

## 特徴稜線認識型メッシュスムージング手法の研究

清水 貴文, 伊達 宏昭, 金井 理, 岸浪 建史

北海道大学大学院情報科学研究科  
060-0814 北海道札幌市北区北14西9

本研究では, 3次元スキャナから得られる測定ノイズを含むメッシュモデルのノイズ除去を目的とした, 特徴稜線認識型メッシュスムージング手法を提案する. はじめに法線投票テンソルの固有値解析により特徴稜線を構成するメッシュエッジ群を抽出する. 次に抽出したメッシュエッジ群を *bilateral filtering* により特徴稜線方向にスムージングし, 最後にメッシュ全体を *bilateral filtering* によりスムージングすることで, 特徴稜線が稜線方向に滑らかで, かつ特徴稜線を境界として持つフェイスの測定ノイズが除去されたメッシュが得られる.

## A Mesh Smoothing Method with Feature Edge Identification

Takafumi Shimizu, Hiroaki Date, Satoshi Kanai, Takeshi Kishinami,

Graduate School of Information Science and Technology, Hokkaido University  
Kita-14, Nishi-9, Kita-ku, Sapporo 060-0814, Japan

In this paper, we propose a mesh smoothing method with feature edge identification for mesh models including the measurement noise obtained from 3D scanner. First, by doing the eigen analysis of normal voting tensor, mesh edge groups that compose the feature edges can be extracted. Second, a bilateral filtering is applied to the extracted mesh edge groups in order to generate smooth edges along feature edge directions. Finally, by applying modified bilateral filtering to the overall mesh, the mesh model is obtained where the feature edges are smoothed along their directions and measurement noise included in faces surrounded by feature edges are removed.

### 1. はじめに

近年, 光学式3次元スキャナや工業用 X 線 CT 装置といった3次元形状スキャナが普及し, デザインモックアップや工業製品の物理モデルから三角形メッシュモデル等のデジタルモデルを生成するリバースエンジニアリング技術が一般化してきた. 工業製品を表すメッシュモデルの表面は, 接平面が連続な領域を三角形面分群で近似したフェイスと, フェイス間の境界線である特徴稜線から成るが, 一般に測定より得られるメッシュには, フェイスや特徴稜線に測定ノイズが含まれる. しかし, メッシュモデルをモデリングや CAE に利用する場合, 特徴稜線が稜線方向に滑らかで, かつフェイスも滑らかなモデルであることが望ましく, これらの測定ノイズを除去する必要がある.

メッシュの測定ノイズを除去する方法として, 従来から様々なメッシュスムージング手法が提案されてきた. Taubin<sup>[1]</sup>は, メッシュに対する離散ラプラシアン行

列を定義し, その固有値解析とローパスフィルタリングによりメッシュのノイズを取り除く手法を提案した. Kobbelt ら<sup>[2]</sup>は, メッシュ表面の力学的エネルギーの最小化に基づくスムージング手法を提案した. また Desbrun ら<sup>[3]</sup>は, 平均曲率フローを用いた拡散方程式に基づく手法を提案した. しかし, これらのスムージング手法は, スムージング後に元のメッシュモデルの特徴稜線が失われるという問題がある.

このような問題を解決するために近年, Jones ら<sup>[4]</sup>や Fleishman ら<sup>[5]</sup>により *bilateral filtering* を用いたメッシュスムージング手法が提案されている. しかし, これらの手法は元のメッシュモデルの特徴稜線は保存できるが, 特徴稜線を稜線方向に滑らかにする効果はない. つまり, 工業製品のように特徴稜線と特徴稜線を境界として持つフェイスから構成される表面を測定して得られたメッシュモデルに対しては, 特徴稜線とフェイスを別個にスムージングする必要があると考えられる

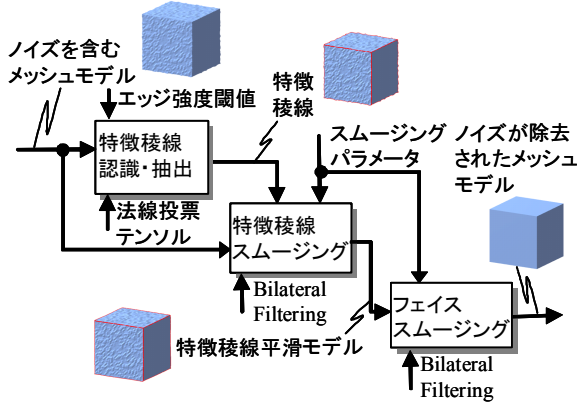


図1 提案手法の概要

が、そのようなスムージング手法はこれまでに提案されていない。

そこで本研究では、測定ノイズを含む工業製品のメッシュモデルからまず特徴稜線を認識し、bilateral filteringにより特徴稜線とフェイスの測定ノイズをそれぞれ除去する、特徴稜線認識型メッシュスムージング手法を提案する。

