

コンピュータグラフィックスにおけるエッジ強調描画法†

齋藤 隆文** 高橋 時市郎**

コンピュータグラフィックスを用いたわかりやすい絵の作成の一環として、エッジ（輪郭線および稜線）を強調描画するための一手法を提案する。人間の描くイラスト画では、強調や省略を施すことによって、形状をわかりやすく描写している。しかし、わかりやすさには絶対的な評価基準がない。このため、コンピュータで描画する場合にも、目的に応じた適切な強調法を決定するには、オペレータの試行錯誤が不可欠である。従来手法では、試行錯誤のたびにすべて処理をやり直す必要があったり、適応範囲に限られるなどの欠点があった。本論文で提案する手法は、隠面消去処理の副産物として得られる画素ごとの奥行き値を距離画像として用い、これに微分オペレータを適用するなどの画像処理を行って、輪郭線や稜線の画像を生成するものである。これによって、人為的な強調処理は、幾何的あるいは物理的要因に基づくCG処理から切り離され、後処理として実現されるため、適切な強調画を試行錯誤により効率よく作成することができる。本手法はさらに、複雑な曲面形状や、反射、屈折物体に対しても適用できる。

1. はじめに

3次元の物体形状を視覚的に伝達するために、人間の手で描かれたイラスト画が広く使われている。通常イラスト画においては、物体を写実的に描くのではなく、さまざまな強調や省略を施すことにより、形状をわかりやすく表現することが行われる^{1),2)}。なかでも物体のエッジ（輪郭線および稜線）の強調描画が効果的である。エッジは、現実には色やテクスチャの不連続線として見えるにすぎないが、これは黒線や白線を重ねて強調することで、よりわかりやすい描写が可能となる。また、対象によっては、エッジを線画で描いただけで、わかりやすい絵になることも、よく知られている。しかしながら、従来のコンピュータグラフィックス(CG)でのレンダリング手法の多くは、写実性の追究を目的としたものである。これに比べて、見る人の理解を促進させるような強調描画法は、あまりかえりみられていない³⁾。

CGで強調画を描く際の最大の問題点は、種々の強調法の中から最適な組合せを決定することにある。なぜなら、強調画におけるわかりやすさの評価は、描画の対象や目的、さらには見る人の好みにまで依存し、定量的に定義できないからである。それゆえ、適切な強調画を自動的に作成することは、現時点では不可能に近く、多くの場合は人間の判断が介入せざるを得ない。わかりやすい描画を目的とするCGシステムにおいては、どの強調法を選択し組み合わせるかはオペ

レータの試行錯誤に委ねるかわりに、描画法の変更に伴う再描画を素早く行えることが望ましい。

本論文では、3次元形状データをもとに、これをCGでわかりやすく表現するための、エッジ強調描画法を提案する。提案手法の特徴は、通常の画像生成の過程で求まる視点からの奥行き値から作られる距離画像を利用することである。これに画像処理を施して、輪郭線、稜線を抽出し、強調処理を行う。提案手法は、これまでの光学現象を忠実にシミュレートすることで得られる効果と、意図的に強調を加えることによる効果の両立が可能である。しかも、両効果は完全に分離して扱うことができる。強調処理は、距離画像に対する画像処理の組合せだけで実現できるため、目的に応じた強調描画法の試行錯誤が容易になり、作画作業の効率化を図ることができる。

本論文の構成は以下のとおりである。まず2章では、従来手法の組合せによるエッジ強調法を紹介し、その問題点を指摘する。3、4章では、提案する新手法について説明し、適用例を5章に示す。最後に、6章で考察を述べる。

2. 従来手法によるエッジ強調とその問題点

従来技術の組合せだけでも、3次元形状データからエッジ強調画を作成する方法はいくつか考えられる。しかし、試行錯誤を前提とした処理の効率性や、種々の強調法への適応性などの点で、満足できる方法はない。以下に、各種方法およびその問題点を示す。

(1) ペイントシステムでの修正

作成したCG画像に対し、ペイントシステムを使って会話的に修正を加えれば、エッジ強調なども自由に

† Edge Enhanced Drawing in Computer Graphics by TAKAFUMI SAITO and TOKIICHIRO TAKAHASHI (Autonomous Robot Systems Laboratory, NTT Human Interface Laboratories).

