効性を調べた。また、なめらかな中間画像を生成する方法を考察した<sup>4)7)</sup>。

### 2. 中割り法

従来の手書きアニメーションにおける中割り法(inbetweening)としては、大量の絵を多人数の協同作業者が描く“キーフレーム法”が採用されてきた。すなわち、アニメーションの流れの中で特に重要な絵を1人の“キーアニメーター”が描き、これらの絵を参考にしながら中間の多数の絵を複数の“補助アニメーター”が描き、全体を統合して1本のアニメーションを効果的に完成していく。キーアニメーターの描く絵を、“キーフレーム(key frame)”と呼ぶ。図1にキーフレームアニメーション制作の概略を示す。

a) 工学教育研究部 情報システム工学科 准教授

b) 情報システム工学科 学部4年生

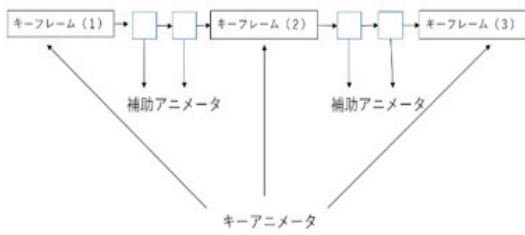


図1：キーフレームアニメーションの概略図<sup>7)</sup>。

中割り作業は経験を必要とする作業である。キーアニメーターの描いた絵から指示に沿って多数の絵を描いていくが、なめらかな動きやキャラクターの表情を描くには様々な技術が要求されるため時間と多数の補助アニメーターが必要になっていた。しかし、コンピュータを用いた中割り法が実用段階に達してきた。いわゆるコンピュータキーフレーム法の登場である。キーフレームは時間軸に沿って並べられ、その間のフレームはソフトウェアによって自動補間される。アニメーターは生成された各フレームを編集し、細かい部分を調整していく。

コンピュータを使ったキーフレーム法には以下のように利点がある。

- 中間画像の自動生成と高速モニタリングが可能になる。
- コンピュータを用いて制作したフレーム画像は中割りの次の処理の彩色が容易になる。
- 精密な動きを表現できる。
- 三次元的な動きの表現が可能になる。

### 3. アルゴリズム

#### 3.1. 中割りアルゴリズムについて

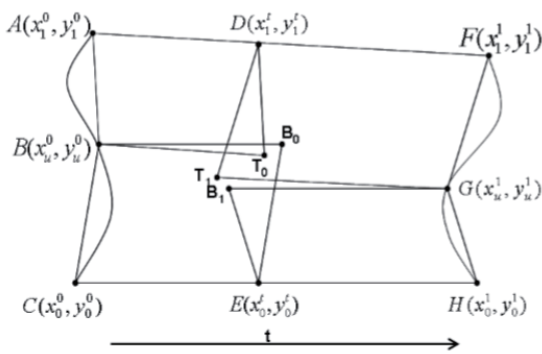


図2：中割りアルゴリズム<sup>8)</sup>。

図2における端点D,Eについて、以下の式(1)で計算する。

$$\begin{aligned} x^t &= (1-t)x^0 + tx^1 \\ y^t &= (1-t)y^0 + ty^1 \end{aligned} \quad (1)$$

次にKF<sub>0</sub>を基準と考え、AD, ABを二辺とする平行四辺形を考える。

これによって得られる第4点をT<sub>0</sub>(x<sub>T0</sub>, y<sub>T0</sub>)とする。また、CB, CEを二辺とし同様にB<sub>0</sub>(x<sub>B0</sub>, y<sub>B0</sub>)を求める。この2点を用いて次式よりM<sub>0</sub>(x<sub>M0</sub>, y<sub>M0</sub>)を求める。

$$\begin{aligned} x_{M0} &= (1-K_0)x_{T0} + K_0x_{B0} \\ y_{M0} &= (1-K_0)y_{T0} + K_0y_{B0} \end{aligned} \quad (2)$$

ただし、 $K_0 = \frac{\sqrt{(x_u^0-x_1^0)^2+(y_u^0-y_1^0)^2}}{\sqrt{(x_0^0-x_1^0)^2+(y_0^0-y_1^0)^2}}$

