

G-03

球面ガウス関数を用いた全周波照明環境下における間接照明計算 Indirect Illumination under All-Frequency Lighting Environments using Spherical Gaussian Functions

古家 互† 岩崎 慶†
Wataru Furuya Kei Iwasaki

1. はじめに

実世界の複雑な照明環境を用いて、写実的でありながら効率的にシーンをレンダリングする手法の研究が数多くなされてきた。Sloan らの提案した Pre-computed Radiance Transfer (PRT) [1]は、大域照明計算における高コストで複雑な計算を事前に算出しておくことにより、写実的なシーンを高速にレンダリングすることに成功した。しかし、その対象は低周波照明環境のみであり、高周波照明成分には対応できなかった。その後、より汎用的なアルゴリズムにするべく PRT の分野で多くの手法が研究されてきており、全周波照明環境下でのシーンのレンダリングが可能になっている。

全周波照明環境下でのリアルタイムな照明計算手法の一つとして、照明を近似する基底関数として球面ガウス関数を用いた照明計算手法[2]が Wang らによって提案されている。しかしながら、この方法は描画されるシーンをより現実感のあるものにするために必要な物体相互の光の伝播である間接光の影響を考慮していない。

そこで本研究は、全周波照明環境下の静的なシーンにおける間接照明を考慮したレンダリング手法を提案する。

2. 関連研究

PRT[1]は光の伝播を前計算することにより、直接照明、間接照明、影等の複雑な照明効果を高速にレンダリングする手法である。しかしながら、この方法は低周波照明のものに限定されていた。Ng ら[4]は Wavelet 基底関数を用いることにより高周波の照明に対応した。しかしながら、この方法は直接照明効果しか考慮しておらず、間接照明効果は考慮されていなかった。

PRT 手法として、Tsai ら[3]は Spherical radial basis function (SRBF) を用いて高周波照明環境下におけるレンダリング手法を提案した。Wang ら[2]は球面ガウス関数を基底関数として用いてレンダリング方程式を近似することにより、全周波照明環境下でのリアルタイムの照明計算に対応した。しかしながら、これらの手法でも間接照明効果は考慮されていない。

そこで、本研究は、Wang らの手法を拡張して静的なシーンにおける間接光成分の計算をおこなう手法を提案する。

3. 提案手法

シーン内の物体上のある位置 x における一次間接光の輝度 L_1 は、次式で計算される。

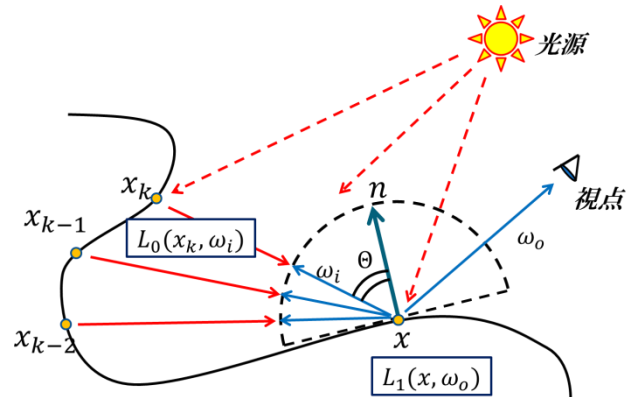


図 1 間接光による照明

$$L_1(x, \omega_o) = \int_{\Omega} L_0(x_k, \omega_i) f_r(\omega_i, \omega_o) \cos \theta d\omega \quad (1)$$

ここで、 Ω は半球上の方向、 L_0 は直接光の輝度、 f_r は BRDF、 $\cos \theta$ は位置 x の法線と入射ベクトル ω_i のなす角の余弦 (負の場合は 0)、 ω_o は出射ベクトルである。また、 x_k は間接光光源となる物体表面上の座標である。BRDF f_r や余弦 $\cos \theta$ 、直接光の輝度 L_0 は球面ガウス関数で表現される。球面ガウス関数は以下の式であらわされる Spherical radial basis function (球面放射基底関数) の一種である。

$$G(\mathbf{v}; \mathbf{p}, \lambda, \mu) = \mu e^{-\lambda(\mathbf{v} \cdot \mathbf{p} - 1)} \quad (2)$$

球面ガウス関数はローブの中心軸 \mathbf{p} とその鋭さ λ により決定される関数であり、ローブの大きさは振幅 μ によって決定される。また、方向 \mathbf{v} は球面ガウス関数の球面上のパラメータであり、この関数の定義域に相当する。

