

# 分散レイトレーシング法への自動微分の適用 (輝度ならびに形状変化を考慮した適応的画像生成法に関する研究)

研究代表者 研 究 員 牧野 光則 (中央大学理工学部)  
共同研究者 研 究 員 趙 晋輝 (中央大学理工学部)

## 1 はじめに

コンピュータ・グラフィックス (CG) 技術の中で、レイ・トレーシング法 [1] は視線追跡という単一なアルゴリズムで、影落ち、映り込み、透過・屈折などの光が起こす現象を容易に表現でき、広く利用されるレンダリング (描画) アルゴリズムである。通常、視線 (レイ) は太さを持たず、1 個の始点 (1 次レイの場合は視点) の各現象に対するレイは 1 本のみ設定される。このため、ある領域の情報 (幾何情報、輝度情報) を反映するためにレイは太さを持つべきである。しかし、実際にはレイは太さを持たず線として追跡する。このため、本来考慮すべき情報が欠落し、現実と大きく異なる場合がある。例えば、画像中にギザギザ (エイリアシング) が生じたり、半影やモーション・ブラーのような現象を表現できない。

そこで本研究では、Distributed Ray Tracing (分散レイ・トレーシング) [2] に自動微分 (1 階偏微分または 2 階偏微分) を利用した文献 [3], [4] の方法を拡張し、半影、モーション・ブラーに対しても適応的にレイを分散させる手法を提案する。提案手法を用いることにより、半影、モーション・ブラーに対する適応的分散を可能とし、効率的に両現象を表現する。また、処理そのものは別々であるがアルゴリズムには共通性があるため、アンチ・エイリアシングとの統一処理が可能となる。

## 2 自動微分を利用した適応的分散

### 2.1 自動微分による適応的分散レイ・トレーシング

以下、本稿で提案する自動微分を用いた適応的分散手法について説明する。本稿では、全ての物体は 2 次以下の関数で記述される物体を対象として説明するが、自動微分が適用可能な物体であればより高次の形状でも同様である。

各データはその値自身 *value* に加え 3 次元座標軸  $x, y, z$  ならびに時間軸  $t$  に関する偏微分値を持つ。これを、

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

で記述する。

#### 2.1.1 半影に対する適用

本節では大きさを持つ光源として、静止している球面光

源と平面 (四角形) 光源を対象とするが、これら以外でも一定の大きさを持つ光源ならば適用できる。この場合、 $t$  に関する偏微分値はすべて 0 となり、結果に影響しないので記述を省略する。

光源の中心位置を  $L$  とすると、 $L$  の座標  $(L_x, L_y, L_z)$  は、

$$L_x = x, \quad L_y = y, \quad L_z = z \quad (1)$$

である。これに座標値に加え  $x, y, z$  に関する偏微分値を持たせると、

$$\begin{aligned} L_x < \frac{\partial L_x}{\partial x} = 1, \frac{\partial L_x}{\partial y} = 0, \frac{\partial L_x}{\partial z} = 0 > \\ L_y < \frac{\partial L_y}{\partial x} = 0, \frac{\partial L_y}{\partial y} = 1, \frac{\partial L_y}{\partial z} = 0 > \\ L_z < \frac{\partial L_z}{\partial x} = 0, \frac{\partial L_z}{\partial y} = 0, \frac{\partial L_z}{\partial z} = 1 > \end{aligned} \quad (2)$$

と表せる。これを用いて影落ち判定を行う。

ある光源 (中心  $L$ ) による点  $p$  の輝度  $I_p$  を計算するアルゴリズムの概要を図 1 に示す。 $L$  方向レイを分散した場合の輝度  $I_p$  は図 1 のアルゴリズムにより、

$$I_p = \frac{1}{n} \sum_{i=1}^n I_i \quad (3)$$

で与えられる。ここで  $n$  はレイの分散数を表す。以上により、光源方向レイに幅を持たせることができ、光源に比べ小さな物体による半影も検出できる。

#### 2.1.2 モーション・ブラーに対する適用

ある与えられた時間区間の中間点を  $T_m$  とすると、

$$T_m = t \quad (4)$$

であり、 $T_m$  に関して

$$T_m < 0, 0, 0, \frac{\partial T_m}{\partial t} = 1 > \quad (5)$$

が成り立つ。

この  $T_m$  において視線追跡した結果から、与えられた時間区間の両端 (開始時刻と終了時刻) での計算結果を 1 次近似し、分散が必要か否かを判定する。本手法では、移動物体表面の各点の位置を次式で示す時刻  $t$  の関数によって定義する。