## 2. 特徴稜線認識型メッシュスムージング

### 2.1 提案手法の概要

図1に提案手法の概要を示す。初めにノイズを含むメッシュから、各頂点に対する法線投票テンソルの固有値解析により特徴稜線を抽出する。次に、抽出された特徴稜線に対し稜線用 bilateral filtering を適用し、稜線方向のスムージングを行う。最後にこの特徴稜線を保存するように、ノイズを含むメッシュに対しフェイス用 bilateral filtering を適用し、スムージングを行う。これら全ての手法はメッシュの接続性を必要としないため、ポリゴンスープモデルにも適用可能である。

### 2.2 特徴稜線認識・抽出

ノイズを含むメッシュモデルから特徴稜線を認識するために、法線投票テンソルの固有値解析による特徴稜線抽出手法<sup>[6]</sup>を用いる。メッシュ頂点 $i$ に対する法線投票テンソル $\mathbf{T}_i$ は $\mathbf{T}_i = \sum_{t \in N(i)} \mu_t \mathbf{n}'_t \mathbf{n}'_t^T$ で計算される。ここで、 $N(i)$ は頂点 $i$ の近傍メッシュ三角形集合、 $\mu_t$ は近傍メッシュ三角形 $t$ の重心点 $\mathbf{c}_t$ と $i$ の座標 $\mathbf{p}_i$ 間の距離に依存する重みで、 $\mu_t = A_t/A_M \exp(-\|\mathbf{c}_t - \mathbf{p}_i\|/\sigma)$ 、ただし、 $A_t$ は近傍メッシュ三角形 $t$ の面積、 $A_M$ はメッシュ三角形の最大面積、 $\sigma$ は近傍を定義する立方体の1辺の長さの1/3である。 $\mathbf{n}'_t$ は $i$ における投票用法線と呼ばれ、 $\mathbf{n}'_t = 2(\mathbf{n}_t \cdot \mathbf{w}_t) \mathbf{w}_t - \mathbf{n}_t$ で算出される。ここで、 $\mathbf{w}_t = \mathbf{m}_t / \|\mathbf{m}_t\|$ 、 $\mathbf{m}_t = \{(\mathbf{c}_t - \mathbf{p}_i) \times \mathbf{n}_t\} \times (\mathbf{c}_t - \mathbf{p}_i)$ である。得られた法線投票テンソルから固有値 $v_1, v_2, v_3$  ( $v_1 \geq v_2 \geq v_3$ )とそれぞれに対応する固有ベクトル $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ を求める。これらの固有値と固有ベクトルからメッシュ頂点 $i$ が帰属する位相要素(フェイス、特徴稜線、コーナー)を推定する。図2に示すように、フェイ

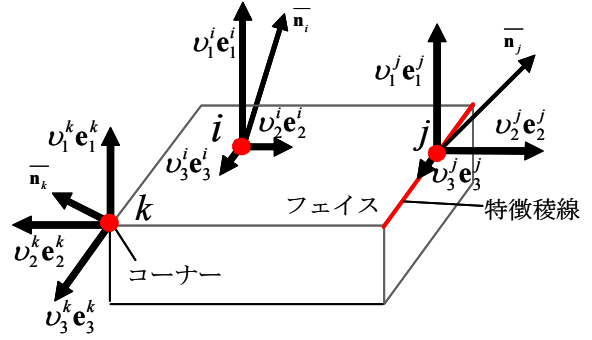


図2 頂点が帰属する位相要素による固有ベクトルの違い

スに帰属する頂点 $i$ に対する平均投票用法線 $\bar{\mathbf{n}}_i$ では、そのフェイスに垂直な固有ベクトルの方向成分が支配的になる。特徴稜線に帰属する頂点 $j$ に対する平均投票用法線 $\bar{\mathbf{n}}_j$ では、特徴稜線に垂直な2つの固有ベクトルの方向成分が支配的になる。同様に、コーナーに帰属する頂点 $k$ に対する平均法線 $\bar{\mathbf{n}}_k$ については、3つの固有ベクトルの方向成分が均等になる。これらの性質を利用し、頂点 $i$ のエッジ強度 $s_i$ を式(1)のように定義する。 $s_i$ は、特徴稜線やコーナーに帰属する頂点では1に近い値を持ち、フェイスに帰属する頂点は0に近い値を持つ。

$$s_i = \begin{cases} 1 & |\bar{\mathbf{n}}_i \cdot \mathbf{e}_1| < \delta \\ (v_3 > \alpha(v_1 - v_2) \wedge v_3 > \beta(v_2 - v_3)) / (v_2 - v_3) / v_1 & \text{Otherwise} \end{cases} \quad (1)$$

