

洋服シワに対するキャラクターイラスト特有の影形状のリアルタイムレンダリング

青木 明優花¹ 阿部 雅樹² 渡辺 大地²

概要: 近年, コンピュータを用いてイラストを描く人が増加しており, ツールの機能向上と描画手法の発展からクオリティの高いイラストを仕上げることができるようになった. イラストのクオリティを上げる重要なポイントはいくつかあるが, そのポイントの1つに陰影がある. 陰影付けの作業には慣れが必要であり, 仮に作業に慣れたとしても作業時間や負荷が大きいという問題がある. キャラクターの陰影付けでは, 様々なパーツに対して陰影付けを行うが, その中でも洋服は大部分を占め重要である. さらに洋服の陰影付けではシワの影響が大きい. イラストでは, 物理法則に従わない陰影の付け方をすることがあり, 影の形状はイラスト制作者によっても異なる. そこで本研究ではイラストへの陰影付けの作業時間や負荷の軽減を目的とし, 洋服シワに対してできるイラスト特有の影の生成手法を提案する. まず入力したシワの情報から陰影計算で使用する法線マップや距離マップを作成する. そのマップを参照し, 3D モデルを使用してシェーダで陰影となる部分の計算を行う. 陰影計算の工程では, 分類したシワ線の種類によって, アルゴリズムを変更する. これらの手法を実装し, 洋服シワに対するイラスト特有の陰影付けを実現した.

1. はじめに

近年, コンピュータを用いてイラストを描く人が増加しており, ツールの機能向上と描画手法の発展からクオリティの高いイラストを仕上げるできるようになった. イラストのクオリティを上げる重要なポイントはいくつかあるが, そのポイントの1つに陰影がある. イラストでは陰影の塗り方によって全体の印象が全く違うため, イラストにおいて陰影は重要である.

陰影付けの作業には慣れが必要であり, 仮に作業に慣れたとしても作業時間や負荷が大きいという問題がある. また, 陰影の形状やイラスト全体のバランスによって描き直しを必要とする反復作業も多い.

グラフィックソフトの ClipStudioPaint[1] や PhotoShop[2] には陰影色の自動調整機能や, デッサン人形を表示する補助機能があるが, 陰影付けを行う機能はない. そこで, 陰影付けを自動で行うことができれば, 作業時間や負荷を減らすことが期待できる.

キャラクターの陰影付けでは, 様々なパーツに対して陰影付けを行うが, その中でも洋服はキャラクターの大部分を占めている. 洋服の陰影付けでは, シワによって陰影の

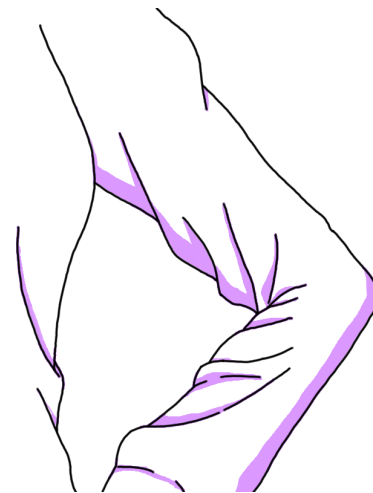


図 1 洋服のイラスト

付く場所が大きく変わる. 図 1 は洋服のイラストである. 洋服に対してシワを示す線が描かれているが, その線に対して陰影がついている. このように, 洋服に対して陰影を付ける場合には, シワを考慮して陰影付けを行う必要がある.

またイラストでは, 物理法則に従わない陰影の付け方をすることがある. 実際には薄い影でもイラストでは立体感を表すために, はっきりと陰影を書き入れることがある. また, シワの形状をわかりやすくするために, 実際には陰影はできない部分にも陰影を書き入れることがある. さ

¹ 東京工科大学大学院バイオ・情報メディア研究科
東京都八王子市片倉町 1404-1

² 東京工科大学メディア学部
東京都八王子市片倉町 1404-1

にシワの形状はイラスト制作者によって異なる。イラストに陰影を付ける時にはこうしたことも考慮する必要がある。

先行研究では、陰影を生成する研究 [3] がある。この研究では線画といくつかの入力から、レリーフ彫刻のようなモデルを作成している。そしてそのモデルを使用して、陰影の計算を行う。しかしこの手法では輪郭を考慮しているため、内部にある洋服のシワに対して陰影をつけるのは難しい。

同じようにモデルを生成するスケッチベースのモデリング手法 [4][5][6] があるが、3D モデルを生成することを目的としているため、シェーディングを行い、陰影を生成するのは難しい。

他に、イラストの内部にある洋服のシワに注目した研究 [7] がある。この研究では、ユーザーが描いたシワがどのような形状を示しているのかを分析し、そこからイラスト風のシェーディングができるモデルを生成している。しかし、この研究ではシワの解析を目的としているため、イラスト特有の陰影や、イラスト作成者が望むような陰影を生成するには、この手法は向いていない。

