

# NPRにおけるストローク方向の決定と 水墨画調レンダリングへの適用

## Generation of Stroke Direction for NPR and Its Application to Suiboku-ga Style Rendering

川崎 敬二      中丸 幸治      大野 義夫

Keiji Kawasaki    Koji Nakamaru    Yoshio Ohno

{keiji, maru, ohno}@on.cs.keio.ac.jp

慶應義塾大学大学院理工学研究科

Graduate School of Science and Technology, Keio University

概要: 本論文では, 筆を用いて描く画法を模倣するような NPR の共通の課題である, 物体形状を反映したストローク方向の自動生成手法を提案する. 提案手法では, 入力として与えられた 3 次元形状に対して 3 次元的な細線化を施し, 得られた物体の骨組みを利用し, ストロークの方向を決定する. 生成されるストローク方向は, 物体表面上で一意的に決定するため, 動画作成時にも効果的である. また, 生成されたストローク方向の可視化の一例として, 水墨画調のレンダリングを実装した. 本手法により生成された方向を用い, 3 次元物体表面上で, 物体全体を塗ることができるような開始点を一意に定めることにより, 静止画のみならず, ちらつきの少ない動画も水墨画調で出力可能である. 本手法により生成されるストローク方向, 並びに動画作成の概念は, 水彩画, 鉛筆画などを模倣する NPR にも拡張可能であると考えられる.

キーワード: ノンフォトリアリスティックレンダリング, ストローク方向, 水墨画, 動画, 筆

Abstract: In the painterly rendering, the direction of the brush stroke is a very important factor. We propose a technique for automatic generation of ideal stroke direction reflecting the feature of the object. Our technique generates the stroke direction using 3D thinning of the object. As this technique specifies the direction on the 3D surface, it is suitable for animation. We also implemented suiboku-ga-style rendering using our technique. We specify the starting points of strokes on the 3D surface in addition to the stroke direction. By doing so, our rendering algorithm can generate not only a still picture but also the animation with little flickering in suiboku-ga style. We believe that our stroke direction specification algorithm can also be applied to other NPRs such as pencil sketch drawing.

keywords: NPR, Stroke direction, Suiboku-ga, Animation, Brush

## 1 はじめに

近年, CG 技術の発展に伴い, 必ずしもリアルではないような出力を得るレンダリング (ノンフォトリアリスティックレンダリング: NPR) が注目されている. ここでは, その中で特に, 油絵, 水彩画など, 筆を用いて描く画法を模倣した NPR を, 筆絵調 NPR と呼ぶ. 筆絵調 NPR においては, 物体を描画するストロークの方向の決定が非常に重要な要素である.

筆絵調 NPR は, これまでに多くの研究がなされているが, 日本の水墨画, たとえば宮本武蔵の作品に見られるような, 少ない本数かつ長いストロークを用いた描画については, あまり扱われていない. また, 筆絵調 NPR ではストロークの流れが残るため, 動画にしたときのフレーム間の

コヒーレンスが保つのが難しく, 特別な処理が必要となる.

そこで, 本研究では, 大域的な物体形状を反映できる, 新しいストローク方向の自動生成手法を提案する. この手法では, 入力に与えられた 3 次元形状に対して 3 次元的な細線化を施し, 得られた骨組みを元にストロークの方向生成を行う. ストロークは 3 次元物体表面上で定められる方向を用いて, 乱数によらずに決定されるため, 動画に適したコヒーレンスを維持することができる.

以下では, まず従来の研究について概観し, 次に本研究の手法の詳細を述べる. そして, 実装による結果を示し, 最後に結論と今後の課題を述べる.

## 2 従来の研究

### 2.1 ストローク方向の生成

自動的に最適なストローク方向を生成することに注力した研究はあまり無く、多くが筆絵調 NPR のシステムの一要素として取り組んでいる。近年の多くの NPR に関連した論文においても、ユーザの入力によって方向を決定するもの、もしくは比較的簡単で汎用的な手法で方向を生成するものが多い。後者の場合、たとえば 1 つの基本ベクトル (視線ベクトル、物体の中心軸を表すベクトル等) と法線によって、ストローク方向を生成する [1, 2, 3, 4]。基本とするベクトルにより、生成される方向に多少の違いがあるが、共通して、全体を 1 つの基本ベクトルにより表せないような複雑な物体に対して適切なストローク方向を生成することができない。

イメージベースのペン画調 NPR の研究 [5] では、インタラクティブにユーザが方向を指定できるようなシステムを構築した。方向さえユーザが適切に定めれば、入力は 2 次元画像であるにも関わらず、非常にクオリティの高い出力が可能である。ペン画においても、描画時のストローク方向が、結果の完成度において重要な要素を担っていると言える。

投影面上ではなく、3 次元形状上で直接ストロークを生成する手法としては、Girshick らの主方向 (Principal Direction) に基づく方法が知られている [6]。これは、表面上の凹凸などにも応じた流れの場を表面に定義することができ、動画に必要なコヒーレンスの維持にも向いた、優れた手法である。しかし、この手法は物体の大域的な形状ではなく、より局所的な曲面形状を反映するため、本研究で目的とするような水墨画の描画には必ずしも適さない。