** NTT ヒューマンインタフェース研究所知能ロボット研究部

行える。通常のペイントシステムのほか、テクニカルイラストレーションのための各種ツールを備えたシステム²⁾も研究されている。人間が介入する分、目的に応じたきめ細かい強調が可能である反面、全体に均一な処理を行うことができず、処理の再現性も低いという弱点がある。また、複雑な画像に対しては手間がかかりすぎる。特に、アニメーションの一連のシーンなどの、類似する多数の画像に対して同一の強調を行うには、非現実的な方法である。

(2) 輝度画像へのエッジ強調フィルタの適用

画像処理の分野では良く知られているように、画像に適当な微分オペレータを作用させると、エッジは強調される^{4),5)}。この方法は、CGで作成した輝度画像に対しても適用できる。ただし、強調されるエッジはあくまでも輝度が不連続に変化する部分に限られ、輪郭線や稜線には必ずしも対応しない。たとえば同色の平行な二面が重なって見える場合のように、輝度差のない輪郭線は強調できない。現実には、このような部分こそ強調処理を施してわかりやすくしたい。

(3) 線画との重ね合わせ

3次元形状をワイヤフレームで定義し、隠線消去を施して線画で表示することは、CGでは古くから行われている。輪郭線および稜線を線画で適切に描くことによって、3次元形状をわかりやすく表示する研究も、試みられている⁶⁾。このようにして作成した線画と、シェーディングを行った面画とを重ねれば、面画による強調画が得られる。しかしこの場合、隠線消去と隠面消去という二つの似て非なる処理を同時に行う必要があり、処理効率は良くない。また、複雑な曲面の輪郭線や交線を効率よく求め、隠線処理することは、一般には困難である。

(4) 面取りされた稜線のハイライト表示⁷⁾

現実の物体では、ほとんどの稜線が面取りされており、そこに広い範囲の光源が映り込んで、しばしばハイライトを生じる。これは、現実の光学条件のもとで稜線が強調されて見える例である。したがって、この現象をCGでシミュレートして表示すれば、写実性のみならず、わかりやすさも向上する。しかし、あくまでも与えられた光源下における正確なハイライトを計算するものであるため、すべての稜線に対して作者が意図するようなハイライトが現れるように、光源を配置することは、しばしば大きな困難を伴う。

3. 画像処理を用いた輪郭線、稜線の描画

本章では、距離画像に対して画像処理を施すことにより、輪郭線および稜線を描画し、あるいは強調する方法を提案する。この方法は、輪郭線および稜線が、視点からの距離あるいはその1次微分量が不連続に変化する部分に対応する、という原理に基づいたものである。しかも、視点からの距離の算出は、通常の隠面消去処理において内部計算として必ず行われるので、距離計算に伴うオーバーヘッドがない、という利点がある。図1に処理の流れを示す。以下、処理の詳細について説明する。

3.1 距離画像の作成

距離画像は、視点からの奥行き値をピクセルごとに求めて2次元配列にしたものである。CGでは隠面消去の際の前後判定に距離計算が必須であるため、処理の副産物として容易に得られる。光線追跡法や z バッファ法では、奥行き値をピクセルごとに求めており、これを2次元配列としてそのまま出力すればよい。スキャンライン法でも、面の境界線や交線での奥行き値は求めているから、その他のピクセルは線形補間などで容易に計算できる。

透視投影法で作画する場合、奥行き値に対しても透視変換を施すことが必要である。この場合、視点座標系における z 座標(奥行き方向)の値を z_0 とすると、距離画像での奥行き値 z は、一般に次式により求め

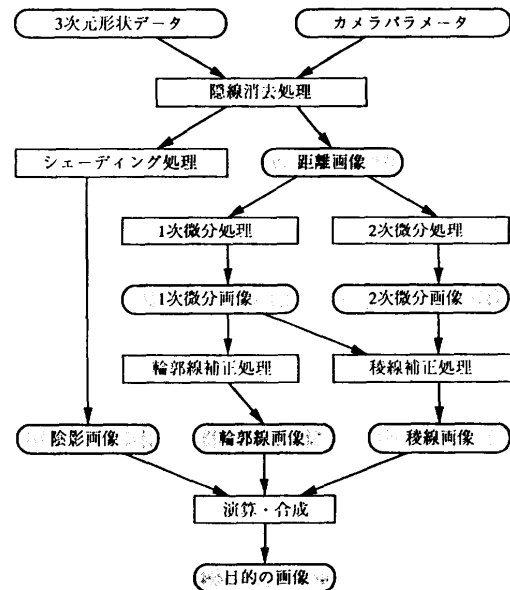


図1 エッジ強調画の作成方法

Fig. 1 Drawing method of edge enhanced pictures.

