

2006年度 卒業論文

RGBの出力値操作によるモアレパターンの軽減

指導教員：渡辺 大地講師

メディア学部 ゲームサイエンスプロジェクト

学籍番号 M0103093

大村 邦福

2006年度 卒業論文概要

論文題目

RGB の出力値操作によるモアレパターンの軽減

メディア学部

学籍番号：M0103093

氏名

大村 邦福

指導
教員

渡辺 大地講師

キーワード

モアレパターン、モアレ、干渉縞、シェーダー、MEL

モアレパターンとは、縞模様のような周期的な模様を2つ以上重ね合わせた時に、それぞれの周期が干渉しあって現れる、大きな周期の模様のことである。身近な所ではテレビや新聞においても見ることが出来るが、これらは意図した映像・画像と異なった模様が現れてしまう為に、一般的には好ましくないものとして扱われている。

近年、3DCG を用いたコンテンツの制作が盛んに行われているが、本研究ではこれら3DCG に於いて生じるモアレパターンの原因となる模様のコントラストを、RGB の出力値を局所的に操作することによって小さくし、生じるモアレパターンを目立たなくする為の計算手法の考案を目的としている。

モアレパターンの発生原因の一つに、処理画素数と出力画素数との差異がある。3DCG では、オブジェクトの形状やスケールによって、オブジェクトに適応されたカラーマップなどの処理画素数と出力画素数との差異が動的に発生する為、モアレパターンが発生してしまう。これらのモアレパターンを効率的に軽減する為には、処理画素数と出力画素数との差異の動きに合わせ、動的に模様のコントラストを下げるよう RGB の出力値を操作する必要がある。

既存手法として、カメラとサンプリングポイントとの距離情報を基に RGB の出力値を操作し、モアレパターンの軽減を行う手法がある。しかし、既存手法はスケールの大きなオブジェクトには有効に作用するが、スケールが小さく移動や変形を伴うオブジェクトでは、サンプリングポイントとの距離が変化してしまう為に、良好にモアレパターンを軽減する事ができないという問題点がある。

本研究では、既存手法の問題点を改善する為に、角度によって引き起こされる処理画素数と出力画素数との差異に注目し、カメラ法線とサンプリングポイントの属するサーフィス法線との角度情報を基に、原因となる模様のコントラストを下げるよう RGB の出力値を操作し、モアレパターンを軽減する新たな手法を提案し、既存手法の問題点を改善した。また、Maya の MEL を用いてそれらを実装し、レンダリング結果を比較することでそれを証明した。

目次

第1章	はじめに	1
第2章	角度情報を用いたモアレパターンの軽減手法	12
2.1	基本方針	12
2.2	出力する模様のコントラストに関して	13
2.2.1	出力する模様のコントラストに関する問題点	13
2.2.2	問題への対処法	18
第3章	MELによる実装	20
3.1	本手法を実装する際のキーワード	20
3.2	ソース解析	21
3.3	「MoireDecrementShader」の使い方	22
第4章	本手法の有用性及び比較による検証	27
4.1	有効性の検証	27
4.2	既存手法との比較	31
4.2.1	既存手法の再現	31
4.2.2	比較検証1 -スフィア形状のオブジェクトの場合-	32
4.2.3	比較検証2 -プレーン形状のオブジェクトの場合-	36
4.3	まとめ及び今後の展望	37
	謝辞	41
	参考文献	42

第 1 章

はじめに

モアレパターン [1][2][3] は周期的な模様、つまり規則正しい繰り返し模様を複数重ね合わせた時に、それらの周期が干渉しあって大きな周期を形成することにより視覚的に発生する縞模様のことであり、略して「モアレ」や「干渉縞」「エリアシング」等とに呼称されている。語源はフランス語で波形模様を意味する「Moire」である。図 1.1 は、2 つの平行線の模様が干渉することで生じるモアレパターンの例である。

モアレパターンは、身近な所ではテレビや新聞においても見ることが出来る。テレビにモアレパターンが発生する理由は、テレビの走査線と被写物の規則性のある模様とが干渉しあうからである。新聞などの印刷にモアレパターンが出現する理由は、新聞の写真は網点で印刷されており、これと被写体の規則性のある模様とが干渉し合うからである。また、コンピューターによる画像処理に於いても、画像は画素とよばれる縦横に周期的に配置した点に分解して表現することから、印刷と同様なモアレパターンが発生する可能性があり、画像処理の過程で画素数の少ない画像を縮小・変形した場合にも発生しやすい。また、処理する画素数と出力する画素数が異なる場合にも発生し、ジャギーなどもこれに当たる。さらに、写真などをスキャナで入力する場合にも写真とスキャナの分解能の差次第でモアレの発生がありうる。これらモアレパターンの発生原理は、音のうなり現象 [4] と同様に考えることが出来る。うなりは波長の異なる 2 つの波が重なり合ったときに、

