

自動微分を用いた適応的・効率的フォトンマッピング

生江拓也*, 牧野光則*

An Adaptive and Efficient Photon Mapping by Automatic Differentiation

Takuya NAMAЕ*, Mitsunori MAKINO*

abstract

Photon mapping is known as one of the efficient techniques to simulate global illumination in the field of computer graphics. The photon map is used in a certain of bidirectional ray tracing. In the first place, by the ray tracing from light sources the map consists of information on a set of photons, which are emitted from the light sources and reflect/refract surfaces in a given scene. Considering the information, an image with global illumination is efficiently rendered by the ray tracing from a viewpoint.

Since quality of images depends on number of photons representing illumination around them, enough photons must be generated from light sources. However, too enough photons cause long computation time. Actually, many photons are not always needed when they are located on “flat” illuminated surfaces. Therefore, an adaptive definition of photons should be developed for efficient photon mapping.

Applying automatic differentiation technique to detecting edges of surfaces and emitting photons efficiently, in this paper we will propose an adaptive photon mapping algorithm, which can generate images more efficiently with global illumination including caustics, half shadows, half shades and half highlights.

1 序論

コンピュータグラフィックス (Computer Graphics: CG) の分野では、より写実的な画像生成を実現するための研究が行われている。CG 画像の写実性を高めるためには、自然現象や物理現象に忠実、かつ、計算コストを抑えるような効率的なレンダリング技術の開発が必須となる。

CG の主要なレンダリング手法の 1 つにレイトレーシング法 [1] がある。目に入る光を目から逆に追跡することにより可視点の色や明るさを求める手法で、陰影や映り込み、透過・屈折などの光学現象を容易に表現できる。また、分散レイトレーシング [2] は、1 本のレイを空間または時間に関して分散させることにより、アンチエイリアシングや半影、モーション・ブラーなど、従来のレイトレーシング法では表現できなかった現象を表現できる。しかし、集光模様や色映りなど、光源から間接的に光が届くことによって生じる現象は表現できない。

これらを効率的に表現できる手法がフォトンマッピング [3] である。この手法は、まずはじめに、光源から放射される光の分布状況を把握するために光源からフォトン放出・追跡し、その分布状況をフォトンマップと呼ばれるデータ構造に格納する。次に、このフォトンマップの情報を用いて画像を生成する。このとき、対

*中央大学理工学部情報工学科 〒112-8551 東京都文京区春日 1-13-27

象物体の形状変化に関わらずフォトンを放出するため、非効率的なフォトン放出を行う可能性がある。このため、効率良くフォトンを放出することで計算コストを抑える工夫が必要である。一方、レンダリングには分散レイトレーシングを用いるが、分散すべきレイの選択と分散の程度の適応的決定が必要である。

これに対して、牧野、篠塚らは自動微分 [4][5] を利用して分散レイの選択を行う手法を提案した [6][7][8][9]。牧野、篠塚らの手法では、画像の各ピクセル中心を通るレイと物体との交差判定結果を微分値付きで計算する。得られた微分値を用いて当該ピクセルの周辺部 4～8 箇所の交差判定の 1 次近似値 (あるいは 2 階微分まで計算して 2 次近似値) を求める。レイ中心の計算結果と周辺部の近似計算結果が異なるピクセルのみレイの分散が必要と判断する。この手法は全ピクセルに対して、中心を通るレイの交差判定に微分計算を伴うため、通常のレイトレーシングの数倍の計算量を要するが、レイの分散が必要なピクセルを適応的に不足なく判定できる。この結果、レイの分散による計算時間の増加を抑えることができ、全体の計算時間も低減できる。

本論文ではこの適応的レイトレーシング法の考え方を踏襲し、自動微分によって適応的にフォトンを放出する手法を提案する。光源から放出したフォトンが物体のエッジ付近と交点を持つ場合はより多くのフォトンを放出し、そうでない場合は冗長なフォトン放出を抑えることにより、効率的なフォトンマップを構築する。また、大きさを持つ光源に対して適応的にオーバーサンプリングを行うことにより、陰影やハイライトなどを効率的に表現する手法を提案する。これにより、これらの現象を含んだ写実的な画像生成の効率化が期待できる。