### 2.2 水墨画調レンダリング

水墨画調のレンダリングに関しては、千葉則茂および岩手大学工学部のグループによる意欲的な研究がある。千葉らは、まず 99 年に、墨と水の移動をシンプルに表現した 2 次元のオートマトンを開発した [7]。濃度平衡や蒸発などをシミュレートすることにより、墨、水の自然な滲みを再現している。

また、同論文で樹木の骨格モデルを元にした樹木のレンダリングも提案している。その他にも、山岳の 3 次元幾何モデルをもとにした水墨画調レンダリングも発表した [8]。

一般的な物体を水墨画調にレンダリングする手法として、ポリゴンデータを入力とするものも提案した [1]。汎用的に

さまざまな物体を水墨画調で出力可能であるが、ストローク方向の作成において非常に簡単な手法を用いており、物体の角度によって方向が全く変わってしまうなど、動画に適し、かつ人間らしいストローク方向は生成されていない。

Shih による研究 [9] は 3 次元の幾何モデルを入力とし水墨画らしい絵を出力する。この論文では、水墨画で良く描かれる対象である岩と木を描くことに特化しており、それぞれテクスチャを定式化し、さまざまなタイプの出力が可能になっている。しかしあくまで出力が可能なのは岩と木に関してのみであり、実的に用いるには不十分であるという欠点がある。

それぞれの研究は動画作成は基本的には考慮していない。筆絵調の NPR における動画作成時のちらつきを抑える手法 [2] があるが、この手法ではストローク開始点を乱数によって定めている。油絵等細かいストロークの集合で描く画法と違い、水墨画では長いストロークでシンプルに描くため、乱数によらずにストローク開始点を定める必要がある。

## 3 ストローク方向の自動生成

### 3.1 手法の流れ

手法の主な流れを図 1 に示す。まず、入力として与えら

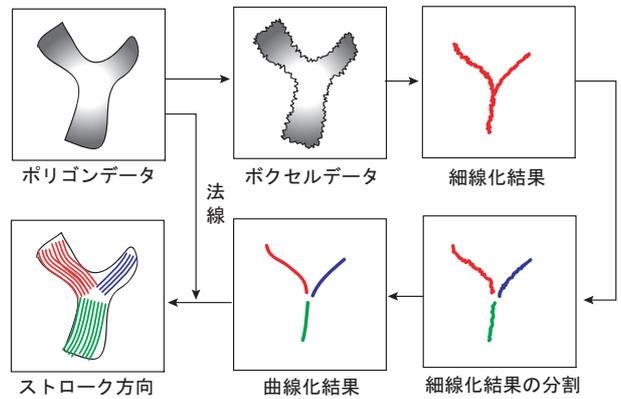


図 1: ストローク方向生成の流れ

れた物体を離散的なボクセルデータに変換し、得られたボクセルデータに対し、細線化処理を施す。これにより、ノイズを含んだ細線化結果が得られる。この細線化結果を、交点でそれぞれの曲線に分解する。結果は離散値でかつノイズを含むため、滑らかなストローク方向を生成するために、それぞれに曲線化処理を適用する。このようにして得られ

た滑らかな中心軸の集合と、物体の法線を元に、物体表面上の各点における方向を求める。

### 3.2 細線化処理

細線化処理には、ボクセルデータを入力として、細線化処理を行う研究 [10] を利用する。

まず与えられた 3 次元物体をボクセルデータに変換する。それぞれのボクセルには、物体表面までの最小距離を格納する。初期段階として、物体の表面に位置するボクセルをリストに登録し、以後はリストへボクセルの登録を繰り返しながら削除判定を行い、削除できる場合は削除し、隣接するボクセルをリストに追加する。ヒゲ (誤って検出される中心軸) の発生を防ぐため、表面からの距離が近く、かつ注目画素の周りの 0 ボクセルが多い順に削除する。削除可能であるかどうかの判定は、そのボクセルを削除したときとでトポロジーが保存されるかどうかを判定する連結数 [11] を調べることにより行う。

得られた細線化結果を、滑らかな曲線に置き換えることができるように、交点においてそれぞれの曲線に分割する。分割処理は、26 近傍に値を持つボクセルが 3 つ以上存在するようなボクセルを、交点とすることにより行う。本研究において曲線を分割する詳細度は、最終的な結果にほとんど影響を与えないため、そのような単純な処理で分割を行うことで十分である。

### 3.3 曲線化

細線化は、3 次元形状データを離散化し行うため、得られる細線も離散値であり、かつ処理の過程でノイズも含む。滑らかなストローク方向を生成するために、分割された細線化結果をそれぞれ滑らかな曲線に変換する。具体的な手順の流れを、図 2 に示す。