$$x = f_x(t), \quad y = f_y(t), \quad z = f_z(t) \quad (6)$$

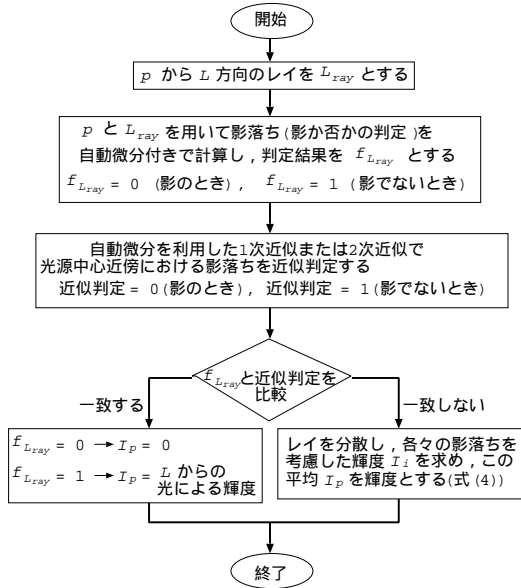


図 1 半影に対するアルゴリズム

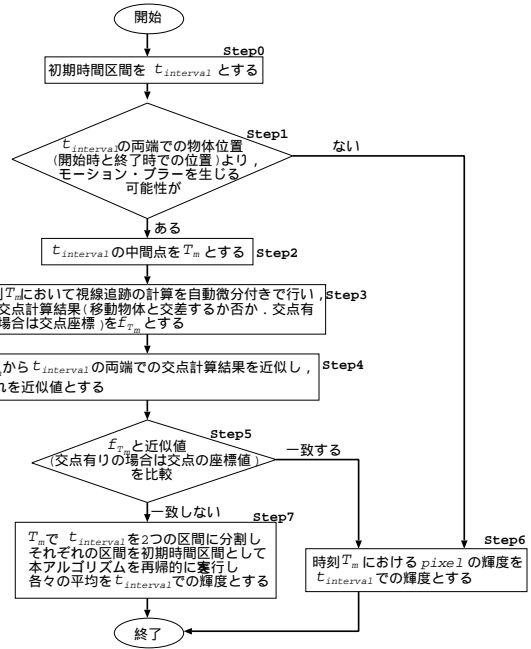


図 2 モーション・ブラーに対するアルゴリズム

あるピクセル  $pixel$  のモーション・ブラーによる輝度  $I_m$  を計算するアルゴリズムの概要を図 2 に示す。

### 2.1.3 アルゴリズム概要

アンチ・エイリアシングを含むアルゴリズム全体の概要は以下の通りである。

Step1 分散領域の中心を選択する。(半影 光源の中心, モーション・ブラー 時間区間の中点, アンチ・エイリアシング ピクセルの中心)

Step2 自動微分付きで視線追跡を行う。

Step3 Step2 の計算結果から, 分散領域内のある点 (半影 光源中心の近傍点, モーション・ブラー 時間区間の両端, アンチ・エイリアシング ピクセル中心の近傍点) における視線追跡の近似値を求め, 分散が必要ならば Step4 へ。分散が不要ならば Step5 へ。

Step4 レイを分散させ, 各計算結果からピクセルの輝度を求め, 終了。

Step5 Step2 の計算結果からピクセルの輝度を求め, 終了。

### 2.2 計算時間の考察

本手法で利用した自動微分は, 関数値計算の高々定数倍の計算量を必要とする。もし, 全てのレイが分散すべき情景ならば, 提案手法の効率は悪化するの明らかである。ここで,

- $a$  分散ピクセル数 / 総ピクセル数
- $b$   $t_n/t'_n$
- $c$  レイの分散数
- $t_n$  レイを分散しないピクセル 1 個の平均計算時間 [s]
- $t'_n$  通常のレイ・トレーシングによるピクセル 1 個の平均計算時間 [s]
- $t_d$  レイを分散するピクセル 1 個の平均計算時間 [s]
- $t'_d$  従来の分散レイ・トレーシングによるピクセル 1 個の平均計算時間 [s]
- $n_p$  画像一辺のピクセル数 (グラフの横軸)
- $t_{img}$  画像生成時間 (グラフの縦軸)[s]

とおくと, 正方形の画像の場合,

$$t_{img} = (1-a)n_p^2 t_n + a n_p^2 t_d = \{(1-a)t_n + a t_d\} n_p^2 \quad (7)$$

となる。また,

$$t'_d = c t'_n \quad (8)$$

であるので,

$$t_d = t_n + t'_d - t'_n = b t'_n + c t'_n - t'_n = (b+c-1)t'_n \quad (9)$$

となる。従来手法を用いた場合の画像生成時間  $t'_{img}$  と  $t_{img}$  の比をとると,

$$\begin{aligned} \frac{t_{img}}{t'_{img}} &= \frac{\{(1-a)t_n + a t_d\} n_p^2}{n_p^2 t'_d} \\ &= \frac{(1-a) b t'_n + a(b+c-1)t'_n}{c t'_n} \\ &= \frac{ac + b - a}{c} \end{aligned} \quad (10)$$