間接光は直接光の輝度計算によって求められた物体表面の輝度 L_0 のうち、位置 x に寄与するものを間接光の光源として扱って求める。そのため、位置 x に ω_i 方向から入射する直接光の輝度を求めるには、 x から ω_i 方向にレイを射出し、レイと物体表面との交点を求める必要がある。しかし、全周囲にレイを飛ばして交差判定を行う手法だと、計算コストが非常に大きくなってしまう。

そこで、提案手法では、シーン内の全頂点に対して頂点法線の方向を上向きとする半球面に、その頂点から見える物体からの距離を記録する。これをディスタンスマップと呼ぶ (図. 2 を参照。ここで \mathbf{p} は頂点、 \mathbf{n} は法線である)。こうして記録したデータは主成分分析を用いて圧縮したのち、テクスチャに格納しておく。

レンダリング時には、まず、Wang らの手法[2]を用いて直接光の計算を行いその結果をテクスチャに記録しておく。こうして記録された直接光成分の結果を用いて間接光の計算を行う。頂点毎に求めておいたディスタンスマップを用いて注目している位置に寄与する直接光成分を取得し、間

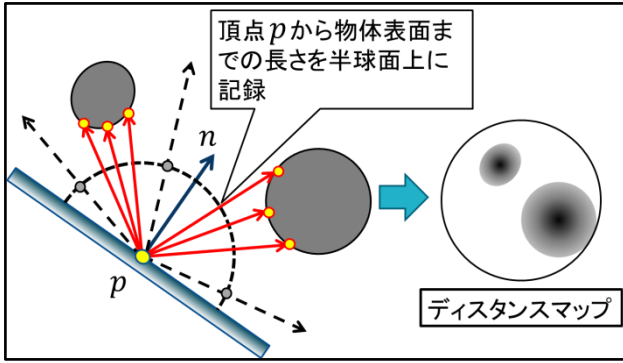


図2 ディスタンスマップ

接光の輝度を算出する。最終的な輝度値は、直接光の輝度と間接光の輝度を足し合わせることで求められる。

間接光計算処理は、オブジェクト空間内ではなく、スクリーン空間内で行われる。オブジェクト空間内で処理を行う場合、処理結果はスクリーン空間での処理に比べて正確になるが、シーン内の全ての物体表面の直接光計算の結果を保持しておく必要がありデータ容量の面で処理が困難になる。そこで、提案手法では間接光の計算をスクリーン空間で行う。

3.1 前計算

ある点における間接光を計算する際に、それに寄与する間接光の光源となる周辺物体の表面の位置を知る必要がある。そのために、本研究ではディスタンスマップを使って、周囲の物体表面の位置を表現する。ディスタンスマップはメッシュの頂点毎に格納され、三角形上でピクセル毎に補間される。

前計算として、各頂点におけるディスタンスマップを頂点法線方向を上向きとする半球面上に記録する。具体的には、シーン内の頂点ごとに、その頂点位置をカメラ座標の原点として、上下左右前後の六面の距離値をキューブマップテクスチャにレンダリングする。さらに、頂点の法線を上向きとする上半球面のみをキューブマップテクスチャから取得して正方形のテクスチャに変換する。

次に、主成分分析 (PCA) を使ってディスタンスマップを圧縮する。

$$DistanceMap_x(\omega) \approx \sum_{j=1}^{N_D} D_j(\omega) w_{x,j}^D \quad (3)$$

ここで、 ω は x から物体表面へ方向ベクトル、 N_D は主成分分析の項の数を表し、 D_j は方向ベクトル ω 方向の固有ベクトルである。また $w_{x,j}^D$ は主成分分析 (PCA) の係数である。現在の実装では $N_D = 48$ としている。固有ベクトル、PCA 係数はどちらもテクスチャとして格納され、レンダリング時に (3) 式を用いてディスタンスマップへと再構成される。

3.2 レンダリング

レンダリングは直接照明計算と間接照明計算に分けられる。

3.2.1 直接照明計算

直接照明計算では Wang らの手法 [2] を用いて直接光の輝度計算を行い、この時に得られるパラメータを使用して間接照明の輝度値を算出する。直接照明計算のパラメータはカメラ位置から見たシーンをテクスチャに描画することにより保存され間接光計算処理部に渡される。