2 準備

2.1 フォトンマッピング

フォトンマッピングは、集光模様や色映りなどの間接照明を容易に表現できる手法である [3]。この手法では、フォントレーシングならびにレンダリングの 2 段階を経て画像を作成する。

初めに、光源から放射される光線の進行状況 (反射・屈折を含む) 分布しているかを把握するために、光源からフォトンを多数放出・追跡し、その分布状況 (集光, 間接光による効果など) をフォトンマップに格納する (Fig. 1)。これをフォントレーシングと呼ぶ。

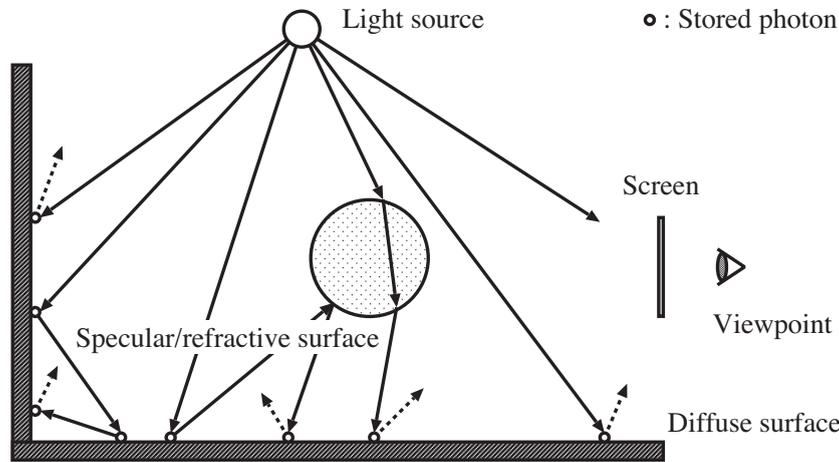


Fig. 1 フォントレーシング

次に、定められた視点から視線追跡 (レイトレーシング) を行い、視線が到達する物体表面上の位置を求める。この位置に関するフォトンマップの情報を用いることによって、光の直接・間接照射、影落ち等が考慮された輝度を得る。これを画像を構成する全てのピクセル (画素) について行い、画像を生成する (Fig. 2)。この画像生成処理をレンダリングと呼ぶ。フォトンマッピングにより、反射あるいは透過の性質を有する物体が存在する情景で、集光や間接光を含む画像を生成できる。

フォトンマッピングの前半であるフォントレーシングでは、多数のフォトン放出・追跡しなければならない。仮にフォトン放出が過少であれば、フォトンマップを作成するフォントレーシングの計算時間は短い

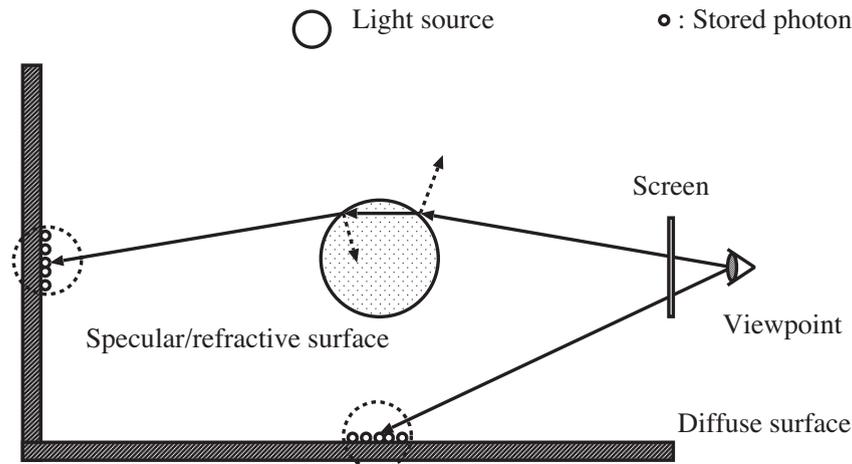


Fig. 2 レンダリング

ものの適切なフォトンマップを作成できず、間接光の表現が不適切となる。一方、過大なフォトンの放出は高品質なフォトンマップを作成できるが、フォントレーシングの計算時間増大を招く。すなわち、フォトン放出数の適切な設定は、生成画像の質と計算時間のバランスを取るために不可欠である。しかし、そのためには、物体の形状、位置、反射率、透過率、ならびに、光源の位置・強さなどの、対象とする空間の詳細な情報から、どこにどの程度の細かさで間接光の影響が出るのかを把握する必要がある。この把握をある程度の計算時間に抑えつつ、フォトン数を必要以上に、かつ、冗長性をできるだけ抑えることが求められている。

