

移動最小二乗法を用いたメッシュ変形による 3D キャラクタの誇張表現 Exaggerated 3D Character Representation via Mesh Deformation Using the Moving Least Squares Method

大室 絢葉[†] 秋元 優希[†] 佐藤 周平^{†‡}
Ayaha Omuro Yuki Akimoto Syuhei Sato

1. はじめに

コンピュータグラフィックス(以下, CG)技術の発達により, アニメ制作においてキャラクターや背景などをCGで作成することで, 制作過程の効率化を図ることが多くなっている. ただし, 現実に則して作成されたCGでは, シーンによっては迫力が乏しいなど, 十分な表現ができない場合がある. そのため, アニメ制作の現場では, キャラクタの手や足を現実ではありえない形に変形し, シーンに迫力を与えるような, いわゆる誇張表現を手作業で行う. ただし手作業の場合, 多くの頂点を動かし目的の形に変形を行わなければならない, 手間がかかる.

そこで我々は, 原画や漫画等の画像を入力とし, それに合うように3Dのキャラクターモデルを変形することで, 誇張表現を自動で行う方法について研究している. そのために, キャラクタの手などのパーツを描画したマスク画像と入力画像を用い, パーツのマスク画像が入力となる画像に近づくように変形する. そしてその変形情報を基にケージベースで3Dモデルを変形するというアプローチにより, 誇張表現されたモデルが得られるかを実験している. 画像の変形には, 移動最小二乗法 (MLS: Moving Least Squares) に基づく変形手法を用いる. 最適化が収束した後, 画像上で得られた変形に基づく制御点の移動量を 3次元の格子空間に適用し, 3Dモデルをケージベースで変形する. このアプローチについて実験を行うため, まず本稿では, 人型の3Dキャラクターの手のモデルに対して, 手動で変形した3Dモデルのレンダリング画像を目標とし, 最適な変形が実現できるかを検証したので, その結果を報告する.

2. 提案手法

本手法の概要を述べる. まず, 変形したいパーツが含まれる画像を入力する. また, 3Dモデルにおいて変形したいパーツの部分のみを抽出したマスク画像を作成する. 本稿では, 実験として手動で変形した3Dモデルをレンダリングした画像に対し, パーツ部分を白, 他を黒としたマスク画像を作成し目標画像とした. 同様に, 変形前の3Dモデルについても, 対象パーツを白, 他を黒としたマスク画像を作成する.

次に, 両画像上に, 3Dモデルを変形させるために3次元の格子上に定義される制御点と格子点を, 画像上の対応位置に座標変換して配置する. そして, 目標となる入力画像に近づけるように制御点を移動させ, マスク画像を,

MLSに基づく手法により変形する. 画像変形の結果は, 変形後のマスク画像と目標画像の論理積/論理和を用いて定量的に評価し, 最急降下法により制御点の移動位置を最適化する.

本稿では, 3Dモデルの変形に, ケージベースのメッシュ変形手法を用いるため, 画像上で得られた制御点の最適な位置を3D空間上に対応付け, 制御点の移動位置をもとに, MLSに基づいてケージを変形させ, 対象の3Dモデルを目標に合うよう変形する. 以下に, 各処理の詳細を述べる.

2.1 3Dモデルの変形

本研究では, 3Dモデルの変形に free-form deformation[1] を利用する. これは, 3Dモデルを任意の分割数の直方体(格子)で囲い, その格子の変形に追従するよう3Dモデルを対応付けることで, 複雑なモデルでも簡単に変形できる方法である.

本手法では, 変形対象のモデルのパーツを x, y, z 方向にそれぞれ10分割された立方体で囲う. このとき, 立方体の中心座標をパーツの中心座標に一致するよう配置する. そして, あらかじめ与えられた制御点の移動量に従って, 立方体の各頂点を移動させることで, 対象のパーツを変形させる. なお, 本実験では, 3Dモデルをマスク画像としてレンダリングする際のカメラ視線に対して, ケージの1面が正対するように設置している. このため, 視線に対する奥行き方向の変形はすべて等しくなると仮定し, 画像上で得られた変形をスケールのみ合わせて3D空間に反映する.

2.2 移動最小二乗法を用いた格子変形

本研究では, 画像と3Dモデルの変形を格子ベースで行い, この格子の変形には, MLSに基づく手法[2]を用いる. MLSでは, ユーザーが動かすハンドルである制御点の元の位置 p_i とそれに対応する移動後の点 q_i の対応関係から, 格子点 v の移動先 $f(v)$ を求める. このとき, 格子点と各制御点との距離に基づいた重み $w_i = 1/|x - p_i|^{2\alpha}$ (α はパラメータ)を用いて, 重み付き最小二乗誤差が最小となるような変換を求める. 変換の種類は, 剛体変換, 類似変換, アフィン変換から選択できるが, 本研究では自由度の高い変形を可能とするため, アフィン変換を採用する. この手法により, 少数の制御点を動かすだけで, 格子全体にスムーズな変形を与えることが可能となる.