本研究では、イラストへの陰影付けの作業時間や負荷の軽減を目的とし、キャラクターイラストの洋服のシワに対して、イラスト特有の陰影を自動で生成することを目指す。また、イラストでは様々な種類の塗り方があるが、本研究ではアニメ塗りやそれをベースとした塗り方を対象とする。イラストでは三次元形状を必要とするような厳密な陰影ではなく、厳密でない陰影を望むこともある。先行研究のような正確な空間情報や 3D モデルから計算することはできない。本研究では、後述するシワの特性を制作者が自由に設定できることによって、作成者が望む陰影の生成を可能とする。また、光源の位置が違えば陰影の付く位置が変わるため、本研究では光源の位置も考慮して陰影の生成を行う。

筆者らは、先行研究 [8] において本研究の基本的な方針の発表を行った。本論文においては、先行研究に対する新規点となる有理ベジェ曲線を用いたシワ形状の柔軟な制御手法の追加と、異なる種類のシワを同時に用いる手法について詳しく述べる。

2. 洋服シワと陰影について

2.1 イラストにおける洋服シワ

イラストにおいてシワは、見栄え良く見えるように記号化した線で表現している。図 2 はシワを描いたイラストである。図 3 は図 2 に陰影を付けたものである。紫色の部分は光源が上の時に、陰影が付く部分を示している。

このように陰影の付き方が違う理由は、シワが示している意味が違うためである。本研究ではシワが示している意味を大きく 2 種類に分けた。

図 4 は、上が布を横から見たときの模式図である。下が

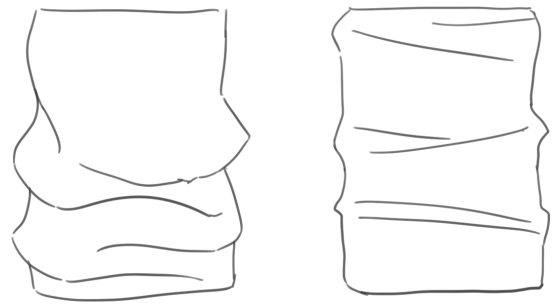


図 2 洋服シワのイラスト

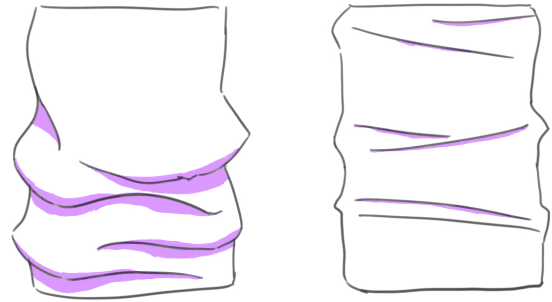


図 3 陰を付けたイラスト

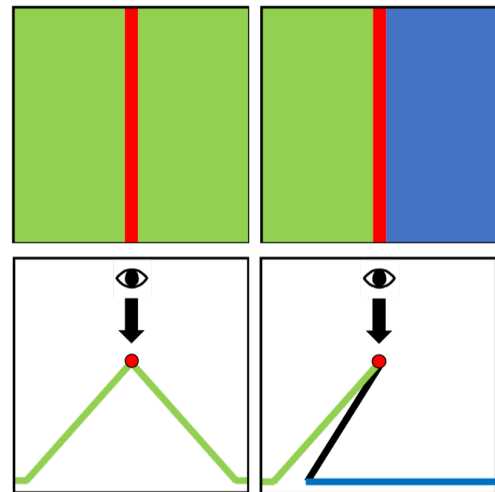


図 4 模式図

布を上から見たときの模式図である。赤の点がある部分がシワの線が描かれている部分を示し、緑や青の部分が布の部分である。

実際にイラストを見る時に見える部分は赤色の部分となる。左の図では見える部分のシワ線と布が連続に続いているが、右の図では見える部分のシワ線と布が連続になっていない。このようにシワ線が左の図のような連続しているシワと、右の図のような連続していないシワの 2 つに分類する。連続しているシワを「稜線シワ」、連続していないシワを「輪郭シワ」と本研究では呼称する。