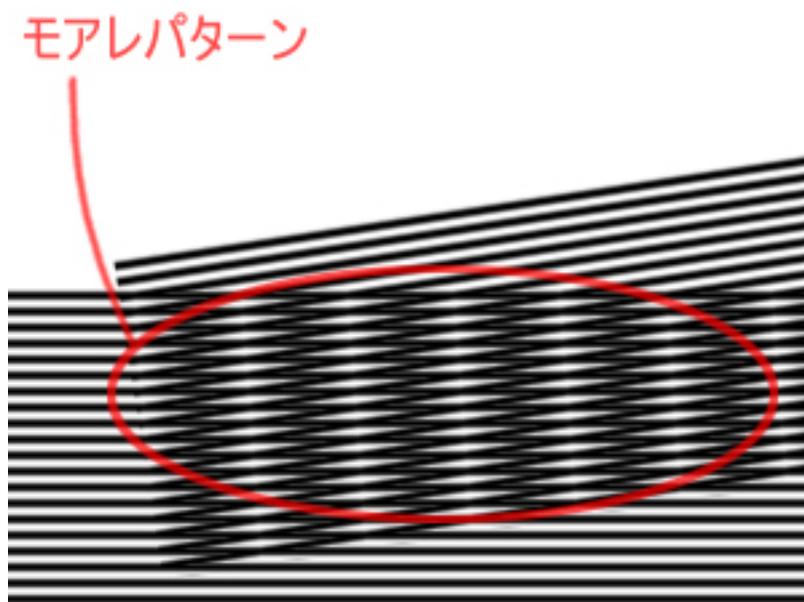


図 1.1: 2つの平行線模様から生じるモアレパターン

それぞれの周期が干渉しあって発生する。同様に、モアレパターンは2つの空間周波数のうなり現象であるといえる。

モアレパターンは、その特徴を用いて様々な測定にも利用されている。例えば高精度の位置決め用の位置センサの一種であるモアレスケール[5]では、わずかにピッチを変えた平行縞を重ねたときに発生するモアレが、二つの並行縞の相対変移よりも大きく移動することを利用し、変移を拡大して測定するものであり、似たような技術に、物体の微視的変位を二次元的に計測する、モアレ干渉法[6]や位相シフトモアレ法[7]という技術がある。また、立体の表面に二つの格子縞を投影したときに発生する干渉縞の形状(等高線)から物体の立体形状を得る方法を、モアレトポグラフィ(モアレ写真法)[8]といい、脊柱側弯症の診断等にも用いられている[9]。しかし、これらモアレパターンは、特に印刷や画像処理の分野に於いては、意図した画像や映像と異なった模様が現れてしまうことから、一般的には望まぬものとして扱われ、取り除く対象とされている。

本研究において題材とするのは、3DCGに於ける、処理画素数と出力画素数の違いによって生じるモアレパターンである。この原因によって引き起こされるモアレパターンは、処理画素数と出力画素数の差異が大きくなる程、干渉し合う波が増える為に、より強いモアレパターンが発生しやすくなる。図1.2、1.3、1.4は、処理画素数と出力画素数の違いによって生じるモアレパターンの例である。図1.2は処理画素数と出力画素数の比が1:1でつり合っている状態であり、図1.3、図1.4は処理画素数と出力画素数の比がそれぞれ1:0.6、1:0.3と差異が生じている状態であり、差異の大きい図1.4が、図1.3よりも強くモアレパターンが発生しているのが確認できる。3DCGは、オブジェクトの形状やスケールによって、これらのモアレパターンの原因となる処理画素数と出力画素数の違いを動的に生じさせてしまう。例えば図1.5のように、プレーン形状でスケールの大きなオブジェクトに周期的な模様のバンプマップ[10]を適応している場合、A点ではバンプマップによって処理される画素数と、出力される画素数との差異が小さい為に、モアレパターンは生じていない。しかし、オブジェクトがカメラに対して傾いている