次にKF<sub>1</sub>基準とし、同様にDF, FGよりT<sub>1</sub>(x<sub>T1</sub>, y<sub>T1</sub>)を、GH, HEよりB<sub>1</sub>(x<sub>B1</sub>, y<sub>B1</sub>)を求め、式(3)より、M<sub>1</sub>(x<sub>M1</sub>, y<sub>M1</sub>)を求める。

$$\begin{aligned} x_{M1} &= (1-K_1)x_{T1} + K_1x_{B1} \\ y_{M1} &= (1-K_1)y_{T1} + K_1y_{B1} \end{aligned} \quad (3)$$

ただし、 $K_1 = \frac{\sqrt{(x_u^1-x_1^1)^2+(y_u^1-y_1^1)^2}}{\sqrt{(x_0^1-x_1^1)^2+(y_0^1-y_1^1)^2}}$

式(2)、(3)より求めたM<sub>0</sub>, M<sub>1</sub>より中間画像の座標M(x<sub>M</sub>, y<sub>M</sub>)を式(4)より求める。

$$\begin{aligned} x_M &= (1-L)x_{M0} + Lx_{M1} \\ y_M &= (1-L)y_{M0} + Ly_{M1} \end{aligned} \quad (4)$$

ただし、 $L = \frac{\sqrt{(x_1^1-x_1^0)^2+(y_1^1-y_1^0)^2}}{\sqrt{(x_1^1-x_1^0)^2+(y_1^1-y_1^0)^2}}$

式(4)を用いて、時間パラメータtを一定にしたときの座標Mを直線で接続することにより、時間tにおける中間画像が得られる。ここで、実際のLの値は、tの値と等しくなる。

ところで、KF<sub>0</sub>, KF<sub>1</sub>において極端に関連がない場合、中間画像が歪んでしまう。そこで、KF<sub>0</sub>自身に位置パラメータと回転パラメータθを入力することにより並進及び回転を施し、最後の画像KF<sub>1</sub>に幅を持たす工夫をした。

まず、KF<sub>0</sub>とKF<sub>1</sub>の2つの画像データより前述のアルゴリズムから中間画像を求め、座標をM<sub>p0</sub>(x<sub>0</sub>, y<sub>0</sub>)とする。そして式(5)より時間tにおいてθだけ回転した座標M<sub>p1</sub>(x<sub>1</sub>, y<sub>1</sub>)が得られる。

$$\begin{aligned} x_1 &= x_0 \cos \theta + y_0 \sin \theta \\ y_1 &= -x_0 \sin \theta + y_0 \cos \theta \end{aligned} \quad (5)$$

本研究では先行研究の中割りアルゴリズムを元に(4)、(5)の式を実装した<sup>9)</sup>。

#### 3.2. 今回の研究

本研究で修正した中割りアルゴリズムの有効性を確かめるために、修正前の中割りアルゴリズムと比較しながらいくつかの画像データの中間画像を生成する。先行研究より、この中割りアルゴリズムは拡大縮小、平行移動、変形の中割りはなめらかな中間画像が生成できたが、回転を含む画像の中割りになると中間画像が歪んでしまうことが確認されているので、今回は回転を含む様々な画像の中割りに絞ってシミュレーションを行う。

回転を含む画像として、

- 画像の一部を回転させたとき
- 画像全体を回転させたとき

この2パターンについてシミュレーションを行い、なめらかな中間画像が生成できる方法の考察、アルゴリズムの修正を行ってきたい。

#### 4. シミュレーション結果

修正前と修正後のアルゴリズムを比較しながらシミュレーション結果の一部の中間画像(図3~図8)を表示する。今回使用した言語はJavaである。

- 足を90度曲げるアニメーション(節点数13)



図3：足(90度)修正前.



図4：足(90度)修正後.

- 手を90度曲げるアニメーション(節点数38)

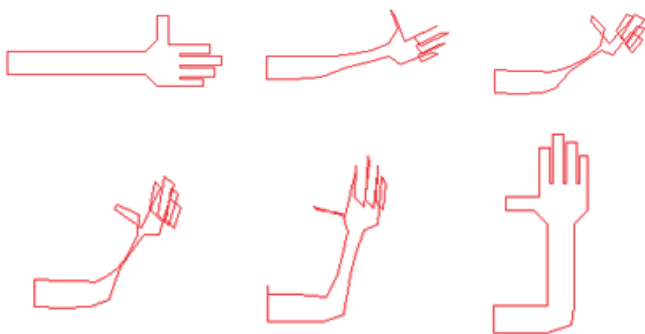


図5：手(90度)修正前.