2.2 イラストにおけるシワ線に対する陰影

イラストにおいて陰影は物理法則に従わない付け方をすることがある。例えば、実際には光が当たる部分で陰影は

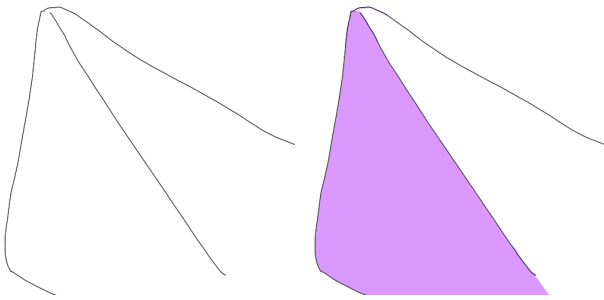


図 5 稜線シワの特徴 1

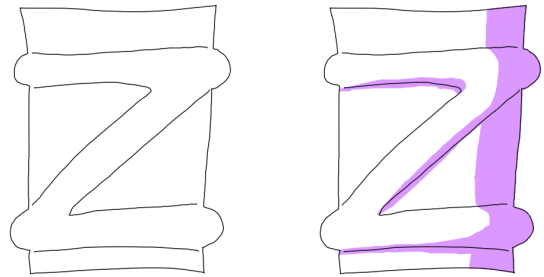


図 7 稜線シワの特徴 2

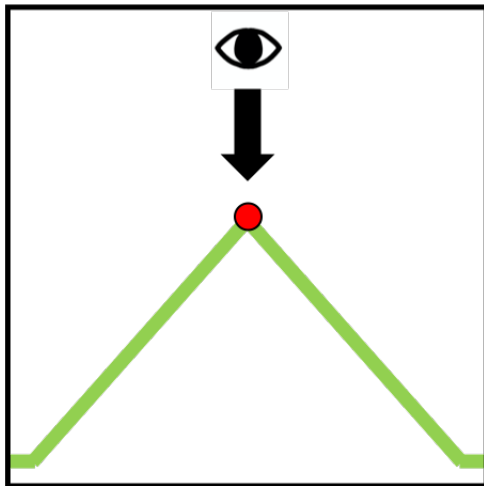


図 6 断面図 1

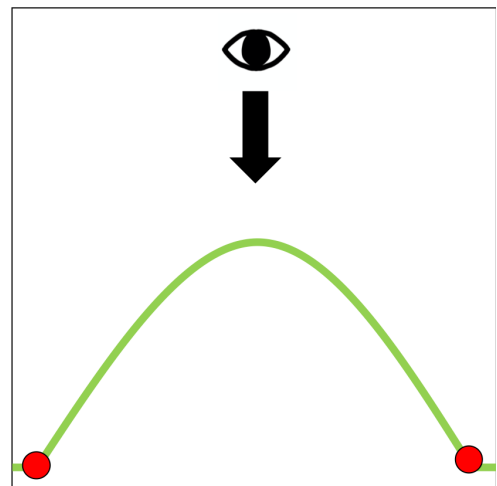


図 8 断面図 2

できないが、シワの形状を示す為に陰影を付けることがある。また、薄い影でもイラストでは立体感を表すために、濃い陰影にすることもある。

イラストにおいて陰影は、イラスト制作者によって異なる。よってイラストにおける陰影は必ずこの形状になるというものはないが、いくつかの特徴や傾向はある。洋服のシワ線に対して陰影は、シワ線に接した陰影を描いていることが多い。通常のモデルでレンダリングを行う場合は、シワ線に接している陰影にならず、離れてしまうことがある。そのため、モデルを使用した通常のレンダリングではイラストの陰影を生成することはできない。

2.3 洋服シワと陰影の形状

2.1 節では稜線シワと輪郭シワに洋服のシワ線を分類した。本節では、稜線シワと輪郭シワが、イラストではどのようなシワ線の形状をし、どのような陰影を付けるのかについて、いくつかの特徴を示す。

2.3.1 稜線シワ

図 5 にシワの形状と陰影の形状の例を示す。稜線シワでは、図 5 の左のような形状のシワがある。これは布の山の頂点の部分にシワ線を示している。右に光源がある場合は図 5 の右のイラストのように大きく陰影がつく特徴がある。断面図を図 6 に示す。赤がシワ線で、緑が布である。

次に 2 本の稜線シワによって表される形状のシワを示

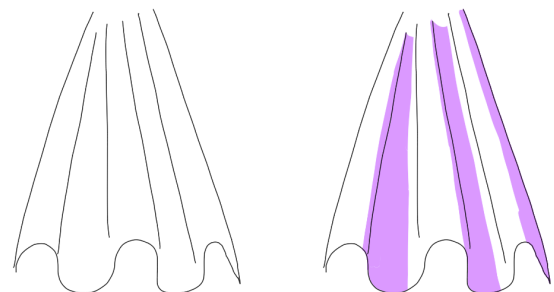


図 9 稜線シワの特徴 3

す。図 7 は、2 本の稜線シワによって表されるシワの形状と陰影の形状の例である。2 本の稜線シワによってできるシワ線には、図 7 の左のような形状のシワがある。これは 2 本のシワ線の間が山のような形状を意味している。左に光源がある場合は図 7 の右のイラストのように陰影がつく特徴がある。断面図を図 8 に示す。赤がシワ線で、緑が布である。

次に、図 9 は 2 本の稜線シワによって表されるシワの形状と陰影の形状の例の 2 つ目である。2 本の稜線シワによってできるシワ線には、図 9 の左のような形状のシワがある。これは 2 本のシワ線の間が谷のような形状を意味している。左に光源がある場合は図 9 の右のイラストのように陰影がつく特徴がある。断面図を図 10 に示す。赤がシワ線で、緑が布である。

