

アニメーションの中割りアルゴリズムによるキャラクターの描写

坂本 真人^{a)}・石崎 裕一朗^{b)}・飯干 淳志^{c)}

Description of the Character by the Inbetweening Algorithm for Animation

Makoto SAKAMOTO, Yuichiro ISHIZAKI, Atsushi IIBOSHI

Abstract

Computer animation is now applied in the scientific research, education, industry, entertainment, and so forth. On the other hand, recently, each local tourist city performs various actions for the restoration of sightseeing spot. Therefore, we would like to cooperate with local tourist city as the sightseeing support by using computer animation. 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, and would like to apply computer animation to sightseeing support. The algorithm is implemented in Java on a personal computer.

Keywords: character, computer animation, inbetweening, Java, keyframe, moving point, sightseeing

1. はじめに

国内宿泊旅行者数は、1990年まではゆるやかに増加してきたが、その後現在に至るまで目立った増加は見られない⁴⁾。各地方の観光都市は、観光地としての復旧を目指し様々な取り組みを行っている。その一環として、制作過程を簡単にする方法の一つであるコンピュータの中割り法を用いた各観光都市のご当地キャラクターなどのアニメーションの制作を試みることにした。



図1. ちほまる⁵⁾



図2. 「ちほまる」のキャラクター⁵⁾

本研究では、我々が提案した中間画像を生成する修正版アルゴリズムを用い、キャラクターのアニメーションを作成した。今回は、高千穂のまるうど酒造の「ちほまる」(図1)という甘酒と乳酸菌を混ぜた商品のキャラクターを用いる(図2)。

2. 中割り法

1960年代の初頭から映画やテレビジョンなどに次々とコンピュータアニメーションが用いられるようになった。特に、超LSIの出現や画像情報処理技術のハードウェアとソフトウェアの両面における進歩は、コンピュータアニメーションの制作をさらに活性化させた。しかし、コンピュータアニメーションではまだ多くの問題点をかかえている¹⁾。その問題点の1つとして制作コストが高価であり、制作に時間がかかりすぎるといった点が挙げられる。そこで、制作過程を簡単にする方法の一つとして中割り法(in-between)がある。

従来の手書きアニメーションにおける中割り法としては、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

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

c) (株)高千穂ムラたび 代表取締役

されてきた。すなわち、アニメーションの流れの中で動作の主要な変化点の絵を1人の“キーアニメータ”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”が描いて全体を統合しながら1本のアニメーションを効果的に完成していく。キーアニメータの描く絵を、“キーフレーム (key frame)” と呼ぶ。図3にキーフレームアニメーション制作の概略を示す。

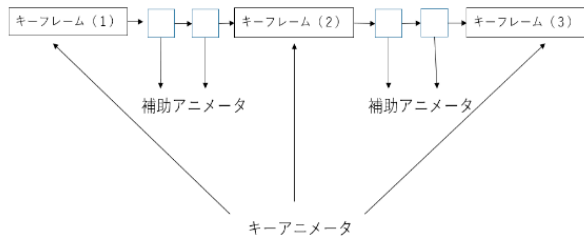


図3. キーフレームアニメーションの概略図¹⁾

中割り作業は、経験が必要とする作業であり、またアニメータ独特の個性表現や細やかな動きの表現、さらに芸術的な技能まで要求される分野であった。そのため、多大な時間や労力を必要としてきた。しかし、コンピュータアニメーションが登場してからは制作がかなり効率的になってきた。

3. 原理

3.1 Miura アルゴリズム

コンピュータアニメーションにおける中割りアルゴリズムとしてはMiura アルゴリズムがある²⁾。このアルゴリズムでは、データを連続的に入力した場合、つまり2つのキーフレームの関連が強い場合、物体の動きに自然さを与えることができる。

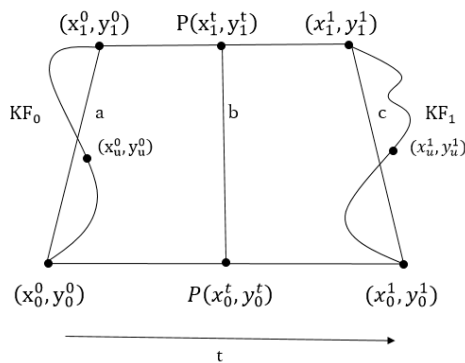


図4. Miura アルゴリズム³⁾

図4において、KF₀とKF₁は入力データ (キーフレーム) であ

り、画像の端から順に3個のデータごとに区切ったものの1つである。これをパッチとも言う。Miura アルゴリズムは、以下の式により時間 t における中間画像の座標P(x_u^t, y_u^t)を求めるものである。なお、中間画像の端点P(x₀^t, y₀^t)およびP(x₁^t, y₁^t)については式 (1) により求める。