また、フォトンマッピングの後半ではレイトレーシングを行い、視線が到達する領域に存在するフォトンのエネルギー量からその箇所の輝度を決定する。フォトンが離散的に分布していることから、視線が少ない(画面解像度が情景に対して十分な細かさがない)場合に輝度変化が滑らかにならず、物体表面のエッジ部分がギザギザになるなど、エイリアシングと呼ばれる現象が発生することがある。これを解決する(アンチエイリアシング)ために、通常1ピクセル(画素)に1本射出するレイ(視線)を複数本当該ピクセル内に分散させ、結果を平均する分散レイトレーシングが用いられる。この場合、分散が必要なピクセルを簡単に検出し、冗長な分散を行わないことが計算量削減に不可欠である。

これまでに、フォトンマッピングの高速化に関して多くの研究が進められている。例えば、Christensenは、多数のフォトンが同じ領域に到達することが多いことに着目し、場所毎の輝度を事前計算することで高速化を図る手法を提案している[10]。また、Kellerらは画像における見かけの重要性からフォトンの放出方向を制御することで効率的にフォトンマップを作成する方法を提案している[11]。ハードウェアを利用する方法としては、例えば、蜂須賀がフォントレーシングをレイトレーシングではなくZバッファ法で置き換える手法を提案し、GPU(Graphics Processing Unit)に実装することにより高速化を達成している[12]。

2.2 自動微分法

ある n 変数からなる微分可能な関数を $f(x_1, x_2, \dots, x_n)$ とする。 f は、 $w = \phi(u, v)$ または $w = \psi(u)$ で表される四則演算や初等関数などの複数の単項・2項演算もしくはそれらの合成関数で構成されているとする。ここで u, v は変数 x_i または ϕ, ψ による演算結果を表す。このとき、使用されている演算の種類が有限個であり、かつ任意の変数 x_i に関する u, v の(1階)偏微分値 $\frac{\partial u}{\partial x_i}, \frac{\partial v}{\partial x_i}$ が既知であれば、 w の x_i に関する偏微分値 $\frac{\partial w}{\partial x_i}$ は、微分の鎖律により高々定数回の演算で計算できる。この操作を f の関数値を計算する手続きに加える形で行えば、 f の関数値と同時に f の x_i に関する偏微分値 $\frac{\partial f}{\partial x_i}$ が得られる。この手法を自動微分 [4][5] と呼ぶ。

自動微分は数式処理ではなく数値計算であるため、対象の式への制約は低い。さらに、打ち切り誤差や丸め誤差の影響を受けにくいいため、一般的な数値微分に比べて扱いやすいといえる。

3 自動微分によるフォトンマッピングの適応化手法

3.1 概要

これまでの高速化手法としては、フォトン放出数削減を行わず、その後の処理を効率化する方法、事前計算によってフォトン放出方向を制御する方法などがある。これに対して本論文では、自動微分をフォトンマッピングのアルゴリズム全体に適用する。これにより、フォントレーシング中でフォトン放出数を適応化し、また、その後の分散レイトレーシング中で分散レイを適応化する。フォトンマッピングのアルゴリズムを大幅には変更せずに、より多くのフォトン放出すべきか、あるいは、より多くレイを分散させるかを状況に応じて判断できることが提案手法の特徴である。提案手法の全体的な流れを以下に示す。