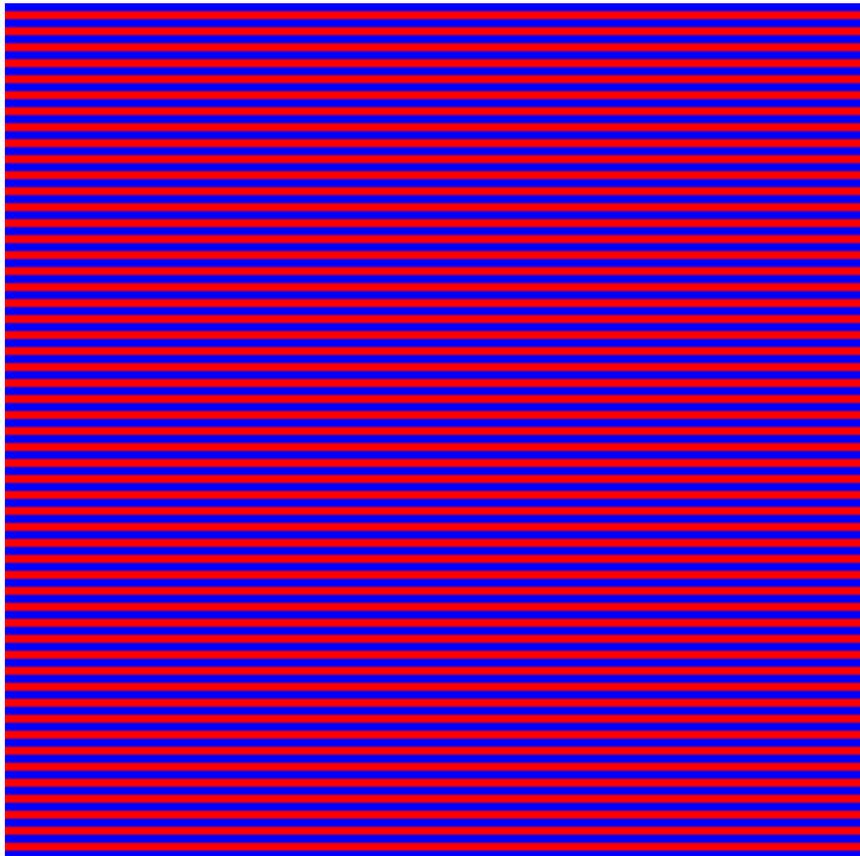


图 1.2: 处理画素数 : 出力画素数 = 1 : 1

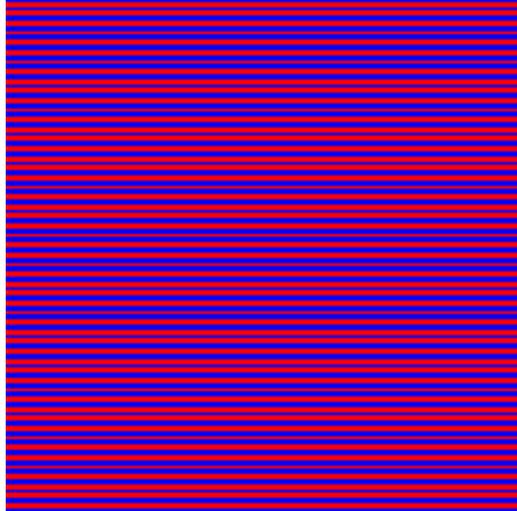


图 1.3: 处理画素数 : 出力画素数 = 1 : 0.6

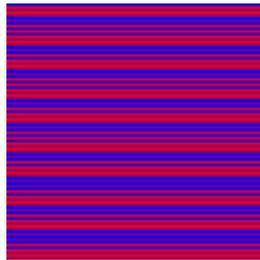


图 1.4: 处理画素数 : 出力画素数 = 1 : 0.3

為、B 点に近づくにつれてカメラとサンプリングポイントとの距離が離れ、1ピクセルあたりに占めるサーフィスの面積が増大する。よって、処理するバンプマップの画素数と、出力する画素数との差異が動的に大きくなってゆき、モアレパターンが発生してきているのが確認できる。