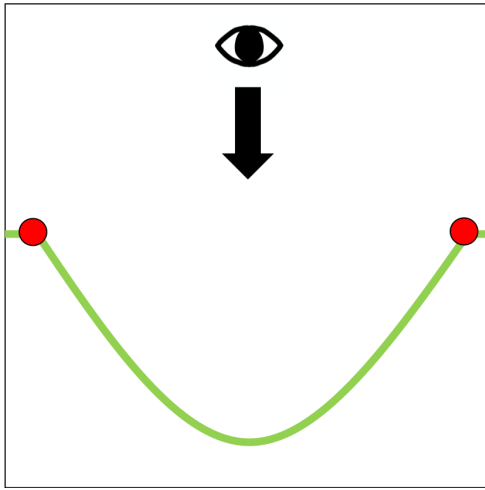


図 10 断面図 3

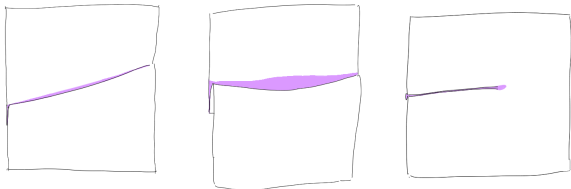


図 11 輪郭シワの特徴

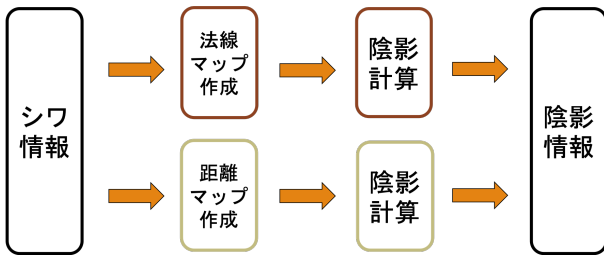


図 12 手法

2.3.2 輪郭シワ

輪郭シワでは、線の片側に影が付くという特徴がある。線の両側に影が付くことはあまりない。図 11 の左のイラストのように同じ幅で陰影が付く場合や、図 11 の中央のイラストのように真ん中の部分が幅広く影が付く場合もある。また、2つの輪郭シワが組み合わさると図 11 の右のイラストのような影ができるという特徴がある。

3. 提案手法

提案手法は、稜線シワか輪郭シワによって、陰影計算でのアルゴリズムを変更することで、陰影を生成する。まずシワ情報を入力する。そこから法線マップと距離マップを作成する。そして陰影計算では、円柱の 3D モデルを使用し、シェーダーで作成したマップを使用して陰影計算を行う。稜線シワは法線マップを使用し、輪郭シワは距離マップを使用して計算を行う。最後に計算結果の陰影の情報を組み合わせて陰影を生成する。手法の流れを図 12 に示す。

3.1 シワの情報

シワの情報は任意の制御点をデータとして持つ曲線データである。本研究では、イラスト作成者が直感的に入力しやすいように、Catmull-Rom Spline 曲線 [9] のアルゴリズムを使用する。

3.2 マップの作成

3.2.1 曲線上近傍点抽出

法線マップや距離マップの作成には、任意点から曲線に対する最近傍点を求める必要がある。法線マップでは複数のシワの曲線すべてに対しての最近傍点を求め、距離マップではシワの曲線ごとに最近傍点を求める。スプライン曲線に対する最近傍点を抽出する方法は、様々な手法が提案されているが、本研究では高い精度は必要としないことや、マップ生成時における処理速度はそれほど高速でなくとも問題がないことから、適当な分割数によるサンプリング探索によって最近傍点を抽出した。以降、マップ値を算出する対象のピクセルを p 、 p に対応した曲線上の最近傍点を C_p 、それぞれの位置ベクトルを \mathbf{P} 、 C_p と表記する。

3.2.2 法線マップ

まず稜線シワのアルゴリズムで使用する法線マップを作成する。法線マップとは面の法線ベクトルを保存したテクスチャ画像のことで、入力されたシワ情報を元に作成する。

まず p に対する C_p を求めた後、両点間の距離 $d = |\mathbf{P} - C_p|$ と単位ベクトル $\mathbf{J} = \frac{\mathbf{P} - C_p}{d}$ を求める。次に、有理ベジェ曲線によって入力したシワ線上にできる形状を設定する。有理ベジェ曲線の制御点を \mathbf{Q}_i 、各制御点の重みを w_i としたとき、有理ベジェ曲線 $\mathbf{R}(t)$ は式 (1) によって定まる [10]。ここで $B_i^n(t)$ は Bernstein 基底関数である。

$$\mathbf{R}(t) = \frac{\sum_i^n B_i^n(t) w_i \mathbf{Q}_i}{\sum_i^n B_i^n(t) w_i} \quad (1)$$