Step1: フォントレーシングを行い、フォトンマップを構築する。

Step1.1: 光源上の任意の点からフォトン放出・追跡し、フォトンマップに格納する。

Step1.2: 光源から放出したフォトンが物体のエッジ付近に衝突した場合、その近傍にフォトンを追加放出・追跡し、フォトンマップに格納する。

Step2: フォトンマップを用いて分散レイトレーシングを行う。

Step2.1: 各ピクセルに対して視点からレイを追跡する。

Step2.2: レイが物体のエッジ部分と交点を持った場合、そのピクセルに対してレイを分散させる。

Step2.3: レイが物体と交点を持った場合、光源に対してレイを追跡し、輝度計算を行う。

Step2.4: レイと物体との交点が半陰、半影、または半ハイライト領域内である場合は、光源に対してオーバーサンプリングを行う。

Step2.5: フォトンマップを用いて集光模様および間接光を計算する。

自動微分は、ピクセルや光源の座標値や、追跡するフォトン放出方向やレイのベクトルに対して利用する。すなわち、偏微分値を持った追跡ベクトルを用いて物体との交点計算を行うことにより、交点計算の解も偏微分値を持つ。この値から近傍値を予測することにより、各判定を行う。なお、簡略のため1階偏微分についてのみ説明するが、2階偏微分についても同様に行う。

各データはその値自身 $value$ に加え、3次元座標軸 x, y, z に関する階偏微分値を持つ。これを、

$$value < \frac{\partial value}{\partial x}, \frac{\partial value}{\partial y}, \frac{\partial value}{\partial z} >$$

で記述する。次節以降、各データはこれらの値を持つものとして説明を進める。

3.2 フォントレーシングに対する適用

光源上の位置を P_L 、フォトンの放出方向ベクトル (単位ベクトル) を V_D とすると、 P_L から V_D 方向へ1進んだ位置 P_N は、次式で表せる。

$$P_N = V_D + P_L \quad (3.1)$$

この P_N に以下の式で x, y, z に関する偏微分値を与える.

$$\begin{aligned} P_{N_x} &< \frac{\partial P_{N_x}}{\partial x} = 1, \frac{\partial P_{N_x}}{\partial y} = 0, \frac{\partial P_{N_x}}{\partial z} = 0 > \\ P_{N_y} &< \frac{\partial P_{N_y}}{\partial x} = 0, \frac{\partial P_{N_y}}{\partial y} = 1, \frac{\partial P_{N_y}}{\partial z} = 0 > \\ P_{N_z} &< \frac{\partial P_{N_z}}{\partial x} = 0, \frac{\partial P_{N_z}}{\partial y} = 0, \frac{\partial P_{N_z}}{\partial z} = 1 > \end{aligned} \quad (3.2)$$

これを用いてフォントレーシングを行うと共に, P_N の近傍におけるフォントレーシングも同時に行う. 以下に本手法で用いる適応的フォントレーシングのアルゴリズム (3.1 の **Step1** に相当) の概要を示す.

Step1: 最大フォトン放出数 $photon_{max}$ を設定する. フォトン放出数 $photon$ を 0 とし, **Step2** 以降を再帰的に繰り返す.

Step2: 新しいフォトンを放出するため, フォトン放出方向ベクトル V_D を求める.

$$photon = photon + 1 \quad (3.3)$$

$$V_D = P_N - P_L \quad (3.4)$$

このとき, レイは次式で表される.

$$P_{ray} = tV_D + P_L, \quad t > 0 \quad (3.5)$$

視線追跡と同様, P_{ray} を用いて交点計算を微分付きで行い, その交点計算結果を $f_{P_{ray}}$ とする.

$$f_{P_{ray}} = \begin{cases} 0 & (\text{物体と交差しない}) \\ 1 & (\text{物体と交差する}) \end{cases} \quad (3.6)$$

Step3: $f_{P_{ray}} = 0$ ならば, **Step9** へ. $f_{P_{ray}} = 1$ ならば, **Step4** へ.

Step4: 自動微分を利用した 1 次近似で P_N 近傍における交点計算結果を予測する.

Step5: **Step4** の結果と $f_{P_{ray}}$ が一致すれば **Step6** へ. そうでなければ **Step7** へ.

Step6: そのフォトンのエネルギーを大きくし, フォトンマップに格納する. **Step9** へ.

Step7: そのフォトンにフォトンマップに格納し, **Step8** へ.

Step8: $f_{P_{ray}}$ 近傍に対してフォトンの追加放出を行い, **Step9** へ.

Step9: フォトン放出数 $photon + 1$ が最大フォトン放出数 $photon_{max}$ を超えていない場合, **Step2** へ.

