

## 立体キャンバスを用いた2次元アニメーション制作支援手法

藤田正樹<sup>†</sup> 齋藤豪<sup>††</sup><sup>†</sup>東京工業大学 工学部 <sup>††</sup>東京工業大学 情報理工学院

## 1 はじめに

近年のアニメーション制作では、求められる画面の情報量の増大と業界の人手不足が言われている。2.5D cartoon model[1] や BetweenIT[2] ではこれらの作業量の削減ができる可能性があるが、これらは作画と補間をそれぞれ独立したものと考えている。

矢田らの手法 [3] では原画作業の流れを変え、補間対象である原画を描く手がかりに計算機を使用することを提案している。本手法ではさらに発展し、原画を描くと同時に作成される補間結果を対話的に確認しながら原画の修正が出来ることを目指す。

## 2 提案手法

## 2.1 入力

ユーザーが行う入力は、対象物を回転平行移動させて描いた2枚の線画と2.2節で述べる対象物に対応する立体キャンバスのカメラに対しての姿勢と運動、そしてその線画間の線同士での対応付けである。投影法、画角等のカメラパラメータは既知とする。補間結果は描画に応じて逐次生成されるので、ユーザはその場で描き、結果を見ながら必要に応じて修正を加えることが出来る。

## 2.2 立体キャンバス

本手法ではユーザが描画平面に対して描画した線は、その奥に配置された立体曲面の表面へと投影され保存される(図1左)。この曲面を立体キャンバスと呼称する。立体キャンバスの移動回転によって、その表面に保存された線に立体的な動きが与えられる(図1右)。

## 2.3 立体キャンバスの形状更新

立体キャンバスの形状は、ある初期形状を与えた上で、1つの表面に投影した複数の線画が誤差が少なく一致するように随時更新することで決定される。

3次元で表されるキャンバス座標系を用意し、その空間で  $z = f(x, y)$  とする立体曲面でキャンバスの形状を定義する。

2枚の入力線画  $I_0, I_1$  が描かれた際のキャンバス座標系でのカメラの位置を  $v_0, v_1$  とする。  $I_0, I_1$  間での

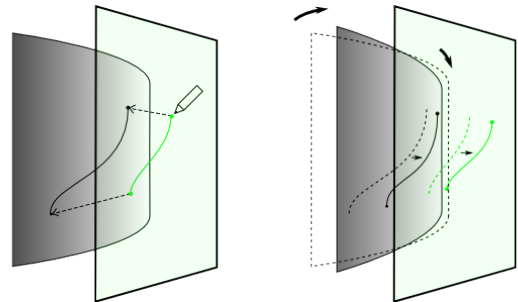


図1: 立体キャンバスと描画平面の関係

となる線の組  $C_0, C_1$  上の点を  $\{c_{ni}\}(n \in \{1, 0\}, i = 0, 1, \dots, m)$  によって表す。  $c_{ni}$  はキャンバス座標系上でカメラの正面に配置された描画平面上の座標である。このとき、カメラの位置  $v_n$  からの  $c_{ni}$  の投影点は直線  $l_{ni}(t) = v_n + t(c_{ni} - v_n)$  上にある。  $l_{0i}(t)$  上で  $l_{1i}(t)$  に最も接近する点  $d_{0i}(v_0, v_1, c_{0i}, c_{1i})$  および  $l_{1i}(t)$  上で  $l_{0i}(t)$  に最も接近する点  $d_{1i}(v_0, v_1, c_{0i}, c_{1i})$  を計算する。

新しい立体キャンバスの形状はこれらの点が可能な限り表面に乗るように、次の誤差  $e$  が最小になるように決定する。

$$e = \sum_{n=0,1} \sum_{C_n \in I_n} \sum_{c_{ni} \in C_n} |z_{ni} - f(x_{ni}, y_{ni})|^2 \quad (1)$$

ただし、  $(x_{ni}, y_{ni}, z_{ni})$  は  $d_{ni}$  の座標要素である。

ここで立体キャンバスの形状が更新されても、元々の入力線画においてユーザが入力した形状を線画が保っていることが必要である。そのため、線は自身が描かれた際の形状を保持しており、形状更新後は改めてキャンバスの表面に投影される。

## 2.4 画像出力

立体キャンバスの姿勢は、最初に入力された立体キャンバスの運動の軌跡  $P(t)$  に沿ってキャンバスを動かすことで得られる。

任意の  $t(0 \leq t \leq 1)$  に対してキャンバス上に表示される画像を求めるにあたって、キャンバス形状の変形のみで線画同士を完全に一致させることは困難なため、  $C_0, C_1$  を立体キャンバス上で線形補間した線  $C(t)$  を最終的に出力する。

computer assisted 2d animation with cubic canvas

<sup>†</sup> Masaki FUJITA<sup>††</sup> Suguru SAITOFaculty of Engineering, Tokyo Institute of Technology (<sup>†</sup>)  
School of Computing, Tokyo Institute of Technology (<sup>††</sup>)