られる⁸⁾：

$$z_s = \alpha + \beta/z_v, \tag{1}$$

ただし、 α, β は定数である。式(1)は、同一直線上において、スクリーン上での奥行き値の線形性を保つための必要十分条件である。ここでは、後の処理を考慮して、特に次式を推奨する：

$$z_s = d^2/(wx_v). \tag{2}$$

ここで、 d は視点からスクリーンまでの距離、 w はスクリーン上での1ピクセルの幅である(図2)。式(2)を用いる利点は、画像の中央部において、なめらかな物体表面上での距離の1次微分値が物体面の傾きに一致する(距離 z_s や画角、解像度などに依存しない)ことである。なお、式(2)では、奥行き値の大小は前後関係とは完全に逆転し、無限遠点が0となる。

図3(a)は、既存のCG技術(z バッファ法)で作成した陰影画像の例である。(b)は、このとき副産物として得られる距離画像である。

3.2 微分処理による輪郭線、稜線の描出

輪郭線は、奥行き値の不連続線である。したがって、距離画像に1次微分オペレータを作用させることによって抽出できる。この処理には、Sobel オペレータ：

$$g = (|A + 2B + C - F - 2G - H| + |C + 2E + H - A - 2D - F|)/8 \tag{3}$$

あるいは、Prewitt オペレータ：

$$g = (|A + B + C - F - G - H| + |C + E + H - A - D - F|)/6 \tag{4}$$

など、種々の1次微分オペレータ^{4),5)}の中から選択して使用する。ここに、 $A \sim H, X$ は、図4に示した近傍画素の値である。また、式(3)、(4)では、 g の値

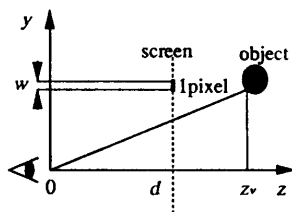


図2 奥行き値の透視変換
Fig. 2 Perspective depth.

A	B	C
D	X	E
F	G	H

図4 近傍画素
Fig. 4 Neighboring pixels.

が1次微分値すなわち面の傾きにはほぼ一致するように、正規化してある。図3(c)は、(b)に式(3)のSobel オペレータを適用したものである。

稜線は、法線方向すなわち距離の1次微分量の不連続線である。したがって、距離画像に2次微分オペレータを作用させることによって抽出できる。この処理には、Laplacian：

$$l = (4X - B - D - E - G) \tag{5}$$

あるいは、これを变形したオペレータ

$$l = (8X - A - B - C - D - E - F - G - H)/3 \tag{6}$$

などの、種々の2次微分オペレータ^{4),5)}の中から選択して使用する。図3(d)は、(b)に式(6)のオペレータを適用したものである。

3.3 補正処理

輪郭線および稜線は、前述の微分オペレータによって抽出できるが、それだけでは描画用としては好ましくない。たとえば、微分量の大小がそのまま線の輝度となって現れるため、輪郭線の濃さが著しく不揃いになる。また、なめらかな面であっても視線に平行に近い場合は、微分値が大きいために、広い面全体が輪郭線として抽出されてしまう。さらに、2次微分オペレータでは、稜線のみならず輪郭線が正負の2重線となって現れる。

これらの問題は、近傍での微分画像の最大値、最小値を用いて、正規化などの補正を行うことによって、かなりの部分が解決できる。いま、ある画素における距離の1次微分値、2次微分値を、それぞれ g, l とする。まず、この画素の近傍(たとえば図4に示した9画素)の中での1次微分の最大値 g_{max} および最小値 g_{min} を求める。このとき、たとえば次式を用いて輝度値 p を求めることにより、輪郭線の輝度は正規化される：

$$p = \begin{cases} (g - g_{min}) / (g_{max} - g_{min}) & (g_{max} - g_{min} \geq k_p \text{ のとき}) \\ (g - g_{min}) / k_p & (g_{max} - g_{min} < k_p \text{ のとき}) \end{cases} \tag{7}$$