ここで、 $\bar{\mathbf{n}}_i = \sum_{t \in N(i)} \mu_t \mathbf{n}'_t$ は平均投票用法線、 $\delta, \alpha, \beta$ は正の実数であり、本研究では実験的に0.1, 0.9, 0.9と定めた。

全メッシュ頂点に対するエッジ強度を求めた後、エッジ強度があるしきい値 $s_i$ 以上の頂点を持つ特徴稜線方向の固有ベクトルと、この頂点が端点となるメッシュエッジの方向ベクトルとの内積が0.9以上のメッシュエッジを特徴稜線として抽出する。

### 2.3 メッシュスムージング手法

Jonesらが提案したbilateral filteringによるメッシュスムージング手法<sup>[4]</sup>は、特徴稜線をある程度保存するようなスムージングが可能である。しかし、この手法は元のメッシュモデルの特徴稜線は保存できるが、特徴稜線を稜線方向に滑らかにする効果はない。そこで本手法では、Jonesらが提案したメッシュ三角形群に対するbilateral filtering手法を、抽出された特徴稜線を構成するメッシュエッジ群にも適用可能なような稜線用 bilateral filteringへ応用する。さらにフェイス用 bilateral filteringも、特徴稜線をより保存するような手法へと改良する。

#### 2.3.1 Bilateral filtering

Bilateral filteringはTomasiとManduchi<sup>[7]</sup>によって提案された、従来の異方性拡散に基づく画像フィルタリン

グに代わる，境界保存型フィルタリング手法である．  
Bilateral filtering は式(2)のように表される．

$$E(p) = \frac{1}{k(p)} \sum_{q \in \Omega} I(q) f(q-p, \sigma_f) g(I(q)-I(p), \sigma_g) \quad (2)$$

$$k(p) = \sum_{q \in \Omega} f(q-p, \sigma_f) g(I(q)-I(p), \sigma_g)$$

ここで  $E(p)$  はピクセル  $p$  の出力濃淡値， $\Omega$  は近傍ピクセル集合， $I(q)$  は  $q$  の濃淡値， $f$  と  $g$  はそれぞれピクセル間の距離，濃淡差を確率密度変数とし， $\sigma_f$ ， $\sigma_g$  を標準偏差とする Gauss 関数である．関数  $g$  は edge-stopping 関数と呼ばれ，濃淡差に依存させることにより，濃淡差の大きい近傍ピクセルのフィルタリングへの寄与を少なくする．これにより，フィルタリングによる境界部分の損失が抑えられる．標準偏差  $\sigma_f$ ， $\sigma_g$  はスケールパラメータであり， $\sigma_f$  を大きくするとスムージングの効果は増大するが境界部分が損失する． $\sigma_g$  を大きくするとスムージングの効果は減少するが，境界部分はよく保存される．

### 2.3.2 特徴稜線スムージング

特徴稜線スムージングは以下の2つの処理からなる．

#### 1) 特徴稜線の方向ベクトルのスムージング

まず，特徴稜線を構成するメッシュエッジの方向ベクトルを bilateral filtering により稜線方向にスムージングする．これは，2)の特徴稜線を構成するメッシュ頂点のスムージングの際，その頂点近傍のメッシュエッジの方向ベクトルにより定義される投影点を求めるが，各メッシュエッジの方向ベクトルが特徴稜線に沿っていないと良いスムージング効果が得られないためである．

特徴稜線を構成する，あるメッシュエッジ  $e$  の単位方向ベクトル  $\mathbf{d}_e$  に対する bilateral filtering は式(3)で表される．

$$\mathbf{d}'_e = \frac{1}{k_m(e)} \sum_{e^* \in Ne(e)} \mathbf{d}_{e^*} l_{e^*} f(\|\mathbf{c}_{e^*} - \mathbf{c}_e\|, \sigma_{mef}) g(\|\mathbf{d}_{e^*} - \mathbf{d}_e\|, \sigma_{meg})$$

$$k_m(e) = \sum_{e^* \in Ne(e)} l_{e^*} f(\|\mathbf{c}_{e^*} - \mathbf{c}_e\|, \sigma_{mef}) g(\|\mathbf{d}_{e^*} - \mathbf{d}_e\|, \sigma_{meg}) \quad (3)$$