$$x^t = (1 - t)x^0 + tx^1$$

$$y^t = (1 - t)y^0 + ty^1 \quad (1)$$

その他については、次の式 (2) より求める。

$$P(x_u^t, y_u^t) = (1 - t)A_0 + tA_1 \quad (2)$$

ただし、

$$A_0 = W_0^t \{ (x_u^t, y_u^t) - (x_1^0, y_1^0) \} + (x_1^t, y_1^t)$$

$$A_1 = W_1^t \{ (x_u^t, y_u^t) - (x_1^1, y_1^1) \} + (x_1^t, y_1^t)$$

$$W_0^t = 1/|a|^2 \cdot \begin{pmatrix} a_x & a_y \\ -a_y & a_x \end{pmatrix} \begin{pmatrix} b_x & -b_y \\ b_y & b_x \end{pmatrix}$$

$$W_1^t = 1/|c|^2 \cdot \begin{pmatrix} c_x & c_y \\ -c_y & c_x \end{pmatrix} \begin{pmatrix} b_x & -b_y \\ b_y & b_x \end{pmatrix}$$

$$a_x = x_1^0 - x_0^0, a_y = y_1^0 - y_0^0$$

$$b_x = x_1^t - x_0^t, b_y = y_1^t - y_0^t$$

$$c_x = x_1^1 - x_0^1, c_y = y_1^1 - y_0^1$$

なお、式 (2) におけるW₀^tは、bベクトルに対するaベクトルの回転を求め、それを縮小している。W₁^tは、cベクトルをbベクトルに対し同様に求めている。また、A₀はW₀^tよりKF₀上の点を回転し、それに座標(x₁^t, y₁^t)を加えることにより移動を計算している。A₁も同様である。そして、最後に時間のパラメータ t を変化させて、中間画像の各点P(x_u^t, y_u^t)を求めることができる。

このアルゴリズムは6個の座標データより1個の中間画像の座標を求める手法であり、2つのキーフレームが時間的に接近しているときに、効果的であるが、そうでないときは中間画像に歪を発生させてしまう。

3.2 修正版アルゴリズム

Miura アルゴリズムではキーフレームの内容によっては対象がひずむ場合がある。そこで、以下のような修正版アルゴリズムを提案する。

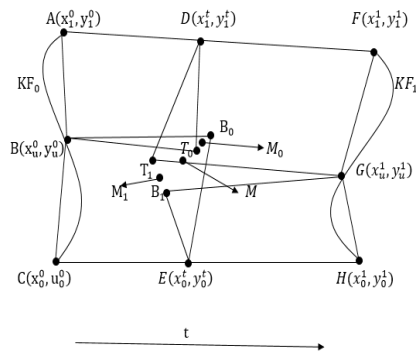


図5. 修正版アルゴリズム³⁾

図5における端点はD,Eは、Miura アルゴリズムの式 (1) により近似を行いその他の点については以下のように計算する。

最初にKF₀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T₀(x_{T0}, y_{T0})とする。また、CB、CEを二辺とし同様にB₀(x_{B0}, y_{B0})を求める。この2点を用いて次式よりM₀(x_{M0}, y_{M0})を求める。

$$x_{M0} = (1 - K_0)x_{T0} + K_0x_{B0}$$

$$y_{M0} = (1 - K_0)y_{T0} + K_0y_{B0} \quad (3)$$

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

次にKF₁基準とし、同様にDF、FGよりT₁(x_{T1}, y_{T1})を、GH、HEよりB₁(x_{B1}, y_{B1})を求め、式 (4) より、M₁(x_{M1}, y_{M1})を求める。

$$x_{M1} = (1 - K_1)x_{T1} + K_1x_{B1}$$

$$y_{M1} = (1 - K_1)y_{T1} + K_1y_{B1} \quad (4)$$

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

式 (3)、(4) より求めたM₀、M₁より中間画像の座標M(x_M, y_M)を式 (5) より求める。

$$x_M = (1 - L)x_{M0} + Lx_{M1}$$

$$y_M = (1 - L)y_{M0} + Ly_{M1} \quad (5)$$

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

式 (5) 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、KF₀とKF₁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、KF₀自身に位置パラメータと回転パラメータを入力することにより並進及び回転を施し、最後の画像

KF₁に幅を持たす工夫をした。

4. シミュレーション

前章の修正版アルゴリズムを用い、まろうど酒造「ちほまろ」のシミュレーションを行った。また、観光キャラクターの試作を行うにあたって、基本的な人の動き（腕、足の動き）についてもシミュレーションを行った。以下、Java (Windows 版) を用いて、コンピュータ (Dell Computer) 上で実行した結果の例を図5-図7に示す。