ここで、 k_p は輪郭線と傾斜とを判別するための定数である。適切な k_p の値は、対象形状や描画目的によって異なる。図3(c)に式(7)の補正を加えたものを、(e)に示す。この例では、 $k_p = 10$ としている。

稜線については、たとえば次式：

$$e = \begin{cases} l & (g_{max} \leq k_l \text{ のとき}) \\ l / (g_{max} / k_l)^2 & (g_{max} > k_l \text{ のとき}) \end{cases} \tag{8}$$

によって輝度値 e を求めることにより、輪郭線を消去することができる。 k_i は、輪郭線消去を行うための1次微分の限界値であり、適切な値は対象形状や描画目的によって異なる。図3(d)に式(8)の処理を施したものが、(f)である。この例では、 $k_i=2$ としている。なお、式(8)では、稜線の輝度を一定にするための正規化は行っていない。まず、稜線が凸か凹かで、符号は正負に分かれる。輝度の絶対値は、スクリーンに対して45度傾いた二つの平面が直交してできる稜線において、 $|l|=2$ となり、鋭角の稜線ほどこの値は大きくなる。これについては、2次微分値の近傍での最大値および最小値を用いて式(7)と同様の補正を加えれば、すべての稜線を一定の輝度で描くこともできる。

3.4 画像の合成

上記の方法で得られる輪郭線と稜線の画像、および、陰影画像に対して、適当な方法で合成を行うことにより、最終の目的画像を得る。ここで、合成とは、線演算、絶対値、最大・最小の打ち切り、などの演算からなり、その最適の組合せは対象や目的によって異なる。図3(g)は、(e)と(f)とを合成することにより、形状を線画のみで表現した例である。(g)と陰影画像(a)とを合成することにより、エッジを強調した陰影画像(h)が得られる。以上の例は、ハイライト描法¹⁾を意識し、凸稜線にハイライトが生じると仮定して白のまま残したものである。これに対し、(f)の絶対値をとってから合成すれば、すべてのエッジを黒で強調することもできる。このとき一般に、輪郭線を強く、稜線を弱く描くと、わかりやすい表現が可能である⁹⁾。その例を、(f'), (g'), (h')に示す。

3.5 アンチエイリアシング

他の描画方法と同様に、本手法でも輪郭線や稜線にはジャギーが生じる。ただし、微分オペレータの特性として、複数の近傍画素を用いて差分値を平均化するため、2値で線を描くほど顕著にはならない。

より完全なアンチエイリアシングが要求される場合は、距離画像上で行う。陰影画像では通常、スーパーサンプリングや面積比計算などにより、画素ごとの平均輝度を求めることにより解決している。このとき、距離画像でも同様の方法で平均奥行き値を求めればよい。

4. 反射、屈折像中のエッジの描画

前章で提案した手法は、距離画像の概念を拡張する

ことによって、反射、屈折像中に現れるエッジをも描画または強調することができる。前にも述べたように、輪郭線、稜線は、視点からの奥行き値に関して一種の不連続性を有する。ここで、奥行き値のかわりに、視点から最終的に映り込む物体までの光路長に着目すると、反射、屈折像中の輪郭線、稜線についても、同様の不連続性を有する。したがって、光路長を各画素値とする距離画像を作成し、1次あるいは2次微分オペレータを作用させると、反射、屈折像中の輪郭線ならびに稜線も、一般の輪郭線、稜線と同時に抽出できる。