ここで， $\mathbf{d}'_e$ ， $Ne(e)$ ， $\mathbf{c}_e$  はそれぞれ  $e$  の，スムージング後の方向ベクトル，近傍のメッシュエッジ集合，エッジ中点であり， $\mathbf{d}_{e^*}$ ， $l_{e^*}$ ， $\mathbf{c}_{e^*}$  はそれぞれ  $e^*$  の方向ベクトル，線分長，エッジ中点で， $k_m(e)$  は正規化因子である． $f$  は  $e$  と  $e^*$  間の距離を確率密度変数， $\sigma_{mef}$  を標準偏差とする Gauss 関数である． $g$  は  $e$  と  $e^*$  の方向ベクトルの差の大きさを確率密度変数， $\sigma_{meg}$  を標準偏差とする Gauss 関数である．式(3)の edge-stopping 関数  $g$  は，方向ベクトルの差が大きいとき小さな値をとるため，特徴稜線同士が交わるコーナー付近において，ある稜線の方向ベクトルのスムージングの際に他の特徴稜線の方向ベクトルの与える影響を減少する．また，

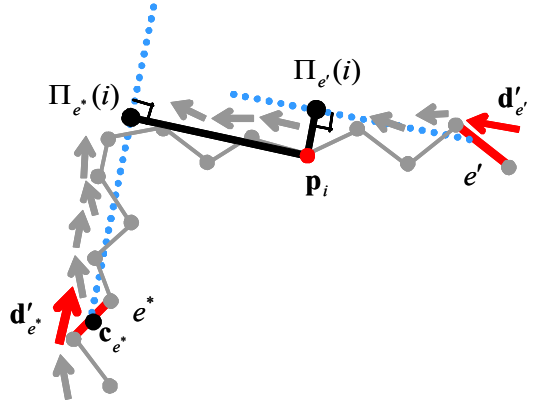


図3 頂点  $i$  に対する  $\Pi_{e^*}(i)$

方向ベクトルの差の大きさが1を超えるとき，その近傍メッシュエッジは同一特徴稜線上のメッシュエッジではないと判断し，edge-stopping 関数に十分に大きな数値を与えることでその近傍メッシュエッジのスムージングへの影響を完全に無くす．これにより，コーナー付近のメッシュエッジの方向ベクトルがより適切にスムージングされる．

#### 2) 特徴稜線を構成する頂点のスムージング

次に，特徴稜線上のメッシュ頂点  $i$  を，式(4)の bilateral filtering によりスムージングする．

$$\mathbf{p}'_i = \frac{1}{k(i)} \sum_{e^* \in Ne(i)} \Pi_{e^*}(i) l_{e^*} f(\|\mathbf{c}_{e^*} - \mathbf{p}_i\|, \sigma_{ef}) g(\|\Pi_{e^*}(i) - \mathbf{p}_i\|, \sigma_{eg})$$

$$k(i) = \sum_{e^* \in Ne(i)} l_{e^*} f(\|\mathbf{c}_{e^*} - \mathbf{p}_i\|, \sigma_{ef}) g(\|\Pi_{e^*}(i) - \mathbf{p}_i\|, \sigma_{eg}) \quad (4)$$

ここで， $\mathbf{p}'_i$ ， $Ne(i)$  はそれぞれ  $i$  の，スムージング後の座標，近傍のメッシュエッジ集合であり， $k(i)$  は正規化因子である． $\Pi_{e^*}(i)$  は図3に示すように， $e^*$  の1)でスムージングされた方向ベクトル  $\mathbf{d}'_{e^*}$  を持ち点  $\mathbf{c}_{e^*}$  を通る直線への  $i$  の投影点である． $f$  は  $i$  と  $e^*$  間の距離を確率密度変数， $\sigma_{ef}$  を標準偏差とする Gauss 関数である． $g$  は  $i$  と  $\Pi_{e^*}(i)$  の距離を確率密度変数， $\sigma_{eg}$  を標準偏差とする Gauss 関数である．図3に示されるように， $i$  と  $\Pi_{e^*}(i)$  の距離は，近傍メッシュエッジが同一特徴稜線上にないときは大きくなる．従って edge-stopping 関数  $g$  を  $i$  と  $\Pi_{e^*}(i)$  の距離に依存させることで，スムージングにおいて，ある特徴稜線上のメッシュエッジは，コーナーをまたいだ位置にある別の特徴稜線上のメッシュエッジの影響を受けにくくなる．

### 2.3.3 フェイススムージング

フェイススムージングは以下の2つの処理からなる．

#### 1) メッシュ三角形の法線のスムージング

まず，メッシュモデルの各メッシュ三角形の法線を bilateral filtering によりスムージングする．これは，特徴稜線スムージングと同様に，2)のメッシュ頂点のス