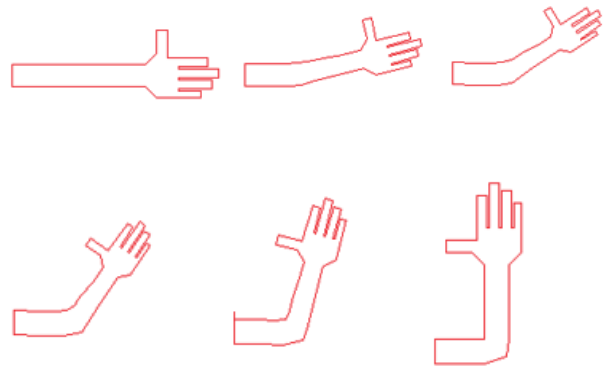


図6：手(90度)修正後.

- 長方形の90度の回転アニメーション



図7：長方形(90度)、節点数7.



図8：長方形(90度)、節点数31.

#### 5. 考察と今後の課題

生成した中間画像の結果から、修正後の中割りアルゴリズムの方がなめらかな中割りができることが確認できた。特に図3~6のように腕や足のアニメーションについては、回転の途中で線が交差して歪んでしまい幅が小さくなっていたのが改善された。手のような細かい部分もきれいに中割りができている。

次に、図7、8のように同じ画像で修正後アルゴリズムを使用して節点数を増やしてシミュレーションしたところ、よりなめらかな中間画像が生成できた。2枚の画像間での変化が大きい部分、特に注目して中割りを行いたい部分には節点数を増やすことによってなめらかな中間画像が生成できることが分かった。

また、今回のシミュレーションで画像の中の端点に当たる始点と終点の部分が歪んでしまうことが確認できた。これは中間画像を求める計算上、端点として始点と終点を先に計算しているために歪みが発生していた。そこで始点と終点を再度設定することで歪みを小さくすることができた。

今後の課題として、生成された中間画像のサイズが小さくなってしまふ場合があると確認できたのでその点について改善が必要である。

また、本研究では2次元の回転の中間画像を生成について研究したが、今後の課題として奥行きを考慮した3次元の動きや回転

についての検証が必要になる。

## 6. おわりに

本研究では、アニメ制作において最も時間のかかる中割りの工程をコンピュータにより支援する中割り法を用いた中間画像生成について行った。先行研究で作成されたアルゴリズムではできなかった画像の回転を含む動きの中割りについて検証とアルゴリズムの修正を行い、修正前よりなめらかな中間画像を生成することができたが、まだまだ2次元、3次元の動きや回転についての研究が必要になる<sup>10)</sup>。

## 参考文献

- 1) 一般社団法人日本動画協会 報告書：アニメ産業レポート，2016.
- 2) 酒井亨：アニメが地方を救う!!，2016.
- 3) 一般社団法人日本アニメーター・演出協会：アニメーション制作者実態報告書，2015.
- 4) 沓沢淳之介：コンピュータアニメーション技法，情報処理，Vol. 29, No. 10, pp. 1090-1096, 1988.
- 5) Miura, T., J. Iwata, and J. Tsuda: An Application of Hybrid Curve Generation, Spring Joint Computer Conference, 1967.
- 6) Reeves, W. T.: Inbetweening for Computer Animation Utilizing Moving Point Constraints, Computer Graphics, Vol. 15, No. 3, pp. 263-269, 1981.
- 7) 友添直子：コンピュータアニメーションの中割り法に関する研究、平成 17 年度卒業論文，宮崎大学工学部情報システム工学科，2005.
- 8) 安居院猛，中嶋正之，大江茂：コンピュータアニメーション，産報出版，1983.
- 9) 青野雅樹：Java で学ぶコンピュータグラフィックス，2001.
- 10) 古澤知英，福里司，岡田成美，平井辰典，森島繁生：正面および側面イラストからのキャラクター顔回転シーンの自動生成，2014.