次に、光路長の具体的な計算方法について説明する。光線追跡法では、処理の過程で視点もしくは物体表面から次の物体までの距離を計算する。基本的には、これらの距離を、視点から最終的に映り込む物体まで加算し、式(2)の透視変換を行えばよい。しかしこの場合、背景(無限遠点)と、背景が映り込んでいる物体とは、光路長が等しくなり、輪郭線は抽出できない。これを防ぐためには、たとえば次のようにすればよい。いま、透過は考えないとし、視点から n 個目の反射面までの距離の和を $d_v(n)$ 、この面の反射率を $r(n)$ とする。このとき、この面までの光路長 $z_i(n)$ を次式により定義する：

$$z_i(n) = (1 - r(n))d^2 / (wd_v(n)) + r(n)z_i(n+1). \quad (9)$$

そして、各画素の $z_i(1)$ の値を距離画像の画素値とすればよい。

なお、すべての映り込みの条件を一つの実数値で区別することは不可能であるため、輪郭線や稜線が抽出できないような例が生じる可能性もある。目的に応じた式(9)も工夫する必要がある。

5. 適用例

提案手法の適用例を図3、図5~7に示す。

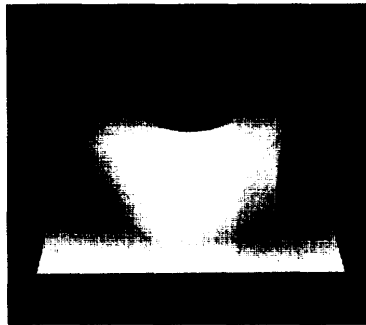
図3については3章で説明したとおりである。この程度の単純な形状であれば、隠線消去で線画を作成することもできる。しかし、エッジを白で強調するか黒で強調するかを試行錯誤で決める場合など、提案手法の方がはるかに容易に対応できる。

図5~7においては、(a)が通常の陰影画像、(b)が距離画像、(c)が画像処理を使って(b)をもとに作成した輪郭線画像、(d)が(a)と(c)を合成した強調画像である。

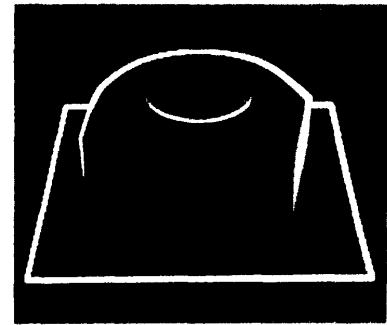
図5は、曲面の例としてトーラス面を描画したもの



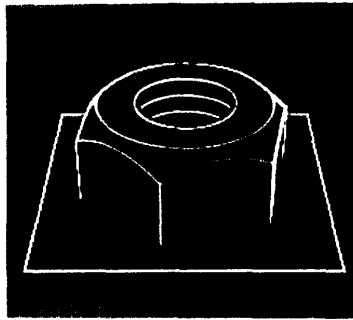
(a) 陰影画像
(a) Shaded image.



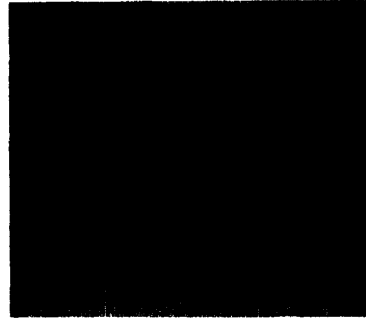
(b) 距離画像
(b) Range image.



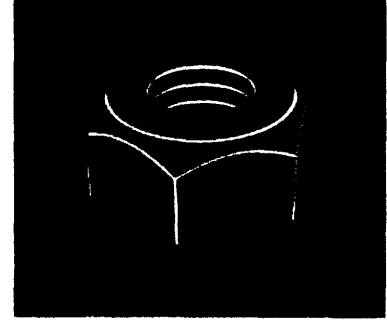
(c) 1次微分画像
(c) 1st order differential image.



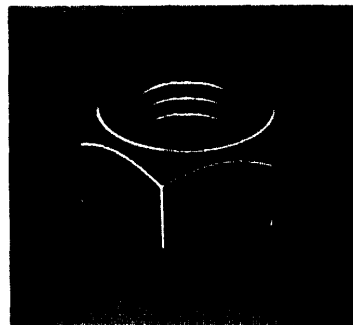
(d) 2次微分画像
(d) 2nd order differential image.



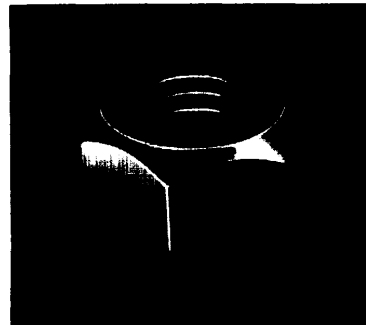
(e) 輪郭線画像
(e) Profile image.