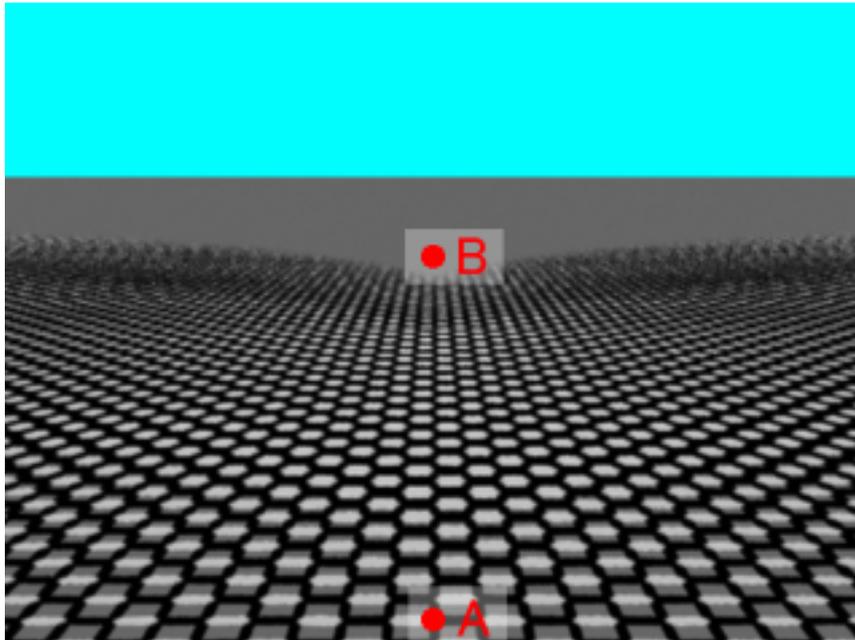


図 1.5: プレーン形状でスケールの大きいオブジェクト

同じく図 1.6 のように、スフィア形状のオブジェクトに周期的な模様のバンプマップを適応している場合も、A 点ではオブジェクトのサーフィスがカメラに対して正面を向いている為に、バンプマップによって処理される画素数と、出力される画素数との差異が小さく、モアレパターンは発生していないが、B 点に近づくにつれてサーフィスのカメラに対する角度が大きくなってくる為、1ピクセルあたりに占めるサーフィスの面積が増大する。よって、処理するバンプマップの画素数と、出力する画素数との差異が動的に大きくなってゆき、モアレパターンが発生してきているのが確認できる。

このように、3DCG ではオブジェクトの形状やスケールによって、モアレパターンを生じさせる原因を作り出してしまうことがある。本研究は、これらの要因に

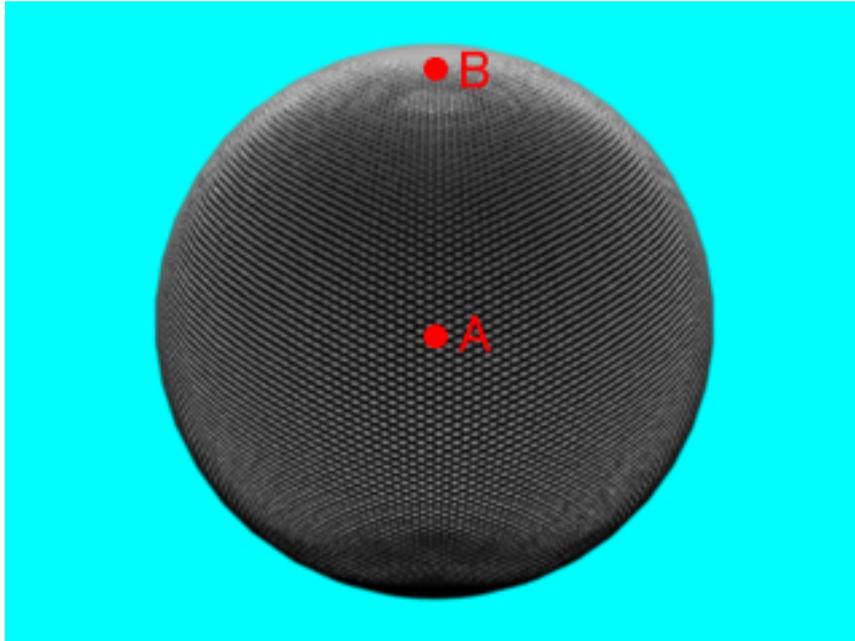


図 1.6: スフィア形状のオブジェクト

よって生じるモアレパターンを軽減する為の手法を提案するものである。

処理画素数と出力画素数の違いによって生じるモアレパターンを軽減させる為の研究には、ジャギーを軽減する為のアンチエイリアシング [11][12] があげられる。アンチエイリアシングはポストエフェクトによって、サンプリングポイントとその周囲の色情報を基に、それらの中間色を配置することで模様をぼかし、モアレパターンを目立たなくする技法である。しかし、本研究で対処すべきモアレパターンを効率的に軽減するには、処理画素数と出力画素数の差異の動きに合わせて、模様のコントラストの大きさを動的に変化させるように RGB の出力値を操作する必要がある。つまり、処理画素と出力画素の差異が大きくなる程にモアレパターンがより強く発生する為、それに比例させて根本的な原因である模様自体のコントラストをより小さくしてゆくのである。模様のコントラストを小さくすると、その干渉によって発生するモアレパターンのコントラストも小さくなるために、モアレパターンは軽減される。しかし同時に、元来の模様も失われてしまうが、この種のモアレパターンが発生する箇所では、少ない出力画素数に対して