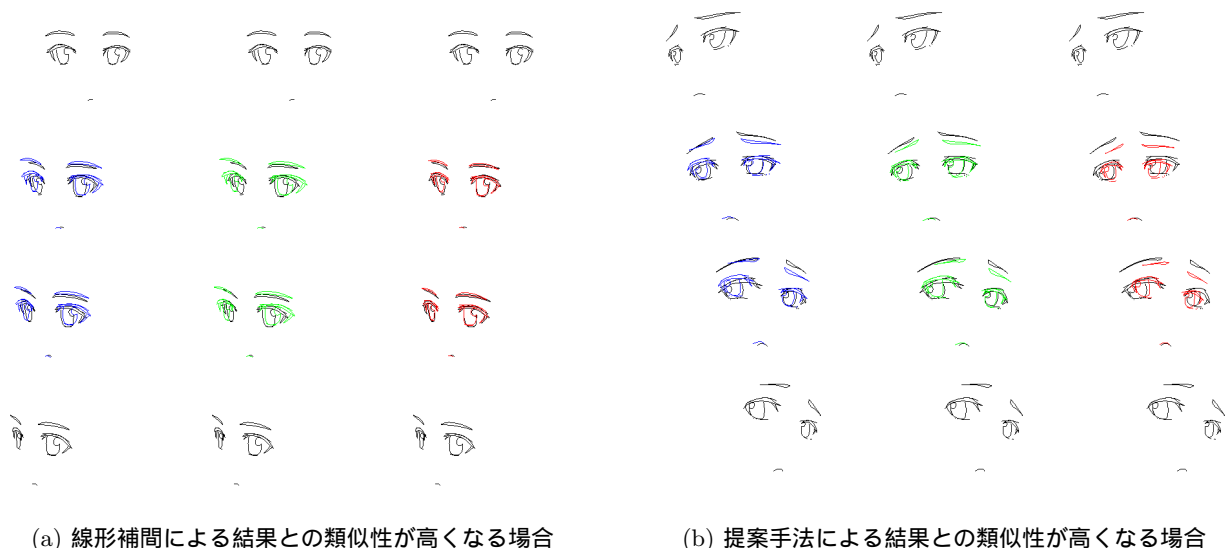


図 2: 実験結果 黒：実際の線画，左・青：提案手法 中央・緑：提案手法（平面形状） 右・赤：線形補間

$$C(t) = \{c_i | c_i = (1-t)c'_{0i} + tc'_{1i}\} \quad (2)$$

$c'_{ni}$  は変形後のキャンパス表面に投影された  $c_{ni}$  である。

### 3 評価実験

ここでは対話性は考慮せず、固定の線画に対して本手法による補間法及び単純な画面上での線形補間法と、矢田らを用いた空間中の平面の移動回転による補間法との比較評価を行う。矢田らの手法は本手法において形状更新を行わず立体キャンバスとして平面を用いることで実装した。

TVで放送されたアニメーション内のキャラクターの顔の動作を図2黒線のようにフレーム毎にトレースし線画化する。得られた一連の線画について、開始及び最終フレームから各々の手法で補間を行い、結果から中間フレームに最も近いものを選択する（図2色つき線）。その際用いる線画間の類似性計算には、対応付けられた各線について十分大きな数でサンプリングした座標同士のピクセル間距離の平均を用いた。

キャンバス形状の最適化には最小二乗法を用いた。

立体キャンバスの形状を表す式には式(3)を用い、式(1)を最小化するように  $p_i (i = 0 \dots 5)$  を決定した。

$$f(x, y) = p_0 + p_1x + p_2x^2 + p_3y + p_4y^2 + p_5y^3 \quad (3)$$

実験の結果、図2(a)のように線形補間に近い作画が中割りにおいて行われているカットでは画面上の線形補間法による結果の類似性が高く、特に回転移動の変化量が少ない場合においてその傾向が強くみられた。

しかし一方で、図2(b)のようにキャラクターが大きな動きをするカットでは立体感を考慮した作画がなされており、そのようなケースではより立体的な補間を生成する本手法の結果の類似性が高かった。

また、本手法のキャンバスの形状更新を矢田らの常に平面を用いる手法と比較すると、図2(b)のようにカメラがキャンバスに正対している状態に近いときはあまり差異はみられなかったが、図2(a)のようにカメラがキャンバスに対して斜めに向かう場合大きなずれが生じ、キャンバス形状更新の効果が認められた。

### 4 まとめと今後の課題

立体キャンバスを用いて、2枚の線画から回転平行移動によるアニメーションの補間を行う手法を提案し評価実験を行った。

今後の課題として、本手法では対応できない、物体の形状によって生まれる輪郭線への対応、キャンバス形状の更新のみならず対象物の移動回転等の自動推定が挙げられる。

#### 参考文献

- [1] Alec Rivers, Takeo Igarashi, and Frédo Durand. 2.5 d cartoon models. In *ACM Transactions on Graphics (TOG)*, Vol. 29, p. 59. ACM, 2010.
- [2] Brian Whited, Gioacchino Noris, Maryann Simons, Robert W Sumner, Markus Gross, and Jarek Rossignac. Betweenit: An interactive tool for tight inbetweening. In *Computer Graphics Forum*, Vol. 29, pp. 605–614. Wiley Online Library, 2010.
- [3] 矢田和沙, 齋藤豪ほか. 首の運動に伴う顔の回転アニメーションの描画補助手法に関する研究. 第77回全国大会講演論文集, Vol. 2015, No. 1, pp. 65–67, 2015.