本手法では, 3Dモデルの変形に用いる直方体ケージを, 奥行き方向を無視した2次元平面として画像上に投影し,

[†]法政大学 Hosei University

[‡]プロメテック CG リサーチ Prometech CG research

画像上の制御点の移動位置を基に、MLS を用いて画像および 3D モデルを変形させる。このように、MLS により制御点の移動に応じた格子の変形が可能となるが、最終的な目標は、変形後のマスク画像が、手動で変形された目標画像と可能な限り一致するように、制御点の移動位置を自動的に求めることである。そのため次節では、この制御点の移動位置を最適化する手法について述べる。

2.3 最急降下法による制御点移動位置の最適化

制御点の移動位置は、マスク画像の変形後の形状と、目標画像との差が最小となるように最適化する。そのために、本研究では、変形後のマスク画像 I_{def} と目標画像 I_{tar} との間で、ピクセル値の論理積および論理和を計算し、それらを用いて類似度を計算する。そしてこの類似度を用いた目的関数 E を以下のように定義し、これを最小化するような制御点位置を求める。

$$E = \left| 1 - \frac{I_{tar} \cap I_{def}}{I_{tar} \cup I_{def}} \right|$$

ここで、 $I_{tar} \cap I_{def}$ は双方の画像ともピクセルが白いところでは 1、それ以外は 0 をすべてのピクセルで合計した値である。また、 $I_{tar} \cup I_{def}$ はいずれかの画像においてピクセルが白いところでは 1、それ以外は 0 をすべてのピクセルで合計した値である。

制御点の移動位置 q の最適化には、最急降下法に基づく手法を用いる。初期状態では、各制御点の移動先 q_i は元の制御点位置 p_i と同じ座標に設定されている。各制御点に対して、 x 方向の差分 dq_x^i だけ動かしたときの評価値 E_x^i 、 z 方向の差分 dq_z^i だけ動かしたときの評価値 E_z^i をそれぞれ計算し、元の評価値との差分で勾配を算出する。

$$q_x^{i+1} = q_x^i - \alpha \frac{E_x^i - E_x^{i-1}}{dq_x^i}$$

$$q_z^{i+1} = q_z^i - \alpha \frac{E_z^i - E_z^{i-1}}{dq_z^i}$$

ここで、 α は学習率である。この勾配に基づいて新たな制御点位置を求め、それによる変形結果の評価値 E_x^{i+1}, E_z^{i+1} が更新前のものより改善されている場合にのみ、該当制御点の座標を更新する。この処理を全制御点に対して行う。この最適化処理は、評価値があらかじめ設定した閾値以下となるか、最大反復回数に達するまで繰り返される。

3. 実験

提案手法を適用して制御点の移動位置を画像上で最適化した結果を図 1 に示す。マスク画像の描画は Blender[3] を用いて行い、VRoid Studio[4] の人型 3D モデルの左手を変形対象とした。左列は画像変形で最適化を行った結果、右列が 3D モデルの様子である。それぞれ、(a) が元の状態、(b) が目標の状態、(c) が結果である。本実験では、手の周囲を囲うように 2 次元平面上に 4 点の制御点を配置し、上部 2 点を移動させることで、縦方向に引き伸ばされた形状を目標とした。

結果より、提案手法による変形は、縦方向については目標画像に近い形状を実現できていることがわかる。しかし、横方向の変形はあまり目標とは合っていない。この原因と

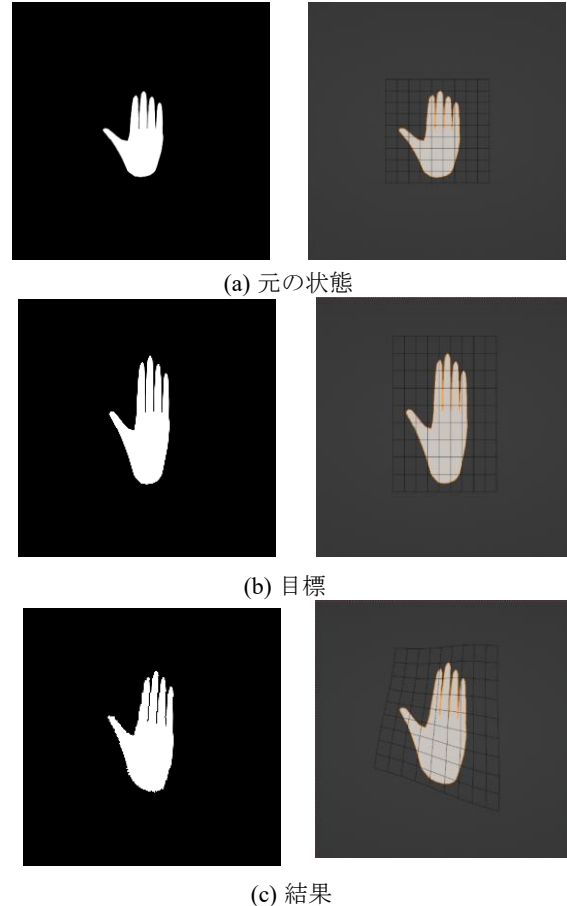


図 1: 提案手法による実験結果。左がマスク画像、右が 3D モデルをレンダリングした画像である。

して、自由度の高い変形を可能にすることを目的としてアフィン変換を利用したため、意図しない部分での変形が大きくなったことが影響していると考えられる。

4. おわりに

本稿では、アニメ等の制作に利用される誇張表現を自動で行うことを目的として、入力した画像に合うよう、移動最小二乗法を用いた変形手法[2]を用いて、画像上で、適切な制御点の移動が得られるかを検証した。実験結果から、縦方向の変形に関しては目標画像に近い結果が得られた。一方で、横方向の変形には課題が残った。この課題に対しては、今後アフィン変換以外の変形手法を導入し、実験・検証をする予定である。

参考文献

- [1] T. W. Sederberg, S. R. Parry, "Free-form deformation of solid geometric models," in Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques, pp.151–160, 1986.
- [2] Schaefer S., McPhail T., Warren J., "Image Deformation Using Moving Least Squares," *ACM Transactions on Graphics*, Vol. 25, No. 3, pp. 533–540, 2006.
- [3] Blender, <https://www.blender.org/>.
- [4] VRoid Studio, <https://vroid.com/studio>.