本手法では曲線次数を 2 とし、実数 a, b, w により各制御点の位置と重み値を以下のように定めるものとする。

$$\begin{aligned} \mathbf{Q}_0 &= (-a, 0), & \mathbf{Q}_1 &= (0, b), & \mathbf{Q}_2 &= (a, 0), \\ w_0 &= 1, & w_1 &= w, & w_2 &= 1 \end{aligned} \quad (2)$$

これにより、稜線シワ形状は幅、高さ、曲がり具合をそれぞれ a, b, w の値を変更することにより制御することができる。図 13 に稜線シワ形状の模式図を示す。

稜線シワは、この曲線形状を先述の Catmull-Rom Spline 曲線に沿ってスウィープしたもものとして定義する。

次に、稜線シワ上での法線ベクトルの求め方について述べる。法線ベクトルを求めるには、まずピクセル p の位置における有理ベジェ曲線上のパラメータ t を求める必要がある。

$\mathbf{R}(t)$ の x 成分に式 (2) の値を代入しまとめると、以下の式を得ることができる。

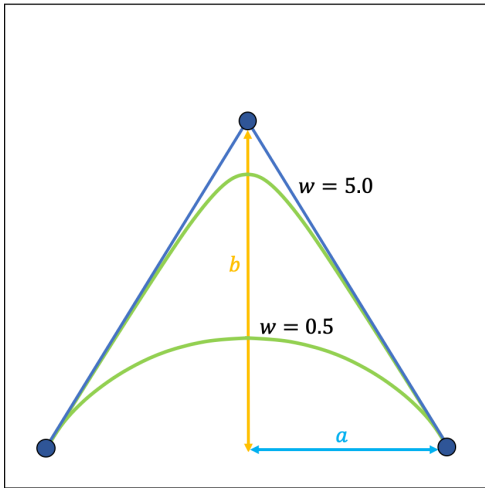


図 13 有理ベジェ曲線による稜線シワ形状

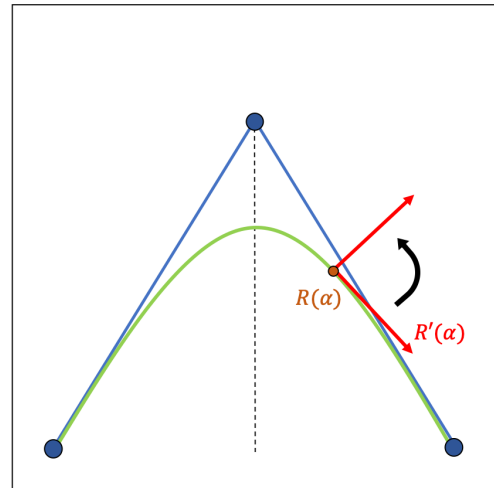


図 15 曲線の微分ベクトルと法線ベクトル

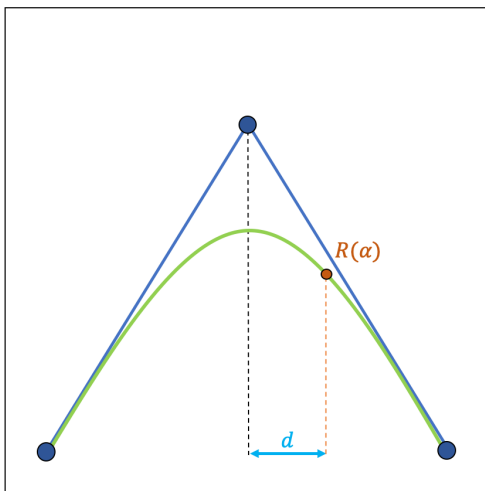


図 14 d と α の関係

$$R_x(t) = \frac{a(2t-1)}{(t^2-t)(2-2w)+1} \quad (3)$$

この値が d と等しくなる場合の t を求めればよい。(3) を d とおいて式をまとめると、以下のような t についての 2 次方程式を得ることができる。

$$(2d-2dw)t^2 + (2dw-2d-2a)t + d+a = 0 \quad (4)$$

この式の実数解のうち、 t の値が $\frac{1}{2}$ から 1 の間にあるものが、求めるパラメータ t となる。この解を α とする。 d と α の関係の模式図を図 14 に示す。

次に、 $\mathbf{R}(t)$ の 1 階微分 $\mathbf{R}'(t)$ より微分ベクトル $\mathbf{D} = \mathbf{R}'(\alpha)$ を求めておく。これを反時計回りに 90 度回転したベクトル $(-D_y, D_x)$ が 2 次元平面上の $\mathbf{R}(\alpha)$ に対する法線ベクトルとなる。 \mathbf{D} に関する模式図を図 15 に示す。