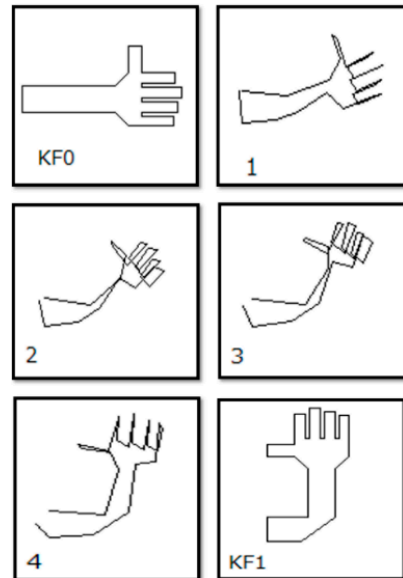


図5. 腕のシミュレーション



図6. 足のシミュレーション

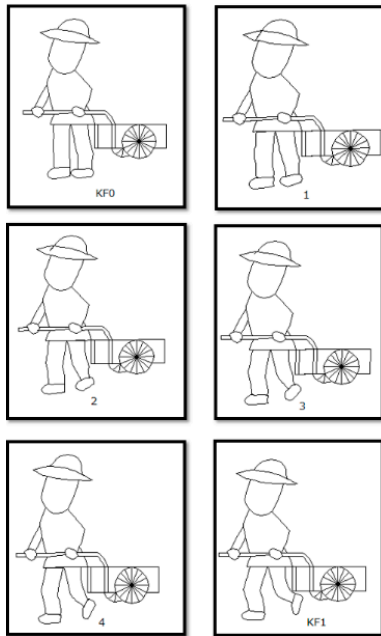


図7. 「ちほまろ」のキャラクター

5. 考察

まず画像データについては、今回方眼用紙で座標を確認しながら作成する原始的な方法を採用したため、画像データ作成に時間を要してしまった。一般的に、アニメーション作業においてキーフレーム作成は経験を必要とし、一番時間がかかるものであるが、デジタイザーなどの画像入力装置を利用する方が効率的であると考えられる。また、キャラクターの試作には、人の基本的な動きを考える必要があるため、こちらについても画像データを作成した。

また、Miura アルゴリズムは処理時間が速いが、入力データが不連続な場合には画像に歪みを発生させ、画質を低下させる。それに対して、修正版アルゴリズムでは、比較的なめらかな画像を生成できたと考えられる。Miura アルゴリズムでは、6個の位置座標から中間画像の1個の座標を求め、連続性の弱いキーフレームの影響を直接受けるという欠点があるのに対して、修正版アルゴリズムではその影響を回避できるように工夫している。

さらに、アニメーションが画像データに依存するか否かも調べてみた。例えば、腕のアニメーションでは節点数が27個に対し、足のアニメーションは101個である。節点数が多いほうが中間画像のなめらかさがあることが確認できた。また、画像データの幾何学的性質をいろいろ変えてみた。その結果アニメー

ションの様々な幾何学的変化に対応して自然な中間画像を生成できることも確認できた。

キャラクターの「ちほまろ」については、動きに変化のない部分が中間画像に変化をもたらすことができなかった。入力データが直線近似でなくなめらかな場合や、2つのキーフレームが極端に関連性の薄い場合には、新しいアルゴリズムでもうまく生成できなかった。そのため、たくさんの方々で異質な画像データを入力して、詳細な分析を必要とするであろう。また、表情の変化などについても今回作成できなかったため、そこについても今後の課題としていきたい。

6. おわりに

本研究では、コンピュータアニメーションにおける観光キャラクターの中間画像および、基本的な人の動きの中間画像を作成した。その結果、なめらかで自然な中間画像を作成することができたが、入力データが直線近似でなくなめらかな場合や、2つのキーフレームが極端に関連性の薄い場合には、修正版アルゴリズムでもうまく生成できなかった。そのため、たくさんの方々で異質な画像データを入力して、詳細な分析をすることが必要とされる。また、表情の変化などについても今回作成できなかったため今後の課題とし、さらに3次元コンピュータグラフィックスやバーチャル技術を念頭に置いて3次元キャラクターへの中割アルゴリズムの応用も検討していきたい。

参考文献

- 1) 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).
- 2) Reeves, W.T.: “Inbetweening for Computer Animation Utilizing Moving Point Constraints”, *Computer Graphics*, Vol. 15, No. 3, pp. 263-269 (1981).
- 3) 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(2006).
- 4) JATA 日本観光協会国内旅行の現状と課題. [Online] https://www.jata-net.or.jp/membership/info-japan/research/03_1st.html
- 5) 高千穂ムラたひ. [Online] <http://takachiho-muratabi.com/osake/>