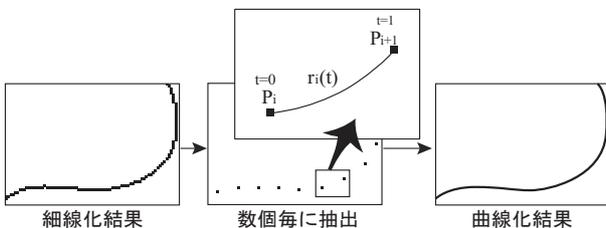


図 2: 曲線化の流れ

まず、細線化結果を元に、数個おきにボクセルを抽出する。この抽出の間隔は、高周波の曲線に対しては細かく、直

線に近い曲線に対しては大きかにする。このようにして抽出されたボクセル間をそれぞれベジエ曲線で補間し、全体として滑らかな 1 つの B スプライン曲線を構成する。i 番目の抽出点  $P_i$  とその次の抽出点  $P_{i+1}$  との間を補間する曲線  $r_i(t)$  は、ベジエ曲線の一般式より、

$$r_i(t) = (1-t)^3 P_i + 3t(1-t)^2 U_{i1} + 3t^2(1-t) U_{i2} + t^3 P_{i+1} \quad (1)$$

となる。このようにして表される補間曲線が、抽出点数 - 1 だけ作成されることになり、それぞれの曲線は未知の係数  $U_{i1}$ ,  $U_{i2}$  を持つ。それら全ての係数は、曲線同士の 1 次、2 次微分値が等しい等の条件を元に連立方程式を解くことにより求められる。

### 3.4 ストローク方向の作成

物体表面上のある点 p における法線ベクトルを n としたとき、p におけるストローク方向は次のようにして求める (図 3 参照)。まず、p からもっとも近い中心曲線上の点 q を探す。そして、この q における中心軸曲線の接線ベクトル c を考える。このような c には、図中のものと、その反対方向のものが考えられるが、空間中でより高い方向に向くものを考えると、 $d = c \times n \times n$  によって求まるベクトルは、点 p における接線ベクトルとなり、その方向は上から下へ中心曲線に沿ったものになる。

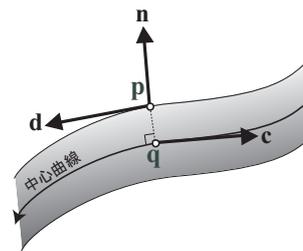


図 3: ストローク方向生成のようす

なお、c と n が完全に平行な場合は、d が零ベクトルになる。このような場合には方向が得られないため、c に垂直なベクトルをつくり、それを c の代わりに用いる。

以上のような処理を、物体表面上の全ての点に対して行うことにより、3 次元物体表面上に一意にストローク方向を決定する。

細線化結果に交点が発見され、複数の曲線により構成される中心軸が得られた場合は、同様の処理により方向を生

成すると、最も近い曲線が異なる箇所において、ストローク方向が急激に変わってしまう。そこで、ある点に対するベクトル  $c$  は、各曲線の接線方向を、各曲線との距離に反比例させた重みにより内挿し、決定する。内挿の際の重みは、2次元画像のモーフィング技術において、複数のベクトルが存在する方向場を元に方向を内挿する研究 [12] を利用し、下記の様に定める。

$$weight = \left( \frac{length^\gamma}{\alpha + dist} \right)^\beta \quad (2)$$

ここで、 $dist$  は注目点と曲線の距離、 $length$  は曲線の長さを表す。パラメータ  $\alpha$  が非常に小さく、同時に  $dist$  も小さかった場合、 $weight$  は非常に大きな値になり、その結果、近くの曲線にしかほとんど影響されない方向が作成される。そのような振舞を調整するパラメータが  $\alpha$  である。 $\beta$  が大きいと、同様に近くの曲線の影響が大きくなり、小さいと他の曲線の影響も受けやすい重みになる。このように、他の曲線の影響の度合を調整するのがパラメータ  $\beta$  である。パラメータ  $\gamma$  は基本的には  $[0, 1]$  の値を持ち、曲線の長さが重みに影響する度合を決める。

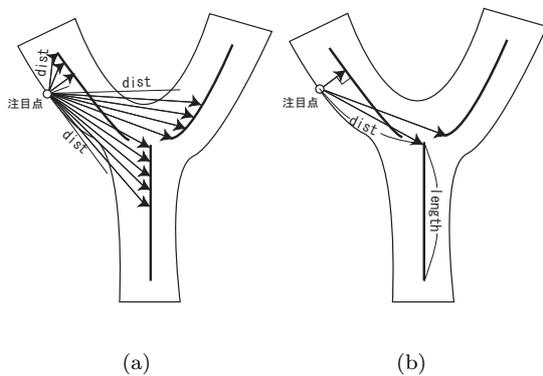


図 4: 注目点に対する曲線検出のようす

本研究では、1つの曲線を長さが一定のベクトルの集合と考えることによって、この重み関数を利用する (図 4(a) 参照)。これにより、注目点における  $c$  が求まる。この際、ベクトルの長さは曲線の形状を十分に近似するように小さくとる。