この \mathbf{D} と \mathbf{J} を用いて、以下のようにして法線ベクトル \mathbf{N} を求めることができる。

$$\mathbf{N}' = (-J_x D_y, -J_y D_x, D_x) \quad \mathbf{N} = \frac{\mathbf{N}'}{|\mathbf{N}'|} \quad (5)$$

次に、法線ベクトル \mathbf{N} をテクスチャデータに格納する



図 16 入力データと生成した法線マップ

ための色値構造に変換する。色値 R, G, B を以下のような式で求め、各成分の $[-1, 1]$ の領域を $[0, 255]$ に線形写像した。

$$\begin{aligned} R &= \frac{255}{2}(N_x + 1), \\ G &= \frac{255}{2}(N_y + 1), \\ B &= \frac{255}{2}(N_z + 1) \end{aligned} \quad (6)$$

図 16 は法線マップ作成の例である。シワ情報の部分が一番高い山型または一番低い谷型になるように、法線マップを作成する。図 16 は左がシワの情報を示した曲線であり、右が作成した山型の法線マップの例である。図 17 は山型の場合の法線ベクトルと RGB を示したものである。

3.2.3 距離マップ

次に輪郭シワのアルゴリズムで使用する距離マップを作成する。距離マップとは、入力されたシワ曲線からの距離を保存したテクスチャ画像とする。距離マップの数値は、曲線から離れるほど数値を大きくして保存する。曲線の片側のみ計算を行い、保存する。作成の例を図 18 に示す。

3.3 陰影計算

陰影計算では円柱モデルを使用し、シェーダーで計算を行う。

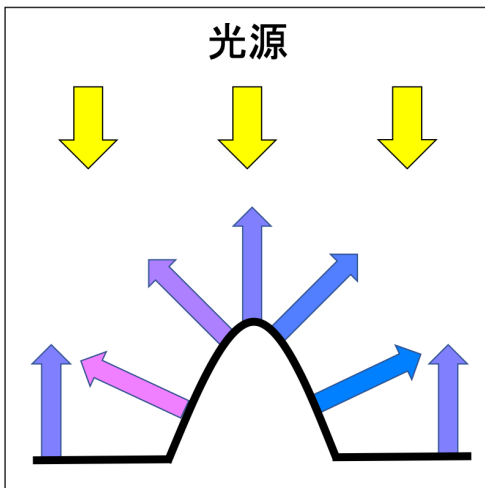


図 17 法線ベクトル

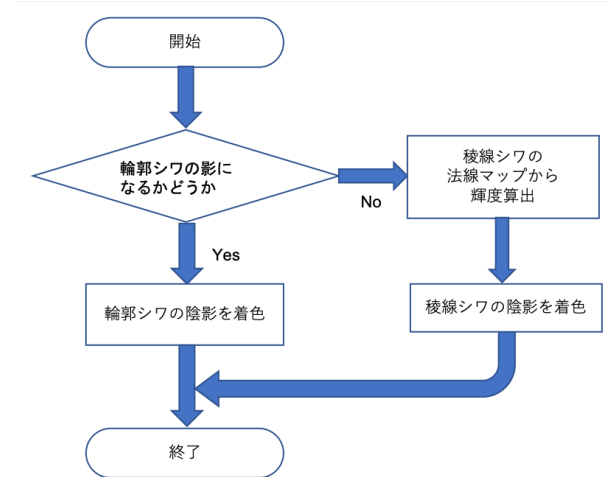


図 20 フローチャート

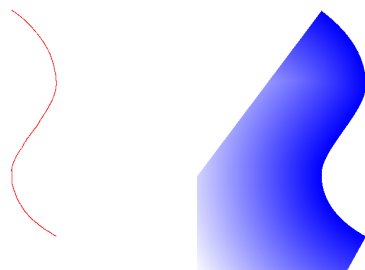


図 18 入力データと生成した距離マップ

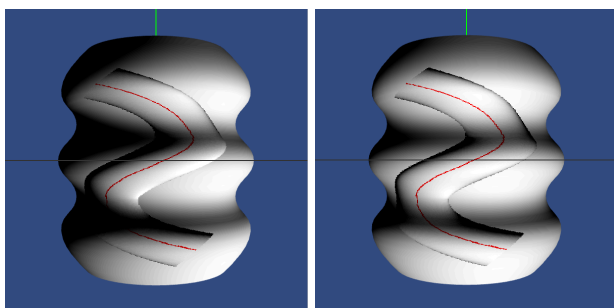


図 19 レンダリング例

3.3.1 法線マップによる陰影計算

法線マップによる計算では、ランバート反射という反射モデルを使用する。面の法線ベクトルを \mathbf{N} 、光源の方向ベクトルの単位ベクトルを \mathbf{L} とする。レンダリングではその時の明度を、面の法線ベクトルと光源の方向ベクトルを反転した単位ベクトルの内積 $-\mathbf{N} \cdot \mathbf{L}$ で求める。