ムーピングの際、近傍のメッシュ三角形の法線により定義される投影点を求めるが、各メッシュ三角形の法線がフェイスごとにそろっていないと良いスムージング効果が得られないためである。

メッシュモデルの任意三角形 $t$ の法線 $\mathbf{n}_t$ に対する bilateral filtering は式(5)で表される。

$$\begin{aligned} \mathbf{n}'_t &= \frac{1}{k_m(t)} \sum_{t^* \in Nt(t)} \mathbf{n}_{t^*} a_{t^*} f(\|\mathbf{c}_{t^*} - \mathbf{c}_t\|, \sigma_{mf}) g(\|\mathbf{n}_{t^*} - \mathbf{n}_t\|, \sigma_{mg}) \\ k_m(t) &= \sum_{t^* \in Nt(t)} a_{t^*} f(\|\mathbf{c}_{t^*} - \mathbf{c}_t\|, \sigma_{mf}) g(\|\mathbf{n}_{t^*} - \mathbf{n}_t\|, \sigma_{mg}) \end{aligned} \quad (5)$$

ここで、 $\mathbf{n}'_t$ 、 $Nt(t)$ 、 $\mathbf{c}_t$ はそれぞれ $t$ の、スムージング後の法線、近傍のメッシュ三角形集合、重心点であり、 $\mathbf{n}_{t^*}$ 、 $a_{t^*}$ 、 $\mathbf{c}_{t^*}$ はそれぞれ $t^*$ の法線、面積、重心点で、 $k_m(t)$ は正規化因子である。 $f$ は $t$ と $t^*$ の重心点間の距離を確率密度変数、 $\sigma_{mf}$ を標準偏差とする Gauss 関数である。 $g$ は $t$ と $t^*$ の法線の差の大きさを確率密度変数、 $\sigma_{mg}$ を標準偏差とする Gauss 関数である。

式(5)の edge-stopping 関数 $g$ は、法線の差が大きいきり小さな値をとるため、フェイス同士が交わる特徴稜線付近において、あるフェイスの法線のスムージングの際に他のフェイスの法線の影響を減少する。また、法線の差の大きさが1を超えるとき、その近傍メッシュ三角形は同一フェイス上のメッシュ三角形ではないと判断し、edge-stopping 関数に十分に大きな数値を与えることでその近傍メッシュ三角形のスムージングへの影響を完全に無くす。これにより、特徴稜線付近のメッシュ三角形の法線がより適切にスムージングされる。

## 2) メッシュ頂点のスムージング

次にメッシュモデルのメッシュ頂点を bilateral filtering によりスムージングする。メッシュ頂点 $i$ の bilateral filtering は式(6)で表される。

$$\begin{aligned} \mathbf{p}'_i &= \frac{1}{k(i)} \sum_{t^* \in Nt(i)} \Pi_{t^*}(i) a_{t^*} f(\|\mathbf{c}_{t^*} - \mathbf{p}_i\|, \sigma_f) g(\|\Pi_{t^*}(i) - \mathbf{p}_i\|, \sigma_g) \\ k(i) &= \sum_{t^* \in Nt(i)} a_{t^*} f(\|\mathbf{c}_{t^*} - \mathbf{p}_i\|, \sigma_f) g(\|\Pi_{t^*}(i) - \mathbf{p}_i\|, \sigma_g) \end{aligned} \quad (6)$$

ここで、 $\mathbf{p}'_i$ 、 $Nt(i)$ はそれぞれ $i$ の、スムージング後の座標、近傍のメッシュ三角形集合であり、 $k(i)$ は正規化因子である。 $\Pi_{t^*}(i)$ は図4に示すように $t^*$ の1)でスムージングされた法線 $\mathbf{n}'_{t^*}$ を持ち、 $\mathbf{c}_{t^*}$ を通る平面への $i$ の投影点である。 $f$ は $i$ と $t^*$ の重心点の距離を確率密度変数、 $\sigma_f$ を標準偏差とする Gauss 関数である。 $g$ は $i$ と $\Pi_{t^*}(i)$ の距離を確率密度変数、 $\sigma_g$ を標準偏差とする Gauss 関数である。図4に示されるように、 $i$ と $\Pi_{t^*}(i)$ の距離は、近傍メッシュ三角形が同一フェイス上にならないときは大きくなる。従って edge-stopping 関数