Wang らは直接照明計算を鏡面反射成分と拡散反射成分に分けて計算している。提案手法でも同様に、間接照明計算を鏡面反射成分と拡散反射成分に分けて行う。間接照明計算において、拡散反射成分の計算は、直接光の拡散反射成分の輝度をそのまま間接光光源の輝度として使用できるが、鏡面反射成分の計算では、物体表面の反射特性である BRDF から間接光光源の輝度を計算する必要がある。そこで、直接光の拡散反射成分は輝度値を、鏡面反射成分は BRDF を求めることができるように輝度値ではなく鏡面反射成分の球面ガウス関数のパラメータを保存する。テクスチャに保存されるパラメータは以下の通りである。

- 位置、法線、深度値
- 三角形の面 ID、補間用の重心座標値
- 直接光の拡散反射成分の輝度値
- 直接光の鏡面反射成分の球面ガウス関数のパラメータ

これらをテクスチャのピクセルの色値として格納する。間接照明計算において、物体表面のある位置における直接照明計算結果を取得するには、その位置の座標値をスクリーンの左上を $(0.0, 0.0)$ 、右下を $(1.0, 1.0)$ とするスクリーン空間内の座標値に変換して上記のテクスチャにアクセスする。また、これらとは別に、シーン内の全三角形の頂点 ID をテクスチャとして格納する。

3.2.2 間接照明計算

間接光の輝度 L_1 を求めるため、(1) 式を離散化した以下の式を計算する。

$$L_1(x, \omega_o) = \sum_{k=1}^{N_s} L_0(x_k, s_k) f_r(s_k, \omega_o) \max(s_k \cdot n, 0.0) \quad (4)$$

ここで、 N_s はサンプリングレイ s_k の数である。 s_k は (1) 式の入射ベクトル ω_i に相当する。この (4) 式を用いて間接照明計算を行う。

まず、鏡面反射成分の計算では、位置 x における反射特性 BRDF の球面ガウス関数を、直接光の計算結果を格納したテクスチャより取得する。鏡面反射成分の間接光は、物体表面の BRDF のローブ形状に従って位置 x の輝度に影響を与える。BRDF を表す球面ガウス関数のローブ方向、つまり、球面ガウス関数の中心軸 p を中心に、鋭さ λ によって決定されるローブの範囲に入射する間接光が、位置 x にもっとも影響を与える成分となる。よって、この範囲に対して重点的にサンプリングすることにより、位置 x に対して影響力のある間接光光源である周辺物体の表面位置を効率よく求めることができる。また、拡散反射成分の計算は、反射特性が偏っている鏡面反射成分と異なり、法線方向を上向きとする半球面上に一律な反射特性を持っている。そこで、半球面上に一律にサンプリングして、間接光光源である周辺物体の表面位置を求める。

しかし、サンプリングレイを飛ばして周囲の物体表面との交点を求める方法だと処理コストが高くつくため、前計算で求めたディスタンスマップを利用して算出する。ディスタンスマップは頂点毎に記録されているため、三角形内部のある点でのディスタンスマップは、その位置の重心座標を重みとして用い、三頂点の各ディスタンスマップを補間して求める。

ディスタンスマップを利用して間接光光源となる周囲の物体の表面位置を求めるには、まず、位置 x から発射されるサンプリングレイを s とし、サンプリングレイ s によりディスタンスマップを参照して、 s 方向の一番近い物体表面の位置 x_k までの距離 l を求める。これを利用して位置 x_k は、以下のように算出できる。

$$x_k = \text{normalize}(s) \times l + x \text{-----}(5)$$

こうして求められた間接光光源となる位置 x_k における直接照明計算のパラメータを取得するため、次に、位置 x_k をスクリーン空間内の位置に変換する。これをインデックスにして位置 x_k の直接照明計算のパラメータを取得し、位置 x_k から位置 x への反射光の強さ L_0 を算出する。

反射光 L_0 の拡散反射成分の輝度値は、直接照明計算で求めた拡散反射成分の輝度値をそのまま用いる。反射光 L_0 の鏡面反射成分の輝度値は、位置 x_k の反射特性を表す球面ガウス関数のパラメータのうち、出射ベクトルを、 x_k から x の方向ベクトルに置き換えて再構成されたものを用いて計算する。

次に、求められた反射光の強さ L_0 と入射ベクトル s_k を入力とし、位置 x の間接光の輝度 L_1 を(4)式によって算出する。最後に、求められた間接光の輝度と直接光の輝度を足し合わせて位置 x の最終的な輝度を求める。