各曲線がそれぞれ直線的である場合には、図 4(b) のように注目点から各曲線に対する最近傍の点を  $dist$  として定め、各曲線の長さを  $length$  として計算してもよい。この処理によって、細かいベクトルによる近似を行う場合とほぼ同様の結果を、高速に求めることができる。

なお、本研究で行った実験では、 $\gamma$  を常に 1 としている

が、上記の高速化の処理のように不均一な長さの曲線を考える場合、 $\gamma$  によって曲線の細かい形状の影響を加減することも可能である。

ストローク方向は物体表面のあらゆる点で求まるが、そのような計算は高価である。そこで、実際の処理では、3角形の各頂点のみでストローク方向を求める。3角形の内部では、後述するように、フォンのスムーズシェーディングと同様の考えによって内挿した方向を用いることにより、滑らかな方向の変化を実現する。

## 4 水墨画調レンダリング

### 4.1 全体の流れ

水墨画調レンダリングの全体の流れは図 5 のようになる。

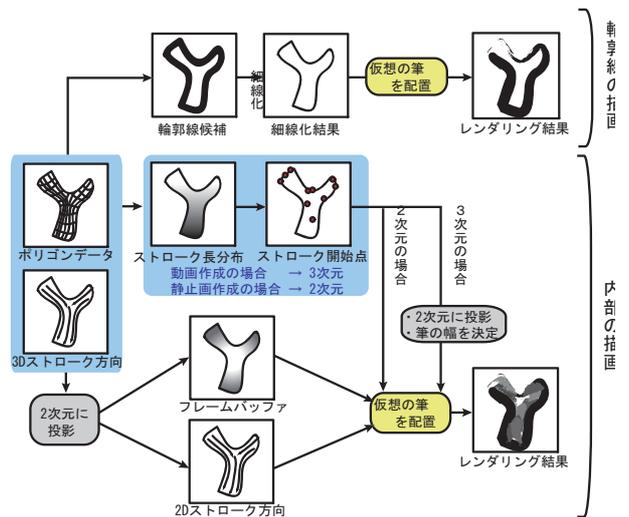


図 5: 水墨画調レンダリングの流れ

図 5 に示すように、輪郭線と内部に分けてそれぞれを描画する。輪郭部分の描画では、まず 3次元物体形状を元に、法線の情報を利用して輪郭部分を形成する候補となるピクセルの集合を 2次元的に求める。そのようにして求められた候補点を細線化し、その線に仮定の筆を配置していくことにより、輪郭部分の描画を行う。

内部の描画では、前処理としてストローク長の分布を求める。これを参照し、できるだけ長いストロークを優先的に確保し、一定の間隔を取りながら、全体をおおうようなストロークの開始点の集合を求める。動画作成を目的とする場合、ストロークの開始点は物体表面上に一意であることが望ましいため、これらの処理を 3次元的に行う。静止画を作成する場合は、2次元的に処理を行うことにより、高

速化が図れる。また、3次元的に開始点を求めた場合、それを2次元に投影し、かつ描画時におけるストロークの幅を計算しなおす必要がある。以下ではより複雑な処理となる3次元での処理について述べる。

物体を2次元に投影したフレームバッファを描画色の要素として、ストローク方向を描画方向の要素として、ストローク開始点を、描画の開始点の要素としてそれぞれ使い、仮定の筆を配置することにより、内部を描画する。筆を始めとする、水、墨などのモデル化には Zhang らの手法 [7] を利用した。

## 4.2 輪郭線の描画

### 4.2.1 輪郭候補の検出

本研究では、様々な物体で比較的安定した処理が行われるよう、以下の条件に当てはまるピクセルを輪郭部分の候補として抽出する。この処理は投影された2次元画像上で行われ、Curtis の手法 [13] とほぼ同様のものである。

- 注目ピクセルにおいて、隣接するピクセルとの法線の変化が一定以上の場合
- 注目ピクセルにおいて、隣接するピクセルとの奥行き値の違いが一定以上の場合 (背景は無限遠と考えるので、背景と隣接するピクセルは必ず該当する)

検出された輪郭候補は、その条件により、多少幅を持つ線の集合となるので、ストロークを配置するために、幅1ピクセルの線の集合になるよう細線化を施す。本研究では一般的な細線化の手法として知られる、Hilditch のアルゴリズムを用いて、輪郭候補を細線化する。

### 4.2.2 輪郭線ストロークの生成

上記手法によって抽出された輪郭線を表すピクセル群に対して、まずストローク開始点を定める。ストローク開始点は、スクリーンにおいて上端から走査することにより検出するが、その際、各走査において右からたどるか左からたどるかを交互にする。そうすることにより、輪郭線が1つのストロークで描けなかった場合に、前回のストロークの開始点から始めることができる。