となる。よって、式 (10) が 1 より小さくなるためには、

$$b < c - a(c - 1) \quad (11)$$

を満たせばよい。

通常の情景では、およそ  $a = 0.1 \sim 0.3$  程度、レイの分散数  $c$  は数十～数百程度であり、 $b$  は数倍～十数倍程度であるので、大概式 (11) を満たす。逆に、式 (11) を満たすような状況であれば提案手法は有効である。

### 3 シミュレーション

本手法で作成した画像を図 3 ~ 図 19 に示す (環境: Ultra SPARC Iii 443MHz)。計算時間、分散ピクセル数を表 1 に示す。

初めに、平面光源による半影に対する適用結果を示す。図 3 ~ 図 8 は 300×300 ピクセル、図 9、図 10 は 600×600 ピクセルである。自動微分については、図 3、図 4、図 9 が 1 階偏微分まで、図 5、図 6 が 2 階偏微分まで行った結果である。また、図 4、図 6 では自動微分によってレイの分散が必要であると判定された領域を色づけした。比較のため、すべてのピクセルにおいて光源方向レイを分散した画像を図 7、図 8、図 10 に示す。光源方向レイの分散数は図 8 で 16 本、それ以外では 49 本とした。

次に、図 11 中央部の球が画像右側へ等速直線運動する場合の、モーション・ブラーに対する適用結果を示す。自動微分については、図 12、図 13、図 18 が 1 階偏微分まで、図 14、図 15 が 2 階偏微分まで行った結果である。また、図 13、図 15 では自動微分によってレイの分散が必要であると判定された領域を色づけした。比較のため、全てのピクセルで時間軸に関してレイを分散した画像を図 16、図 17、図 19 に示す。

表 1 計算時間・分散ピクセル数

	画像	計算時間 [s]	分散ピクセル数
半影	図 3	155	27901
	図 5	527	27706
	図 7	550	物体と交差した すべての 1 次レイ数
	図 8	169	物体と交差した すべての 1 次レイ数
	図 9	610	111544
	図 10	1936	物体と交差した すべての 1 次レイ数
モーション・ブラー	図 12	139	15684
	図 14	623	10348
	図 16	2696	全て (90000)
	図 17	129	全て (90000)
	図 18	626	62785
	図 19	21805	全て (360000)

### 4 むすび

本研究では、分散レイ・トレーシングに自動微分を適用し、適応的にレイの分散を行い、低計算コストで良質の CG 画像を生成する手法を提案した。本手法により、半影、モーション・ブラーをアンチ・エイリアシングと統一して処理できる。

### 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, June, 1980.
- [2] R.L.Cook, T.Porter and L.Carpenter, "Distributed Ray Tracing," Computer Graphics, Vol.18, No.3, pp.137-145, 1984.
- [3] 井下, 牧野, 柏木, 大石, "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 NICO-GRAPH 論文コンテスト論文集, pp.22-32, 1994.
- [4] 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.
- [5] 大淵, 青野, "Low-discrepancy sequence を用いた準モンテカルロレンダリング", 情報処理学会 グラフィクスと CAD 研究会報告, Vol.96, No.77, pp.91-96, 1996.



図 3 結果 (1 階)



図 4 検出 (1 階)

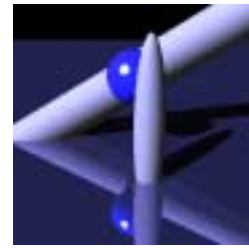


図 11 結果 (静止)



図 5 結果 (2 階)



図 6 検出 (2 階)

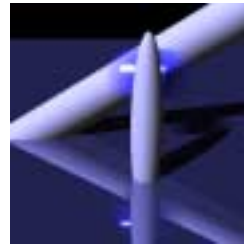


図 12 結果 (1 階)



図 13 検出 (1 階)



図 7 全て分散 (49)



図 8 全て分散 (16)

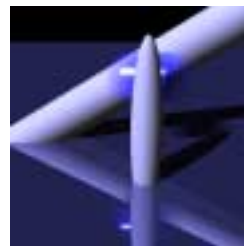


図 14 結果 (2 階)



図 15 検出 (2 階)



図 9 結果 (1 階, 解像度倍)



図 10 全て分散 (49, 解像度倍)

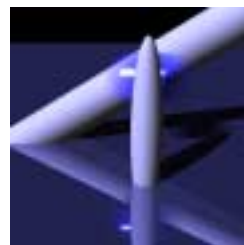


図 16 全て分散 (128)



図 17 全て分散 (5)



図 18 結果 (1 階, 解像度倍)

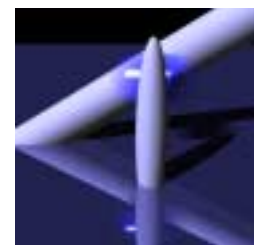


図 19 全て分散 (128, 解像度倍)