以上のように, 光源から放出したフォトンが物体のエッジ付近と交点を持つ場合, その付近にフォトンの追加放出を行い, そうでない場合は冗長なフォトン放出を抑えることにより, 効率的なフォトンマップを構築する. フォトンがエッジ付近でない領域に到達する場合, そのフォトンのエネルギーを大きくしてフォトンマップに格納することによって, エッジ付近に集中的にフォトンを放出することに対する調整を行う. また, P_N における近傍値は, P_N において V_D に直交する正方形の頂点において求める (Fig. 3).

3.3 大きさを持つ光源に対する適用

本節では大きさを持つ光源として四角形面光源を対象とするが, これ以外でも一定の大きさを持つ光源ならば本手法を適用できる.

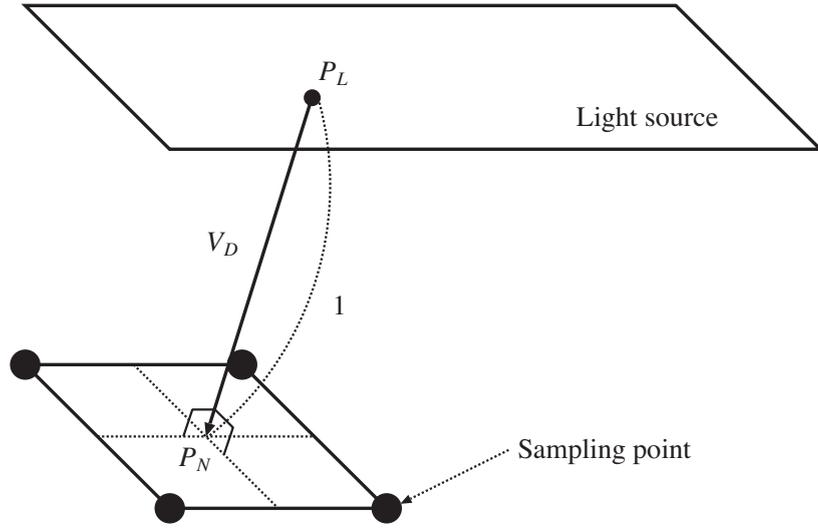


Fig. 3 フォトンのサンプリング点

光源の中心位置を \mathbf{L} とし, \mathbf{L} の座標 (L_x, L_y, L_z) に次式で x, y, z に関する偏微分値を与える.

$$\begin{aligned} L_x &< \frac{\partial L_x}{\partial x} = 1, \frac{\partial L_x}{\partial y} = 0, \frac{\partial L_x}{\partial z} = 0 > \\ L_y &< \frac{\partial L_y}{\partial x} = 0, \frac{\partial L_y}{\partial y} = 1, \frac{\partial L_y}{\partial z} = 0 > \\ L_z &< \frac{\partial L_z}{\partial x} = 0, \frac{\partial L_z}{\partial y} = 0, \frac{\partial L_z}{\partial z} = 1 > \end{aligned} \quad (3.7)$$

これを用いて \mathbf{L} からの光による半陰判定, 半影判定, および半ハイライト判定を行うと共に, \mathbf{L} の近傍からの光による各判定も同時に行う.

ある有限の大きさを持つ光源 (中心 \mathbf{L}) に対し, 物体表面上のある位置 \mathbf{x} から $\boldsymbol{\omega}$ 方向に反射する輝度 $L_r(\mathbf{x}, \boldsymbol{\omega})$ を計算するアルゴリズム (3.1 の **Step2.3**, **Step2.4** に相当) の概要を以下に示す.

Step1: \mathbf{x} から \mathbf{L} 方向のレイ (単位ベクトル) を \mathbf{L}_{ray} とする.

$$\mathbf{L}_{ray} = \frac{\mathbf{L} - \mathbf{x}}{|\mathbf{L} - \mathbf{x}|} \quad (3.8)$$

Step2: \mathbf{x} と \mathbf{L}_{ray} を用いて陰領域かどうかの判定を自動微分付きで計算し, 判定結果を $f_1 \mathbf{L}_{ray}$ とする.

$$f_1 \mathbf{L}_{ray} = \begin{cases} 0 & (\text{陰領域のとき}) \\ 1 & (\text{陰領域でないとき}) \end{cases} \quad (3.9)$$

Step3: 自動微分を利用した 1 次近似または 2 次近似で光源中心近傍における陰判定を行う.