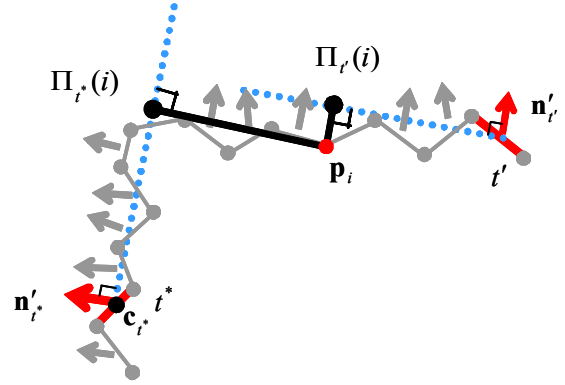


図4 頂点 $i$ に対する $\Pi_{t^*}(i)$

$g$ を $i$ と $\Pi_{t^*}(i)$ の距離に依存させることで、メッシュ頂点は、スムージングにおいて、特徴稜線をまたいだ位置にあるメッシュ三角形の影響を受けにくくなる。

## 3. スムージングパラメータの設定

Bilateral filtering によるスムージングの効果は2つの Gauss 関数の標準偏差 $\sigma$ によって調整される。本手法では、Jones らの提案に基づき、特徴稜線スムージングの方向ベクトルスムージングは $\sigma_{mf} = a/2 \times |e|$ 、 $\sigma_{mg} = 0.6$ 、頂点処理は $\sigma_{ef} = a \times |e|$ 、 $\sigma_{eg} = b \times |e|$ 、フェイススムージングの法線スムージングは $\sigma_{mf} = c/2 \times |e|$ 、 $\sigma_{mg} = 0.6$ 、頂点処理は $\sigma_f = c \times |e|$ 、 $\sigma_g = d \times |e|$ とした。ただし $a, b, c, d$ はスムージングパラメータで対象のモデルに依存して定める正の実数、 $|e|$ はモデル内のメッシュエッジの平均辺長である。

## 4. 検証用モデルのスムージング結果

図5(a)の立方体を表すメッシュモデル(面分数: 10240)に、平均がゼロ、偏差、振幅がモデルの平均辺長 $|e|$ の1/5の Gaussian ノイズを加えたモデル(図5(b))に本手法を適用した結果を図5(c)-(f)に示す。特徴稜線抽出の法線投票では、メッシュ頂点 $i$ を中心とする1辺が $3|e|$ の立方体内に含まれるメッシュ三角形群を、頂点 $i$ の近傍メッシュ三角形群として利用した。エッジ強度しきい値 $s_t = 0.337$ で、まず、スムージングパラメータ $(a, b) = (4, 1)$ で特徴稜線スムージングを1回を行い、さらに $(c, d) = (4, 1)$ と $(2, 1)$ でフェイススムージングを2回行った。図5(d)より、特徴稜線が稜線方向に滑らかになったことが確認できる。また、図5(f)より特徴稜線を保存しながらフェイスのノイズが除去されていることが確認できる。

次に、図6(a)のFandiskを表すメッシュモデル(面分数: 51784)に、平均がゼロ、偏差、振幅がモデルの平均辺長 $|e|$ の1/5の Gaussian ノイズを加えたモデル(図6(b))に本手法を適用した結果を図6(c)-(f)に示す。エッジ強度しきい値 $s_t$ は0.251で、まず、スムージングパラメータ $(a, b) = (4, 1)$ で特徴稜線スムージングを1回

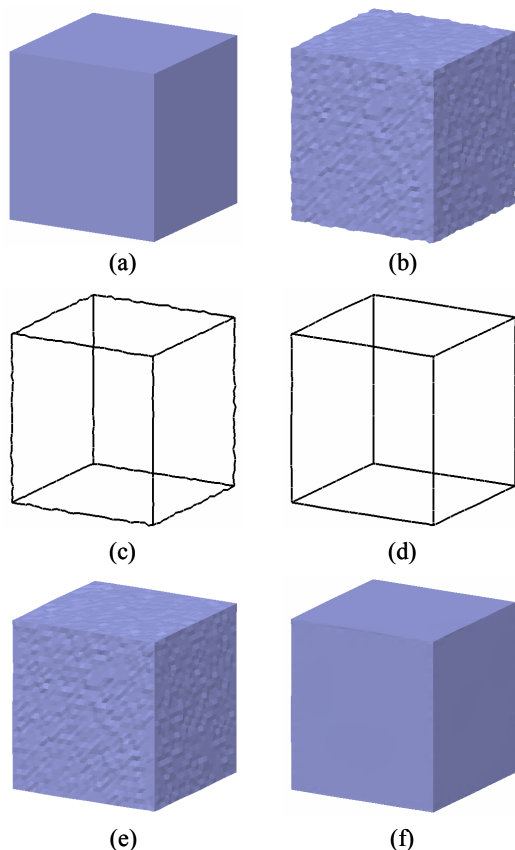


図5 Cubelに対する本手法の適用結果. (a) 元のモデル. (b) ノイズを加えたモデル. (c) 抽出した特徴稜線. (d),(e) 特徴稜線スムージング結果. (f) フェイススムージング結果.