そのようにして決められたストローク開始点から、8近傍に輪郭線が無いかどうかを調べる。あればそのピクセルをストロークとして登録し、今度はそのピクセルに関して

8近傍を調べ、同じ事を行う。以上を繰り返して、ストロークを生成する。終了条件は以下の通りである。

- ストロークが画面の外に出てしまうとき
- ストロークがループ、並びに交差を形成したとき
- ある一定の長さを超えた場合

上記終了条件によって終了したストロークを結果画像に配置し、今回描いたストロークに関しては輪郭線候補から除外する。

これを輪郭線を表すピクセルが全てなくなるまで繰り返す。

## 4.3 内部の描画

### 4.3.1 ストローク長分布

各ポリゴンの頂点からストローク方向をたどり、終了するまでに通った長さを各頂点に保存しておくことにより、ストローク長の分布を求める。この処理によって、実際に配置するストロークを生成するわけではないので、3角形内の方向は3つの頂点における方向を平均した方向とし、簡易的にストローク方向を追跡する。

追跡の開始点がある辺以外の2辺において、それぞれ構成する頂点を  $p_1, p_2$  としたとき、その辺は  $p_1(1-t) + p_2t$  ( $0 \leq t \leq 1$ ) により表される。開始点から3角形の平均した方向ベクトルを伸ばした直線が、この直線と交わる交点において  $t$  が  $0 \leq t \leq 1$  であったとき、その点を次の開始点とする。次にたどる3角形は、同じ辺を共有する3角形となる。また、特殊な条件下 (追跡途中、3角形の頂点に行き着いた場合など) において、ループするのを防ぐため、1つ前にたどった3角形はたどらないようにする。ストロークをたどる終了条件は以下の通りである。

1. 次にたどる3角形がない場合 (ポリゴンがそこで切れてしまっている場合)
2. 次にたどる3角形の法線が、前の3角形の法線と一定以上の角度を持つ場合
3. 方向をたどることにより、今まで既にたどった3角形に行き着いた場合 (ストロークがループ、並びに交差を形成した場合)

法線変化によりストロークを終了させないと、ある物体を構成する様々な部位を全て1つにたどってしまうこと

になる。入力として与えられたデータから、部位を構成するポリゴンが参照できる場合は、そちらを利用することによって、より厳密に部位毎にストロークを追跡できる。

#### 4.3.2 ストローク開始点

動画作成時のちらつきを抑えるために、物体全体を塗ることができるストロークの開始点の集合を一意に求める。その際、4.3.1 節で作成されたストローク長分布を参照しながら、できるだけ長いストロークを優先的に配置していく。具体的な処理は図 6 のようになる。

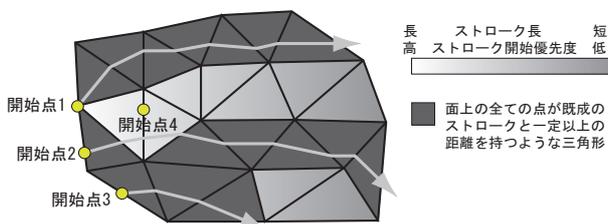


図 6: ストローク作成のようす

まず、各 3 角形上で、ストローク長の分布を頂点に保持されたストローク長を線形補間した値として定める。このストローク長の分布を示しているのが、図中の濃淡部分である (より明るい部分がよい長いストローク長を示す)。

このとき、最初のストロークの開始点は、辺上のすべての点のうち、ストローク長のもっとも大きい点 (図中、開始点 1) とし、そこからストロークを 4.3.1 節と同様に伸ばしていく。ストロークの終了条件は、4.3.1 節に記載したものに加えて、ストロークを伸ばして既成のストロークと一定距離以内になった場合、という条件を加える。作成されたストロークに対して、面全体の点が全て一定の距離以内にあるような 3 角形にはラベルを付ける (図中、暗い 3 角形)。ラベル付けられた 3 角形は、今後ストローク開始点を作成せず、また他のストロークが伸びてきてもそこで終了になる。

2 番目以降の開始点は、ラベル付けられていない 3 角形の辺上で、これまでに作成されたストロークから一定の距離を保ち、かつ、最もストローク長の大きい点とする。ストロークの追跡は、先と同様に行う。こうして開始点の配置とストロークの追跡を繰り返し、配置可能な開始点がなくなった時点で終了となる。

追跡したストロークが筆の幅に対してあまりにも短かった場合、描画時に不自然なので、ストロークは作成せず、開始点作成候補からも除外する。逆に、できるだけ短いスト

ロークが生成されるのを防ぐために、ある程度の重なりを許容する。

絵画風の出力を行う際、物体全体の基本的な色を大まかに、陰影づけなどを細かく描画したい場合がある。そのような場合には、複数の幅で別々に開始点の決定およびストロークの追跡を行う。

#### 4.3.3 ストローク開始点の 2 次元への投影

4.3.2 節によって 3 次元的に決定されたストローク開始点が、現在の視点から見える位置にあるとは限らない。また、同様の理由により、3 次元的には 1 つのストロークであっても、2 つ、3 つと分割される可能性がある。