(f) 稜線画像
(f) Internal-edge image.



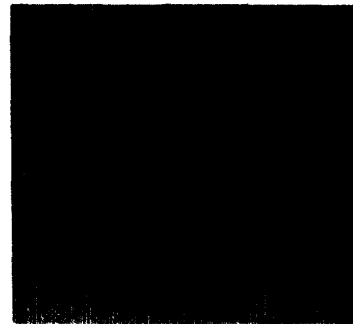
(g) 輪郭線, 稜線画像
(g) Profile and edge image.



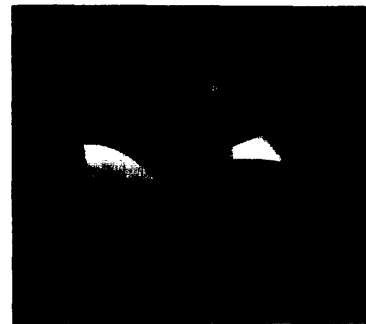
(h) 強調された画像
(h) Enhanced image.



(f') 稜線画像
(f') Internal-edge image.



(g') 輪郭線, 稜線画像
(g') Profile and edge image.

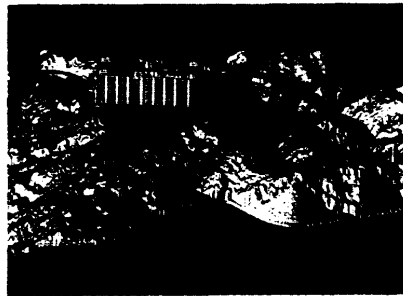


(h') 強調された画像
(h') Enhanced image.

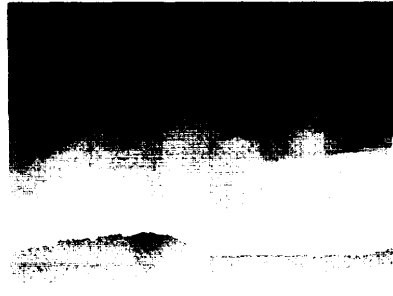
図3 エッジ強調画の描画過程
(f), (g), (h)は、ハイライトを意識して凸稜線を白で強調した例, (f'), (g'), (h')は、すべてのエッジを黒で強調した例である。

Fig. 3 Drawing process of edge enhanced pictures.

Two examples are shown: convex edges are enhanced with white lines in (f), (g), and (h); all edges are enhanced with black lines in (f'), (g'), and (h').



(a) 陰影画像
(a) Shaded image.



(b) 距離画像
(b) Range image.



(c) 輪郭線画像
(c) Profile image.

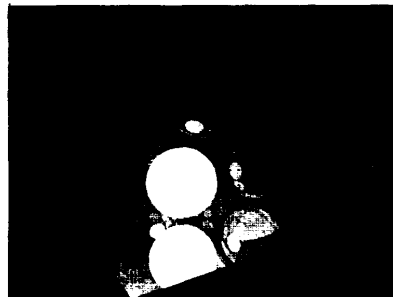


(d) 強調された画像
(d) Enhanced image.

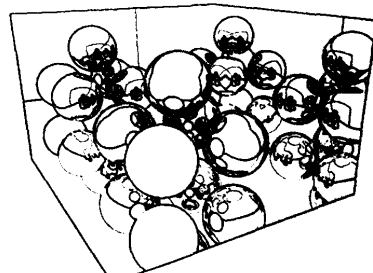
図 6 複雑な形状への適用例
Fig. 6 An example of complicated surfaces.



(a) 陰影画像
(a) Shaded image.



(b) 距離画像
(b) Range image.



(c) 輪郭線画像
(c) Profile image.



(d) 強調された画像
(d) Enhanced image.

図 7 反射物体への適用例
Fig. 7 An example of reflected objects.