行い、さらに(c,d) = (4,1) と (2,1) でフェイススムージングを2回行った. 図 6(d)より、特徴稜線が稜線方向に滑らかになったことが確認できる. また、図 6(f)より特徴稜線を保存しながらフェイスのノイズが除去されていることが確認できる.

## 5. 測定モデルのスムージング結果

IT 機器筐体のデザインモックアップを comet400 によりスキャンして得られたメッシュモデル (面分数: 612013, 頂点数: 305992, 図 7(a)) に本手法を適用した結果を図 7(b),(c),(d)に示す. 特徴稜線抽出の法線投票では、メッシュ頂点 $i$ を中心とする1辺が $9|e|$ の立方体内に含まれるメッシュ三角形群を、頂点 $i$ の近傍メッシュ三角形群として利用した. エッジ強度しきい値 $s_e$ は 0.102 とし、まず、スムージングパラメータ $(a,b) = (10,1)$ で特徴稜線スムージングを行い、さらに $(c,d) = (2,1)$ でフェイススムージングを1回行った. 各特徴稜線は、1本の連結されたメッシュエッジ群から構成されることが望ましいが、図 7(b)に見られるように、特徴稜線が存在すると思われるメッシュ部分に連結されていないメッシュエッジ群が多数抽出された. しかし、特徴稜線スムージングによりこれらが1本の

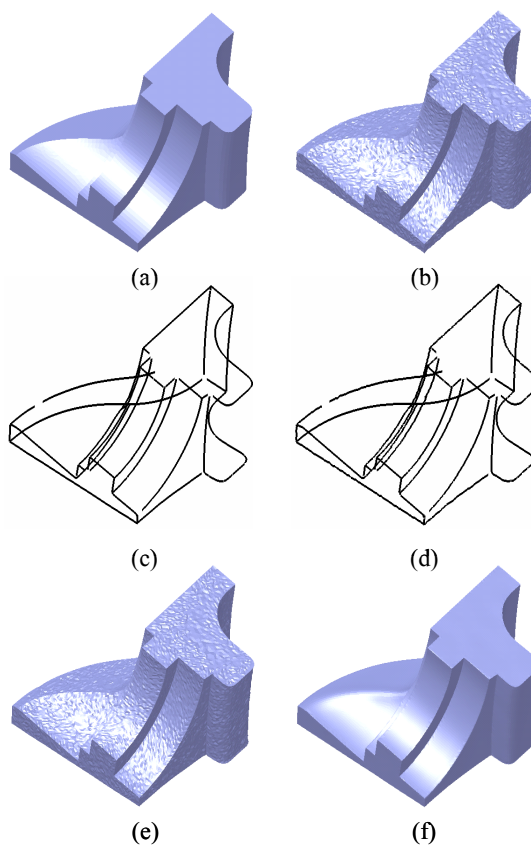


図6 Fandiskに対する適用結果. (a) 元のモデル. (b) ノイズを加えたモデル. (c) 抽出した特徴稜線. (d),(e) 特徴稜線スムージング結果. (f) フェイススムージング結果.

連結されたメッシュエッジ群を構成するように集約された (図 7(c)). また、図 7(c)より、特徴稜線が稜線方向に滑らかになったことが確認でき、さらに、図 7(d)より特徴稜線を保存しながらフェイスのノイズが除去されていることが確認できる.

## 6. まとめ

本研究では、法線投票テンソルの固有値解析と改良した bilateral filtering を用いた特徴稜線認識型メッシュスムージング手法を提案した. また、簡単なモデルと実際のスキャンモデルに本手法を適用し、その効果を確認した. 今後の課題として、より頑強な特徴稜線抽出手法を検討することと、スムージング効果の定量評価が挙げられる.

### 【参考文献】

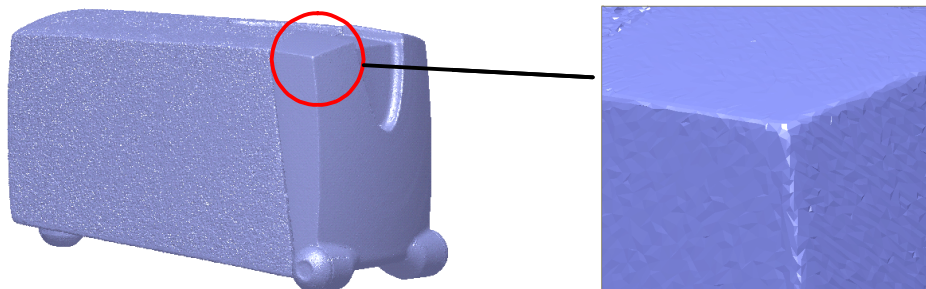
- [1] G.Taubin : A signal processing approach to fair surface design, Proc.SIGGRAPH'95, pp.351-358,1995
- [2] L.Kobbelt, S.Campagna, J.Vorsatz, and H.-P.Seidel: Interactive MultiResolution Modeling on Arbitrary Meshes, Proc. SIGGRAPH'98, pp105-115,1998
- [3] M.Desbrun, M.Meyer, P.Schroder and A.H.Barr: Implicit