開始点が視点から見えない場合は、見える位置まで追跡した後、そこを 2 次的に用いる開始点とする。また、再び物体に隠れ、現れる場合もあるので、追跡を続け、あらたに現れた点を同様に開始点とする。3 次元的に決定された全てのストローク開始点について、以上のような処理を行い、2 次元に投影することにより、実際にレンダリング結果においてストロークの開始点となる点を求める。

#### 4.3.4 フレームバッファの分割

水墨画を再現するためにシンプルにストロークを配置するといっても、物体表面を一度ずつ塗ったのでは、光による陰影がまったく表現されない、平面的な出力になる。実際的水墨画でも、最低でも色を元に 2, 3 の階層にわけ、塗り重ねることにより立体感を表現する。よって、色の見本となるフレームバッファを、色の濃度値によってユーザの指定した数に分割し、分割されたフレームバッファにたいして、それぞれ内部を塗る処理を行う。

ここで、分割する数を  $d$  とし、ユーザが任意に定められるようにする。分割の閾値は、最大の明るさを 255 としたとき、 $d$  で割った値とする。このようにして分割したフレームバッファに対してそれぞれ内部を塗る処理を行う。これにより、ユーザが望むリアルさの範囲で描画を行える。

#### 4.3.5 ストロークの幅の決定

4.3.2 節により求められたストローク開始点は、ストロークを伸ばした際に、3 次元的に均等な配置になるような点として決定される。そのため、投影した際に必ずしも 2 次的に均等の距離にあるとは限らない。そのため、投影された開始点を元に、それぞれの開始点から延ばすストロ

クの幅を決定しなおす必要がある。具体的には、以下のよ  
うな手順でストロークの幅を決定する (図 7 参照)。

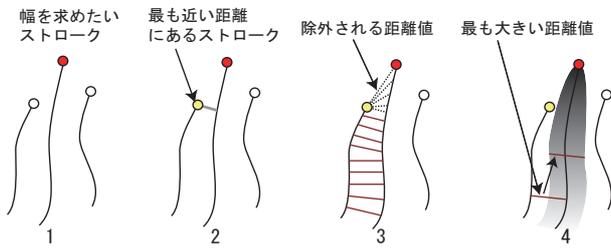


図 7: ストロークの幅の決定手順

1. 2次元に投影された全ての開始点からストローク方向をたどり、ストロークを延ばす
2. 幅を求めたいストロークから、最も近い距離にあるストローク (隣に配置されるストローク) を探索する
3. 幅を求めたいストローク上の全ての点から、2で求められたストロークへの最も近い距離の集合を求める。その際、どちらかのストロークの端点を通るものは除外する
4. 求められた距離の集合の中から、もっとも大きい値をストロークの幅として決定する

#### 4.3.6 内部ストロークの生成

図 5 のように、濃さを決定する要素としてフレームバッファを、方向を決定する要素として 2次元に投影されたストローク方向を、開始点を決定する要素として 2次元に投影されたストローク開始点をそれぞれ参照しながら、ストロークを配置する。描画する濃さは、フレームバッファにおいて、配置するストロークが通過するピクセル集合の中で最も明るい色とする。3次元の方向追跡はポリゴン単位で簡易的に行ったが、ここではフォンのスムーズシェーディングと同様の考え方で内挿し、投影されたストローク方向を追跡することにより、より滑らかなストロークを作成する。

追跡において、現在のわれわれの実装では、単にピクセルごとにストロークの方向を定める方法をとっている。この方法では、3次元で求められたストロークからずれる可能性があるが、モデルを形成するポリゴンが細かく、3角形の隣接頂点ごと方向の変化が滑らかであれば、静止画としてほぼ問題のない結果が得られる。また動画においても開始点が一定しているため、フレーム間のコヒーレンスが保

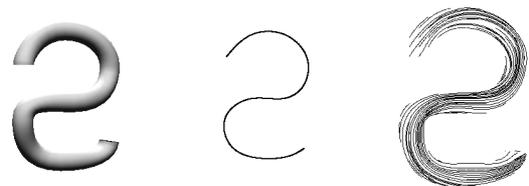
たれる。より厳密に 3次元のストロークに沿わせるためには、たとえばストロークの各頂点において内挿された方向を決め、エルミート補間を用いることなどが考えられる。

ストローク追跡の終了条件は、当然 4.3.1 節で 3次元的に方向をたどった際の終了条件を、2次元的に考えた場合と同様になる。加えて、ストロークがスクリーンの外に出してしまう場合と、一定の長さを越えた場合も終了とする。

また、4.3.4 節で述べたように、実際はフレームバッファを明るさにより分割するので、開始点を描画領域まで移動する必要がある場合がある。描画に必要なストロークの開始点は 2次元に投影されたもので全てなので、その投影された開始点を 2次元的に追跡し、描画領域に入った点を開始点とすればよい。

## 5 結果

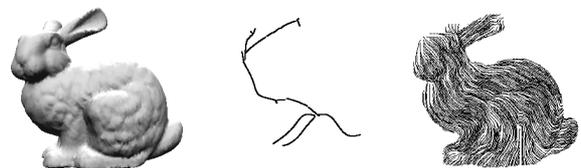
本研究の実装、実行は Pentium4 1.6GHz、メインメモリ 1024MByte、Windows2000 上で行った。