Step4: **Step3** の結果と $f_1 \mathbf{L}_{ray}$ が一致すれば **Step5** へ. そうでないならば **Step15** へ.

Step5: $f_1 \mathbf{L}_{ray} = 0$ の場合,

$$L_r(\mathbf{x}, \boldsymbol{\omega}) = 0 \quad (3.10)$$

として終了. そうでない場合は **Step6** へ.

Step6: \mathbf{x} と \mathbf{L}_{ray} を用いて影領域かどうかの判定を自動微分付きで計算し、判定結果を $f_2 \mathbf{L}_{ray}$ とする。

$$f_2 \mathbf{L}_{ray} = \begin{cases} 0 & (\text{影領域のとき}) \\ 1 & (\text{影領域でないとき}) \end{cases} \quad (3.11)$$

Step7: 自動微分を利用した 1 次近似または 2 次近似で光源中心近傍における影落ち判定を行う。

Step8: **Step7** の結果と $f_2 \mathbf{L}_{ray}$ が一致すれば **Step9** へ。そうでないならば **Step15** へ。

Step9: $f_2 \mathbf{L}_{ray} = 0$ の場合、

$$L_r(\mathbf{x}, \boldsymbol{\omega}) = 0 \quad (3.12)$$

として終了。そうでない場合は **Step10** へ。

Step10: その物体が鏡面反射性物体である場合、**Step11** へ。そうでない場合は **Step14** へ。

Step11: \mathbf{x} と \mathbf{L}_{ray} を用いてハイライト領域かどうかの判定を自動微分付きで計算し、判定結果を $f_3 \mathbf{L}_{ray}$ とする。

$$f_3 \mathbf{L}_{ray} = \begin{cases} 0 & (\text{ハイライト領域のとき}) \\ 1 & (\text{ハイライト領域でないとき}) \end{cases} \quad (3.13)$$

Step12: 自動微分を用いた 1 次または 2 次近似で光源中心近傍におけるハイライト判定を行う。

Step13: **Step12** の結果と $f_3 \mathbf{L}_{ray}$ が一致すれば **Step14** へ。そうでないならば **Step15** へ。

Step14: 以下の式を計算して終了。

$$L_r(\mathbf{x}, \boldsymbol{\omega}) = f_r(\mathbf{x}, \mathbf{L}_{ray}, \boldsymbol{\omega}) L_i(\mathbf{x}, \mathbf{L}_{ray}) (\mathbf{L}_{ray} \cdot \mathbf{n}) \quad (3.14)$$

Step15: 以下の式を計算して終了。

$$L_r(\mathbf{x}, \boldsymbol{\omega}) = \frac{1}{m} \sum_{j=1}^m f_r(\mathbf{x}, \mathbf{l}_j, \boldsymbol{\omega}) L_i(\mathbf{x}, \mathbf{l}_j) (\mathbf{l}_j \cdot \mathbf{n}) \quad (3.15)$$

この結果、冗長なオーバーサンプリングを削減した効率的な輝度計算を実現している。ただし、 $f_r(\mathbf{x}, \mathbf{l}_j, \boldsymbol{\omega})$ は、ある位置 \mathbf{x} において光源方向ベクトル \mathbf{l}_j 方向から受けた光を $\boldsymbol{\omega}$ 方向に反射する割合、 $(\mathbf{l}_j \cdot \mathbf{n})$ は、光源方向ベクトル \mathbf{l}_j とある位置 \mathbf{x} における法線 \mathbf{n} との内積である。また、 \mathbf{L} における近傍値は、頂点と各辺の中点において求める (Fig. 4)。

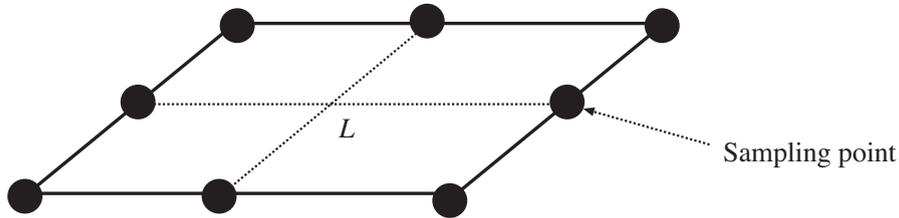


Fig. 4 四角形面光源のサンプリング点

4 シミュレーション

提案手法を用いて作成した画像を示す。“The Cornell Box”[13]を模した、閉空間中にそれぞれ拡散反射、鏡面反射、透過屈折の属性を有する3個の球を配置した空間を対象とした。画像はすべて300×450ピクセルであり、CPU: Pentium4(2.0GHz), OS: Red Hat Linux 7.2で作成した。