多くの処理画素が割り当てられているので、元々模様ははっきりと表示されない為、弊害は少なく、効率的且つ演出性を損なわずにモアレパターンを軽減することが可能である。このような処理を行うのに適した手法として、カメラとサンプリングポイントとの距離情報を基に、模様のコントラストを下げゆく手法がある[13][14]。地面など背景の表現に用いられる、図 1.5 の様な巨大なオブジェクトの場合、カメラとサンプリングポイントとの距離が大きくなるにつれて処理画素数と出力画素数の差異が大きくなってゆく。その為、カメラとサンプリングポイントとの距離の大きさに応じて、モアレパターンを発生させる模様のコントラストを小さくしてゆくことで、効率的にモアレパターンを軽減することができる。

図 1.7 はカメラとサンプリングポイントとの距離情報を基に、バンプマップによって作られる模様のコントラストを操作して、モアレパターンを軽減させたものである。処理を行っていない図 1.8 と比較すると、モアレパターンが軽減されているのが確認できる。カメラとサンプリングポイントとの距離情報を基に模様のコントラストを操作する既存手法は、図 1.7 の例のように、背景オブジェクトの様な巨大なオブジェクトに対しては有効であるが、キャラクターやスケールの小さいオブジェクトには不向きである、という問題点がある。この手法に於いて処理を実行する条件の設定は、3DCG に於いて、カメラのピント（焦点）を再現する手法の処理条件の設定に近似していると言える。その為スケールの小さな被写体を映す際には、カメラの移動やオブジェクトの移動・変形に伴ってピントがズレてしまうのと同じように、モアレパターンを軽減する為の、模様のコントラスト操作の処理範囲がズレてしまい、良好な結果が得られなくなってしまうのである。ただし、被写界深度を深くすればピントはずれにくくなるのに対して、モアレパターンの軽減処理に於いては、処理範囲を大きくすると、同一面積あたりの模様のコントラストの変化量が小さくなってしまふ為、全体的に模様が薄くなり、良好な結果が得られなくなってしまう。

図 1.9 は、スフィア形状のオブジェクトのスケールに合わせて処理範囲を設定し、モアレパターンを軽減しているものである。そして図 1.10 は、図 1.9 のオブ