(a) S 字型の形状



(b) 鳥の形状 (式 (2) で  $\alpha = 0, \beta = 10, \gamma = 1$  とした)



(c) 兎の形状 (式 (2) で  $\alpha = 0, \beta = 10, \gamma = 1$  とした)

図 8: 細線化およびストローク方向生成結果

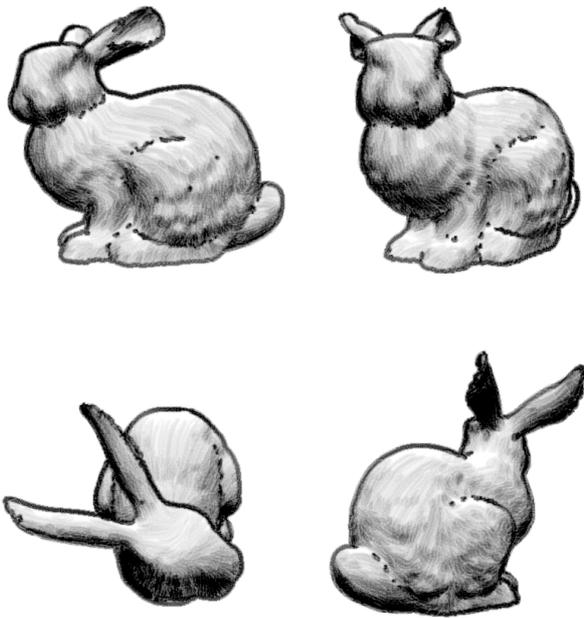


図 9: 兎の形状に対する静止画出力結果 (分割数  $n = 10$ )

従来手法では理想的な方向を求めにくいような3つの物体に対して、細線化し、ストローク方向を生成した結果を図8に示す。水墨画調レンダリングにおける、静止画の出力例として、図9に兎の形状に対する出力結果を示す。動画の出力例として、図10に瓶の形状を回転させた際の一連のフレームを示す。静止画を連続で出力したものと比較すると、格段にちらつきが抑えられた結果になった。

細線化を行い、ストローク方向を生成するのに要する時間は、ポリゴン数により大幅に異なるが、6万ポリゴンを越えるような物体に対して2分程度であった。1フレームあたりの水墨画調レンダリングに要する時間は、ポリゴン数と分割数  $n$  により影響され、2~30分程度であった。

## 6 結論

本手法により、筆絵調の NPR において、従来ユーザが指定しなくてはならなかった理想的なストローク方向を、多くの物体に対して自動的に求めることができるようになった。ある物体に対して、部位ごとに処理しなくても、ほぼ安定した結果が出力可能である点において、対象データを選ばない汎用性の高い手法である。生成されるストローク方向は3次元的に一意に定められているので、動画を作成する際にも非常に有効である。

また、筆絵調レンダリングの1つである水墨画調レンダリングにおいて、本手法により求められたストローク方向

を用いることにより、水墨画らしい出力が可能であることが示された。同時に、できるだけ長いストロークでシンプルに塗るという水墨画において、乱数によらない配置により3次元物体表面上にストロークの集合を作成し、動画作成時におけるちらつきを抑える手法を提案した。

## 7 今後の課題

本手法において、生成されるストローク方向に直接的に影響する、細線化アルゴリズムの精度上昇や拡張がまず挙げられる。たとえば一点に収束する物体の場合、そのままではストローク方向を求めることができないが、短い線分を点の位置に配置することで、本研究の枠内で処理できるようになる。物体表面上の皺に沿ったストローク方向なども、細線化アルゴリズムの拡張を行うことで、考慮できる可能性がある。この他、Girshick らの手法 [6] などで得られるストローク方向と、本手法のストローク方向を組み合わせることで、双方の欠点を補い合うことも考えられる。

水墨画調レンダリングにおいては、水墨画調ならではの特殊な塗り方(省略、テクスチャの表現、塗り重ねなど)を同じシステム上で扱えるようになれば、よりさまざまな出力を生成することが可能になる。

## 謝辞

この研究の一部は、文部科学省 21 世紀 COE 「アクセス網高度化光・電子デバイス技術」に対する補助を得て実施したものである。

## 参考文献

- [1] 佐藤, 藤本, 村岡, 千葉, “ポリゴンベース幾何モデルの水墨画調レンダリング法,” 情報処理学会グラフィックスと CAD 研究会, 103-007, pp. 37-42, 2001.
- [2] B. J. Meier, “Painterly rendering for animation,” SIGGRAPH 1996, pp. 477-484, 1996.
- [3] T. Haga, H. Johan, T. Nishita, “Animation Method for Pen-and-Ink Illustrations Using Stroke Coherency,” CAD & Graphics 2001, pp.333-343, 2001.
- [4] 川崎, 大野, “ポリゴンデータを入力とする水墨画調レンダリング,” NICOGRAPH, 18, pp. 121-126, 2002.