fairing of irregular meshes using diffusion and curvature flow.  
Proc. SIGGRAPH'99, pp.317-324, 1999

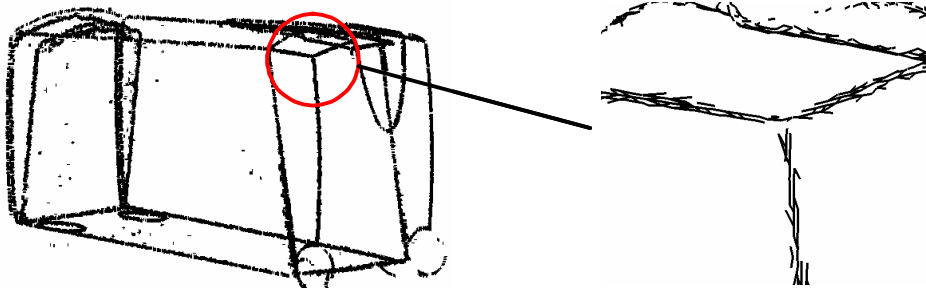
- [4] T. R. Jones, F. Durand, M. Desbrun : Non-Iterative, Feature-Preserving Mesh Smoothing, Proc. SIGGRAPH2003, pp. 943-949, 2003
- [5] S. Fleishman, I. Drori, D. Cohen-Or : Bilateral Mesh Denoising, SIGGRAPH2003, pp.950-953, 2003
- [6] Y. Sun, D.L. Page, J.K. Paik, A. Koschan and M.A. Abidi : Triangle mesh-based edge detection and its application to surface segmentation and adaptive surface smoothing,

Proc. IEEE ICIP2002, pp.825-828, 2002

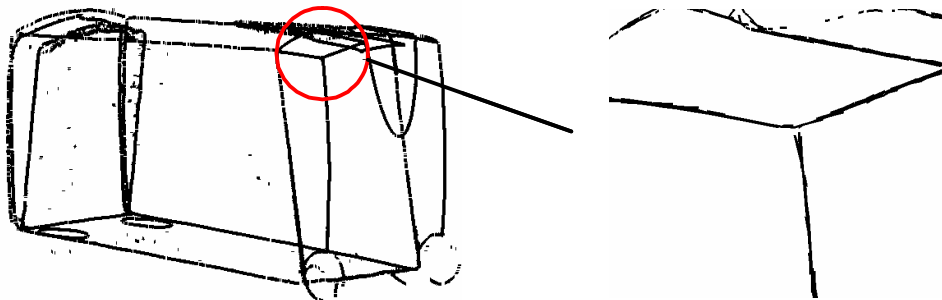
- [7] Tomasi, C, Manduchi, R: Bilateral Filtering for Gray and Color Images, In Proc IEEE Int. Conf. on Computer Vision, pp.836-846, 1998.



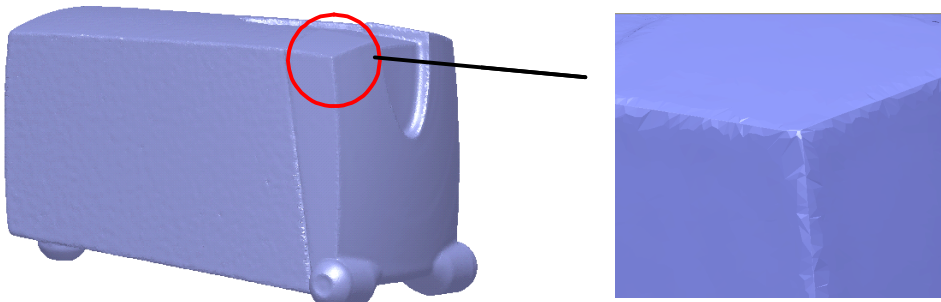
(a) スキャンモデル



(b) 抽出した特徴稜線



(c) 特徴稜線スムージング結果



(d) フェイススムージング結果

図7 IT機器筐体のデザインモックアップに対する適用結果