4. 結果

本手法でレンダリングされた結果画像を図 3、図 4 に示す。このドラゴンと箱のシーンは、ドラゴンの脚部や腹部が床面からの照り返しによる間接光によって緑に染まっていることを確認できる。また、赤い箱からドラゴンへの照り返しによってドラゴンの後頭部や首回り、前足の上部分が赤く染まっている。計算環境は、Intel Core i7 CPU 920 2.67GHz、3GB RAM、NVIDIA GeForce GTX 470 を搭載した PC である。直接光及び、間接光の輝度計算は GPU 上で GLSL を用いて行った。描画されているシーン全体は、三角形数 78,692、頂点数 39,283 で構成されており、その内訳は、ドラゴン 37,000 頂点、箱 602 頂点、床 1,681 頂点である。ディスタンスマップの容量は圧縮前が 2.4GB、圧縮後は固有値ベクトルデータが 3.0MB、PCA 係数データが 7.4MB となっている。間接照明計算時のサンプリングレイの数は 16 である。描画更新レートは 63fps であり、リアルタイムな描画速度を実現できた。本手法により、全周波照明環境下のシーンにおいて、間接照明を考慮したレンダリングを行うことができた。今後の課題として、間接光計算の高速化や動的なシーンへの対応が挙げられる。

参考文献

[1]P. Sloan et al., Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments, ACM Transactions on Graphics,

Vol. 21, No. 3, pp. 527-536, 2002.

[2]J. Wang et al., All-frequency rendering of dynamic, spatially-varying reflectance, ACM Transactions on Graphics, Vol. 28, No. 5, 2009.

[3]Y. Tsai et al., All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation, ACM Transactions on Graphics, Vol. 25, No. 3, pp. 967-976, 2006.

[4]R. Ng et al., All-frequency shadows using non-linear wavelet lighting approximation, ACM Transactions on Graphics, Vol. 22, No. 3, pp. 376-381, 2003.

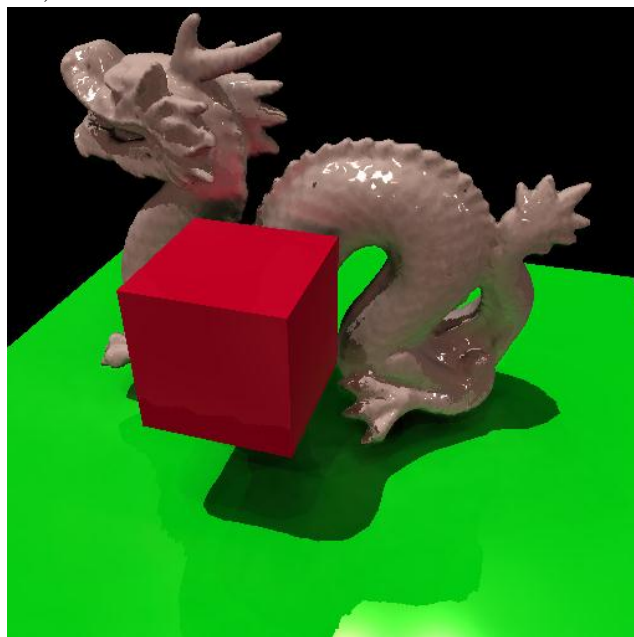


図 3 ドラゴンと箱のシーン

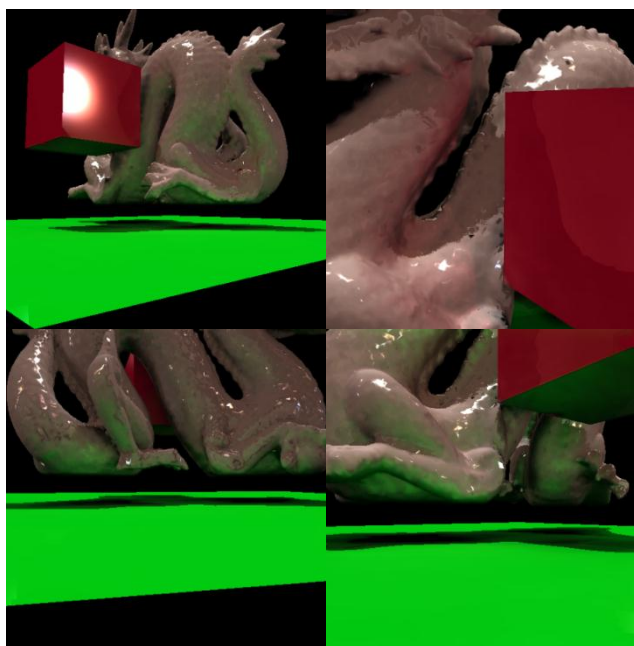


図 4 シーンの一部を拡大したもの