図 19 に図 16 の入力データと法線マップによるフォンシェーディングのレンダリング結果の例を示す。赤色の線が入力データとなる。本研究ではアニメ塗りを対象としているため、陰影の出力ではトゥーンレンダリングの結果を使用する。トゥーンレンダリングでは、階調を数段階程度に落としてレンダリングを行う。

3.3.2 距離マップによる陰影計算

まずシワデータに対して、シワの高さを h を設定する。

そのシワの高さ h と、面の法線ベクトルと光源の方向ベクトルの角度を θ によって、曲線からどの距離まで陰影ができるかを計算する。曲線からの距離は、距離マップから取得した数値 d を使用する。その式を式 (7) に示す。この式 (7) を満たす時、陰影を表示する。

$$d \leq h \tan \theta \quad (7)$$

3.3.3 2種類の陰影計算の組み合わせ

複数のシワ線に対して陰影計算を行うとき、稜線シワと輪郭シワの両方があり、法線マップと距離マップの両方の陰影計算をする場合がある。このときの陰影計算を図 20 にフローチャートで示す。対象のピクセルが距離マップによって陰影が付く範囲だった場合、陰影を付ける。そうでない場合は法線マップによるトゥーンレンダリングの結果を反映する。

4. 結果と考察

4.1 稜線シワ

稜線シワでは法線マップを作成し、それを使用して陰影計算を行う。本研究ではアニメ塗りを対象としているため、トゥーンレンダリングを行った。明度が 0.1 以下、0.5 以下、それ以外で色分けを行った。

4.1.1 1本の稜線シワ

図 21 は左が入力したシワのデータであり、中央がシワのイラスト、右が生成した山型の法線マップである。そして図 22 がレンダリング結果となる。左図は光源が正面の場合、右図は光源が右にある場合である。各制御点の位置と重みは、 $a = 30.0, b = 100.0, w = 5.0$ で出力した。

法線マップによるレンダリングによって、稜線シワの 1 本のときにできる陰影を目指した。結果、シワ線の部分が高くなり、片側に広く陰影がついている。これによって稜線シワの 1 本の場合の、布が盛り上がっている部分にできる陰影を生成できた。