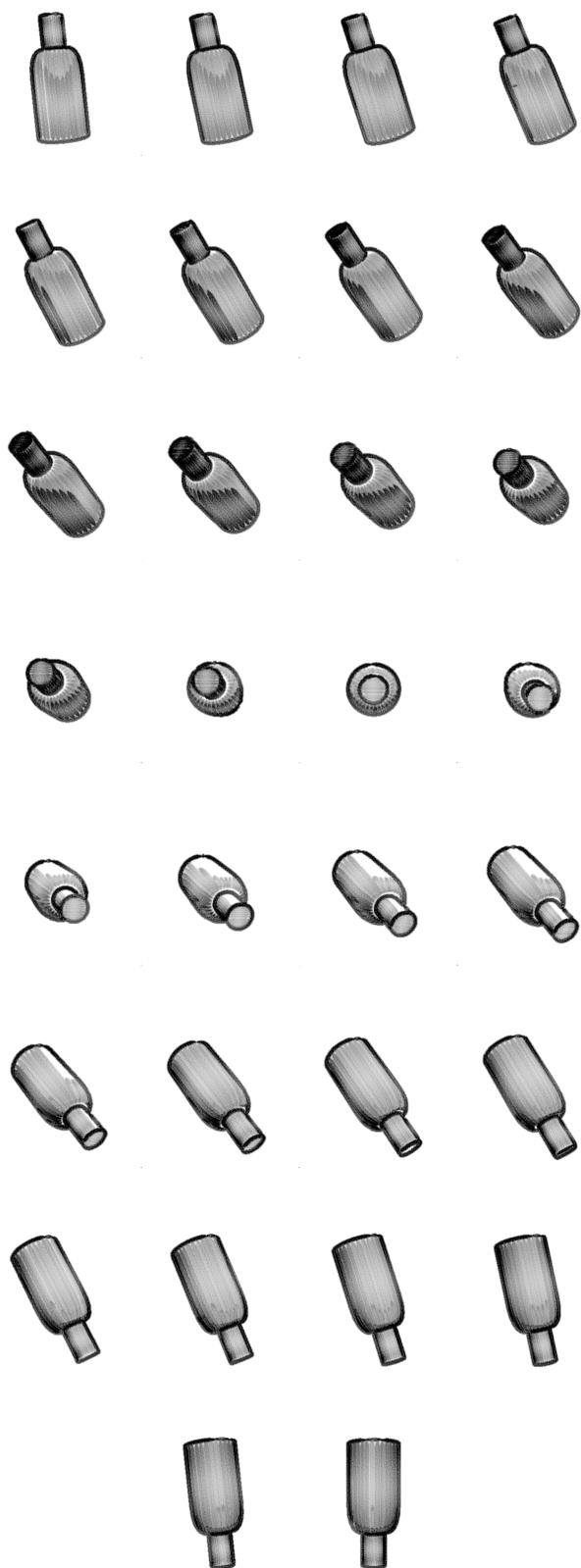


図 10: 瓶の形状に対する動画出力結果 (分割数  $n = 3$ )

- [5] M. P. Salisbury, M. T. Wong, J. F. Hughes, D. H. Salesin, "Orientable Textures for Image-Based Pen-and-Ink Illustration," SIGGRAPH 1997, pp. 401-406, 1997.
- [6] Ahna Girshick, Victoria Interrante, Steven Haker, Todd Lemoine, "Line direction matters: An argument for the use of principal directions in 3D line drawings," NPAR 2000, pp. 43-52, 2000.
- [7] Q. Zhang, Y. Sato, J. Takahashi, K. Muraoka, N. Chiba, "Simple Cellular Automaton-based Simulation of Ink Behaviour and Its Application to Suibokuga-like 3D Rendering of Trees," J. of Visualization and Computer Animation, Vol. 10, No. 1, pp. 27-37, 1999.
- [8] 佐藤, 千葉, 村岡, "山岳の水墨画調レンダリング," 情報処理学会グラフィックスとCAD研究会, 99-CG-94, pp. 61-66, 2000.
- [9] Z. Shih, "The Synthesis of Chinese Ink Paintings," NICOGRAPH International, pp. 11-20, 2002.
- [10] 斎藤, 森, 鳥脇, "ユークリッド距離変換を用いた3次元デジタル画像の薄面化および細線化の逐次型アルゴリズムとその諸性質," 電子情報通信学会論文誌, J79-D-II, 10, pp. 1675-1685, 1996.
- [11] 米倉, 櫻井, 鳥脇, 福村, "三次元デジタル画像における連結数と消去可能性について," 電子通信学会論文誌, J65-D, 5, pp. 652-659, 1982.
- [12] T. Beier, "Feature-Based Image Metamorphosis," SIGGRAPH 1992, pp. 35-42, 1992.
- [13] Cassidy Curtis, "Loose and Sketchy Animation," SIGGRAPH 1998, Conference Abstracts and Applications, p. 317, 1998.