Fig. 5は通常のフォトンマッピングによって作成した画像である。Fig. 6はフォントレーシング部分のみ、Fig. 7は大きさを持つ光源によって生じる現象に対して、それぞれ自動微分を適用した手法を用いて作成した、提案手法を部分的に取り入れた場合の画像である。そして、Fig. 8は提案手法全体、すなわちフォントレーシング、大きさを持つ光源によって生じる現象、およびアンチエイリアシングに対して自動微分を適用した手法を用いて作成した画像である。ここで、従来手法(通常のフォトンマッピング)と提案手法を、フォトン数、分散ピクセル数、計算時間で比較した結果をTable 1に示す。なお、Table 1には、フォントレーシングと大きさを持つ光源によって生じる現象の双方に自動微分を適用した結果も示している。

各図の違いがほとんど見られないことから、提案手法は所望の画質を実現している。また、提案手法は、従来手法に比べ計算時間が少ない。本来の自動微分は通常計算の数倍の処理量となるが、提案手法は自動微分によってフォトンの放出やレイの分散の是非を判断し、不要とした箇所の計算を行わない。この結果、冗長な計算の削除量が自動微分による計算量増加を上回ったため高速化を達成した。

最高速処理を達成した、手法全体に自動微分を導入した場合には、従来手法の $\frac{1}{100}$ 以下となった。これは、自動微分を用いることによって、効率的に適切なフォトンマップを構築でき、適切なフォトンマップに対する冗長なオーバーサンプリングを削減できたことを示している。すなわち、提案手法はフォトンマッピング全体に自動微分を適用することで、計算の効率性の高い相乗効果を得た。

Table 1 初期設定と計算結果

| | 従来手法 | 提案手法 |
|--------------------------------------|-----------|---------|
| コースティックフォトン放出数 | 25,000 | 5,000 |
| グローバルフォトン放出数 | 500,000 | 100,000 |
| コースティックフォトン格納数 | 15,538 | 4,383 |
| グローバルフォトン格納数 | 713,106 | 149,957 |
| 光源に対する分散ピクセル数 | 300 × 450 | 10,384 |
| AA(※)に対する分散ピクセル数 | 300 × 450 | 3,386 |
| Fig. 5 (従来手法) の計算時間 [s] | 15,538 | – |
| Fig. 6 (フォントレーシングに適用) の計算時間 [s] | – | 11,056 |
| Fig. 7 (大きさを持つ光源計算に適用) の計算時間 [s] | – | 2,159 |
| フォトンレーシングと大きさを持つ光源計算に適用した場合の計算時間 [s] | – | 1,272 |
| Fig. 8 (アルゴリズム全体に適用) の計算時間 [s] | – | 121 |

※ AA: アンチエイリアシング

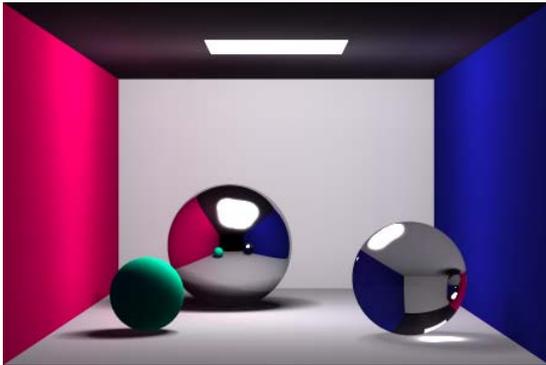


Fig. 5 従来手法



Fig. 6 提案手法 (1) (フォントレーシングに自動微分を適用)



Fig. 7 提案手法 (2) (大きさをもつ光源に自動微分を適用)