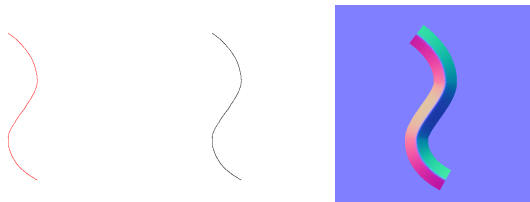


図 21 1本の稜線シワの法線マップの生成

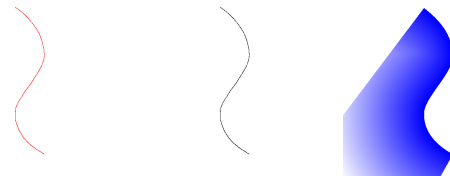


図 25 距離マップの生成

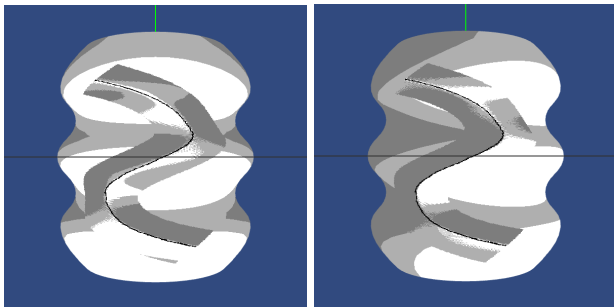


図 22 1本の稜線シワの法線マップによるレンダリング結果

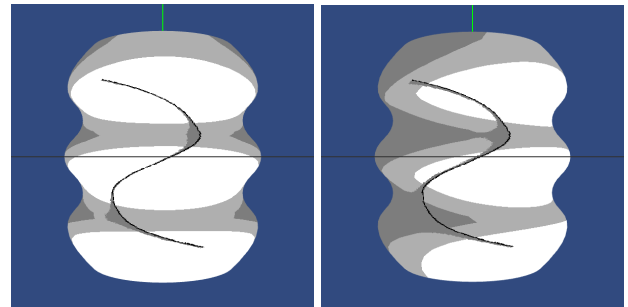


図 26 距離マップによるレンダリング結果

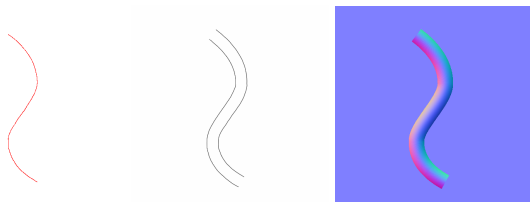


図 23 2本の稜線シワの法線マップの生成

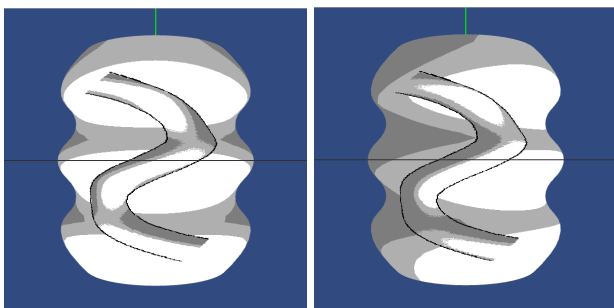


図 24 2本の稜線シワの法線マップによるレンダリング結果

山型の稜線の結果では、陰影のジャギーが目立ってしまっている。これは、法線マップのサイズが小さく、精度が低いために起きていると考えられる。

4.1.2 2本の稜線シワ

図 23 は左が入力したシワのデータであり、中央がシワのイラスト、右が生成した2本の稜線シワの法線マップである。そして図 24 がレンダリング結果となる。左図は光源が正面の場合、右図は光源が右にある場合である。各制御点の位置と重みは、 $a = 20.0, b = 100.0, w = 0.3$ で出力した。

法線マップによるレンダリングによって、稜線シワにできる陰影を目指した。結果、左右の線の間で布が山のように膨らんでいる陰影を生成することができた。これによって稜線シワの布が膨らんでいる部分にできる陰影を生成で

きた。

これも1本の稜線シワと同じように、陰影のジャギーが目立ってしまっている。

4.2 輪郭シワ

輪郭シワでは距離マップを作成し、それを使用して陰影計算を行う。図 25 は左が入力したシワのデータであり、中央がシワのイラスト、右が生成した距離マップである。そして図 26 がレンダリング結果となる。左図は光源が正面の場合、右図は光源が右にある場合である。

結果、輪郭シワの特徴であるシワ線の片側に対してのみ、陰影を生成することができた。これによって輪郭シワにできる陰影を生成できた。

5. まとめ

本論文では、キャラクターイラストの洋服のシワに対して、キャラクターイラスト特有の陰影を自動で生成する手法を提案した。シワを分類し、その分類したシワに対してアルゴリズムを変更することで、洋服シワに対するイラスト特有の陰影付けを実現した。しかし、シワ線1本で稜線シワと輪郭シワの両方の意味を持つシワ線に対しては両方設定する必要がある。また、今回は光源が一方の場合の計算方法であり、複数の方向の光源がある場合には使用するの難しい。今後は、イラスト作成者の望む陰影生成がより実現できるように、上記の部分の実装を進めていく必要がある。

参考文献

- [1] CELSYS, I.: CLIP STUDIO PAINT, <https://www.clipstudio.net/>. 参照: 2019.02.15.
- [2] Adobe Systems, C.: adobe photoshop CC, <https://>

- www.adobe.com/jp/products/photoshop.html. 参照:
2019.02.15.
- [3] Sýkora, D., Kavan, L., Čadík, M., Jamriška, O., Jacobson, A., Whited, B., Simmons, M. and Sorkine-Hornung, O.: Ink-and-Ray: Bas-Relief Meshes for Adding Global Illumination Effects to Hand-Drawn Characters, *ACM Transaction on Graphics*, Vol. 33, No. 2, pp. 16:1–16:15 (2014).
 - [4] Igarashi, T., Matsuoka, S. and Tanaka, H.: Teddy: A Sketching Interface for 3D Freeform Design, *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, New York, NY, USA, ACM Press/Addison-Wesley Publishing Co., pp. 409–416 (online), DOI: 10.1145/311535.311602 (1999).
 - [5] Nealen, A., Igarashi, T., Sorkine, O. and Alexa, M.: FiberMesh: Designing Freeform Surfaces with 3D Curves, *ACM Trans. Graph.*, Vol. 26, No. 3 (online), DOI: 10.1145/1276377.1276429 (2007).
 - [6] Gingold, Y., Igarashi, T. and Zorin, D.: Structured Annotations for 2D-to-3D Modeling, *ACM Trans. Graph.*, Vol. 28, No. 5, pp. 148:1–148:9 (online), DOI: 10.1145/1618452.1618494 (2009).
 - [7] Jayaraman, P. K., Fu, C.-W., Zheng, J., Liu, X. and Tien-Tsin: Wong Globally Consistent Wrinkle-Aware Shading of Line Drawings, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 24, No. 7, pp. 2103–2117 (2018).
 - [8] 青木明優花, 阿部雅樹, 渡辺大地: キャラクターイラストにおける洋服シワを考慮した影形状のリアルタイムレンダリング, *NICOGRAPH2019* (2019).
 - [9] Barry, P. J. and Goldman, R. N.: Recursive evaluation algorithm for a class of Catmull-Rom splines, *ACM SIGGRAPH Computer Graphics* (1998).
 - [10] G.E.Farin: NURBS - 射影幾何学から実務まで - 第2版, 共立出版 (2001).