である。どのような曲面であっても、面画として（隠面消去して）描画することができれば、輪郭線は単純な画像処理だけを用いて求められる。曲面から輪郭線を直接求める方法とは異なり、凹部の輪郭線の見落としが問題になることはない。

図6は、複雑な形状の例として、NTT 横須賀 R&D センター周辺の地形を描画した鳥瞰図である。原データは、高度のメッシュデータ（2m 間隔の実測値）であり、これを多面体近似したものを形状データとしてレンダリングしている。陰影付けされた画像（a）だけでも、地形のかなりの部分は理解できるが、山の重なり方（連続した山か、異なる山か）などに不明確な点がいくつかある。画像(c), (d)では、輪郭線を描画または強調することにより、このような部分もわかりやすくなっている。

図7は、反射による映り込みの多い物体を光線追跡法で描画し、4章で述べた方法（式(9)）を適用したものである。（c）において、反射像中の輪郭線も明確に現れていることがよくわかる。

なお、図3、図5～7の作画例では、すべての画像データを浮動小数点数として扱っている。

6. 考 察

ここでは、提案手法の長所および短所について考察する。

6.1 提案手法の長所

(1) エッジ強調は、幾何学的あるいは光学的要因に基づく本来のCG処理とは切り放された形で、簡単な2次元画像処理の組合せで実現している。画像処理専用ハードウェアにより、リアルタイム表示も可能であるため、画像処理手法やパラメータの選択も効率よく行うことができる。また、ある画像に対して適当な強調法がわかれば、類似の画像、特にアニメーションにおける一連のシーンに対しては、同一処理によって一括して強調を施すことができる。処理時間は画素数のみに依存し、形状の複雑さとは無関係である。

(2) 複雑な曲面に対しても、全く同じ手法が適用できる。また、もとの画像を光線追跡法で作成し、奥

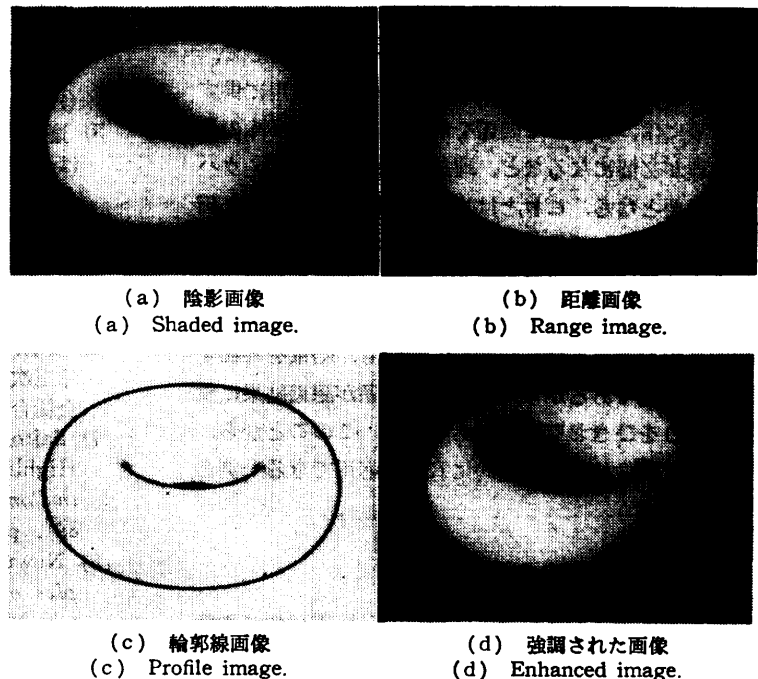


図5 曲面形状への適用例

Fig. 5 An example of curved surface.

行き値の概念を拡張することにより、これまで極めて困難であった反射、屈折像中のエッジをも描くことができる。

6.2 提案手法の短所

(1) 輪郭線、稜線の一部に、線のときれやノイズが生じる。これは、補正処理の工夫で軽減はできるが、完全に取り去ることは困難である。また、本手法で微細な領域を正確に描くことも困難である。

(2) 曲面を多面体近似した場合、輝度画像ではスムーズシェーディングにより十分なめらかに見えても、距離画像を2次微分した際に不連続線が目立ってしまう場合がある（図3(d)）。これを防ぐには、分割を細かくするか、奥行き値を2次以上の関数で補間するかで対処できる。

(3) 比較的大きな作業領域（メモリ、ディスク等）が必要である。これは、物体形状の複雑さとは無関係だが、解像度に比例して増大する。距離画像など、いくつかの画像や中間結果は、浮動小数点数またはビット数の多い整数での表現が不可欠である。作業領域の問題は、ハードウェアの進歩により近い将来には解決されると期待できる。