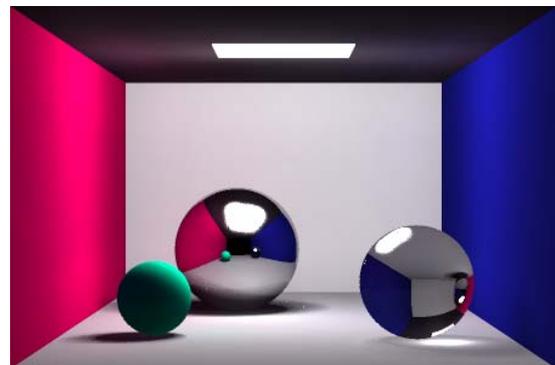


Fig. 8 提案手法 (3) (全体に自動微分を適用)

5 結論

本論文では、フォントレーシングおよびレンダリングにおいてレイの分散が必要であるか否かを自動微分を利用し判定することで、フォトンマッピングを適応化する手法を提案した。シミュレーションによって、本手法は従来手法による画像と同等の質の画像をより効率的に生成できることを確認した。提案手法の特徴として以下が挙げられる。

- 自動微分によって分散すべきレイとそうでないものを区別できる。
- 効率的なフォトンマップの構築および画像生成ができる。
- フォントレーシング、大きさを持つ光源、およびアンチエイリアシングに対して適用できる。
- そのすべての組み合わせに対する適応化を同時に行える。
- レイトレーシング法と同様の物体を扱える。

今後の課題として、自動微分の適用箇所の追加・適用方法の改善などによる、提案手法の更なる効率化が挙げられる。特に、対象空間に微小な物体が多数存在する場合、提案手法は自動微分を適用しても冗長性を排除できず、結果として計算量が従来手法を上回る可能性がある。このため、適用箇所の追加などで提案手法を改善し、対象とする空間の条件を緩和する必要がある。

謝辞

本研究は、中央大学理工学研究所共同研究の一環として行われたものである。

参考文献

- [1] T. Whitted: “An improved illumination model for shaded display”, *Communications of the ACM*, Vol.23, No.6, pp.343–349, 1980.
- [2] R. L. Cook, T. Porter, and L. Carpenter: “Distributed ray tracing”, *Computer Graphics*, Vol.18, No.3, pp.137–145, 1984.
- [3] H. W. Jensen: “Realistic Image Synthesis Using Photon Mapping”, A K Peters, Ltd., 2001.
- [4] 久保田光一, 伊理正夫: “アルゴリズムの自動微分と応用”, コロナ社, 1998.
- [5] 福井義成, 野寺隆志, 久保田光一, 戸川隼人: “新数値計算”, 共立出版, 1999.
- [6] M. Makino: “Detection of edges and approximation of surfaces in the use of automatic differentiation in computer graphics”, *IEICE Trans. Fundamentals*, Vol.85-A, No.3, 2002.
- [7] 篠塚五月, 牧野光則: “自動微分を利用した適応的分散レイトレーシング”, *電子情報通信学会論文誌 A*, Vol.J85-A, No.10, pp.1115–1125, 2002.
- [8] 井下雅美, 牧野光則, 柏木雅英, 大石進一: “自動微分を援用したオーバーサンプリングレイトレーシングの改良”, 第10回 NICOGRAPH 論文コンテスト論文集, pp.22–32, 1994.
- [9] 牧野光則, 井下雅美, 柏木雅英, 大石進一: “自動微分を援用した適応的オーバーサンプリングレイトレーシング法”, *中央大学理工学研究所論文集*, Vol.7, pp.45–55, 2002.
- [10] P. H. Christensen: “Faster Photon Map Global Illumination”, *ACM Journal of Graphics Tools*, Vol.4, No.3, pp.1–10, 1999.
- [11] A. Keller, I. Wald: “Efficient Importance Sampling Techniques for the Photon Map”, *Proc. of Vision Modeling and Visualization 2000*, pp.271–279, 2000.
- [12] 蜂須賀恵也: “ハードウェアの支援による 高速な大域照明レンダラーの開発”, *情報処理振興事業協会, 2002年度成果報告集第二版, 未踏ソフトウェア創造事業*, <http://www.ipa.go.jp/SPC/report/02fy-pro/report/1260/paper.pdf>, 2003.
- [13] Cornell University Program of Computer Graphics: “The Cornell Box”, <http://www.graphics.cornell.edu/online/box/>