6.3 画像処理としての位置づけ

画像処理の立場からは、以下のことがいえる。一般に画像処理は、誤差の多い実測データから、何らかの

認識結果を計算機上に得るために、用いられることが多い。この場合、入力には厳密さが無いのに対して、出力には厳密さを要求される。それゆえ、実用に供するためには、ささいな問題を解決するために多くの例外処理が必要になるなど、理論よりも経験的なノウハウが重要となる。これとは逆に、提案手法で用いた画像処理は、CG 画像から人間が見てわかりやすい絵を作るものである。すなわち、一般に誤差のきわめて少ない理想的な入力データが使えるため、処理効果を理論的に解析できる余地がある。しかも、結果を評価するのは人間であるから、ささいな問題が認識結果に重大な誤りをひきおこすことも少ない。このことから、本手法は画像処理を理論的に有効に応用できる一例といえよう。

7. おわりに

隠面処理中に副産物として得られる距離画像に画像処理を施すことにより、エッジを強調したわかりやすい画像を作成する方法を提案した。強調という人為的処理を、幾何的あるいは光学的要因に基づく処理から分離したものであり、人間が試行錯誤を行って作画する場合に適している。また、従来困難であった、複雑な曲面や反射、屈折像中の輪郭線、稜線に対しても、容易に適用できる。今後はこの考え方をさらに進め、イラスト画のようなわかりやすい絵を CG で描くための、種々の画像処理的手法について、研究を進めていく。

謝辞 本研究の機会を与えていただいた、NTT HI 研究所知能ロボット研究部 高部陸男部長、滝川啓主幹研究員（現在、画像電信事業部）、本研究に関して御教示御討論いただいた、日本医科大学 玉井仁氏、ならびに、NTT 知能ロボット部の田中敏光研究主任、武川直樹主任研究員をはじめとする同僚諸氏に、感謝の意を表す。また、適切なコメントをいただいた査読者の方々に感謝する。

参考文献

- 1) 工業デザイン全集編集会編：デザイン技法—造形技法—一，工業デザイン全集 第4巻，日本出版サービス，東京（1982）。

- 2) 近藤邦雄，木村文彦，田嶋太郎：インタラクティブレンダリングシステムによる3次元形状の表現，情報処理，Vol. 26，No. 11，pp. 1401-1408（1985）。
- 3) 斎藤隆文：コンピュータに「絵」を描かせるには…，第27回情報処理学会プログラミングシンポジウム報告集，pp. 153-158（1986）。
- 4) 長尾 真：画像認識論，情報工学講座 16，コロナ社，東京（1983）。
- 5) 田村秀行：コンピュータ画像処理入門，総研出版，東京（1985）。
- 6) 安居院猛，羽生田千春，中嶋正之：特徴線による三次元データの表現，昭和60年度電子通信学会総合全国大会，No. 1647（1985）。
- 7) Saito, T., Shinya, M. and Takahashi, T.: Highlighting Rounded Edges, *New Advances in Computer Graphics (Proc. CG International 89)*, pp. 613-629（1989）。
- 8) Newmann, W. M. and Sproull, R. F.: *Principles of Interactive Computer Graphics*, 2nd ed., McGraw-Hill, New York（1979）。
- 9) Beasley, D.: *Design Illustration*, Heinemann Educational Books, London（1979）。

（平成2年1月31日受付）

（平成2年4月17日採録）



斎藤 隆文（正会員）

1959年生。1982年東京大学工学部計数工学科卒業。1987年同大学院工学系研究科情報工学専攻博士課程満期退学。同年日本電信電話(株)入社。現在、同社ヒューマンインタフェース研究所知能ロボット研究部勤務。形状処理，コンピュータグラフィックスの研究に従事。工学博士。1990年本学会学術奨励賞受賞。ACM SIGGRAPH 会員。



高橋時市郎

1954年生。1977年新潟大学工学部電子卒業。同年NTT電気通信研究所入所。以来、文字認識，画像生成の研究に従事。現在、ヒューマンインタフェース研究所知能ロボット研究部主幹研究員。電子情報通信学会，IEEE，AVIRG 各会員。