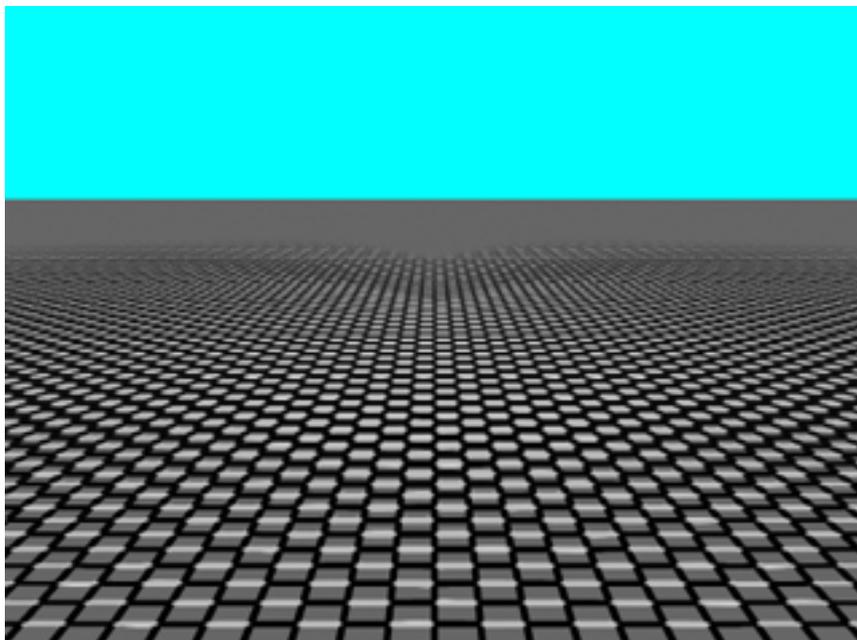


図 1.7: 距離情報を基にコントラストを操作する処理を施した状態

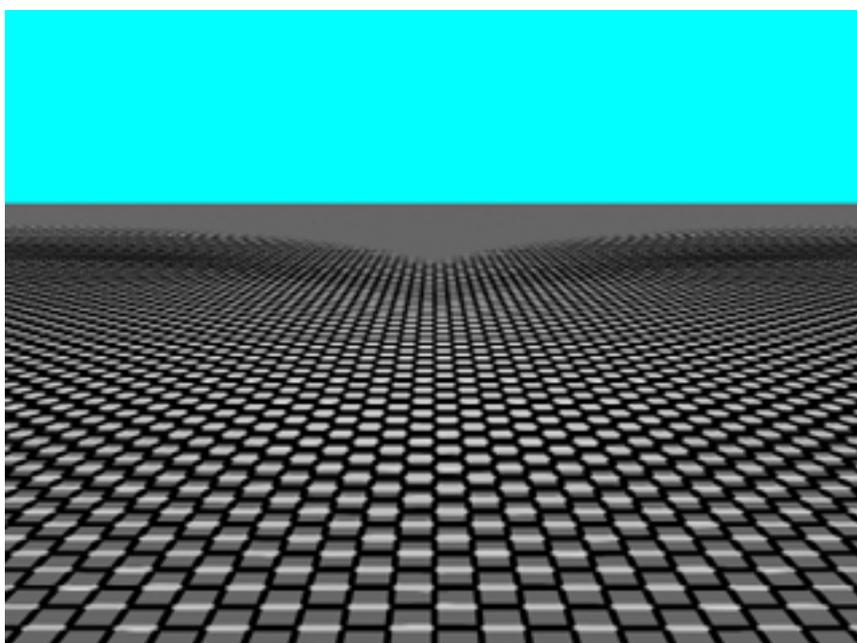


図 1.8: 未処理の状態

ジェクトを画面奥（Z軸方向）に向かって移動させたものである。カメラとサンプリングポイントとの距離が変化した為、設定した操作範囲からオブジェクトがズレて、必要以上にコントラストを小さくする処理が作用し、模様自体が消失してしまっている。

本研究は、距離情報ではなく角度情報を基準として模様のコントラストを小さくし、モアレパターンを軽減させるよう、RGBの出力値を操作する手法を提案するものである。それによって、既存手法では対応することのできなかった、カメラの移動やオブジェクトの移動・変形に於いても、モアレパターンを効率的に軽減することに成功した。また、それらを既存手法によるレンダリング結果と、本手法によるレンダリング結果とを比較検証することで、本手法の有用性を示した。



図 1.9: オブジェクトが適切な処理範囲に収まっている状態

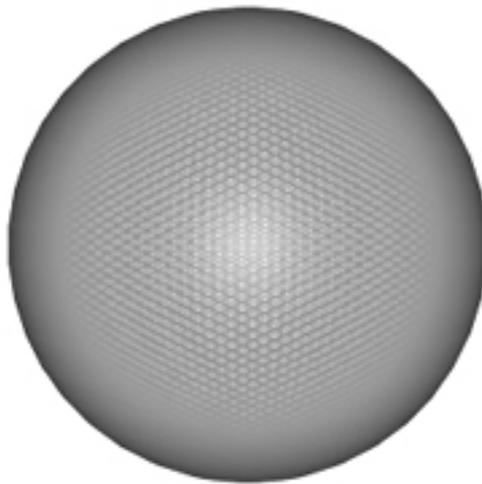


図 1.10: オブジェクトの移動に伴い適切な処理範囲から外れてしまった状態

第 2 章

角度情報を用いたモアレパターンの軽減手法

2.1 基本方針

角度情報の角度とは、具体的にはカメラの法線と、サンプリングポイントの属するサーフィスの法線との内角である。

図 2.1 は、カメラと任意のサーフィス上に存在するサンプリングポイントとの位置関係を示したものに、カメラからサンプリングポイントへ向かうカメラ法線と、サンプリングポイントの属するサーフィスの法線を書き加えた図である。図中に示したように、カメラ法線の単位ベクトルを

$$\vec{C} \tag{2.1}$$

サンプリングポイントの属するサーフィス法線の単位ベクトルを

$$\vec{N} \tag{2.2}$$

とすると、その内角の値 θ は、内積を求める以下の式 2.3 で求められる。

$$\vec{C} \cdot \vec{N} = \cos \theta \tag{2.3}$$

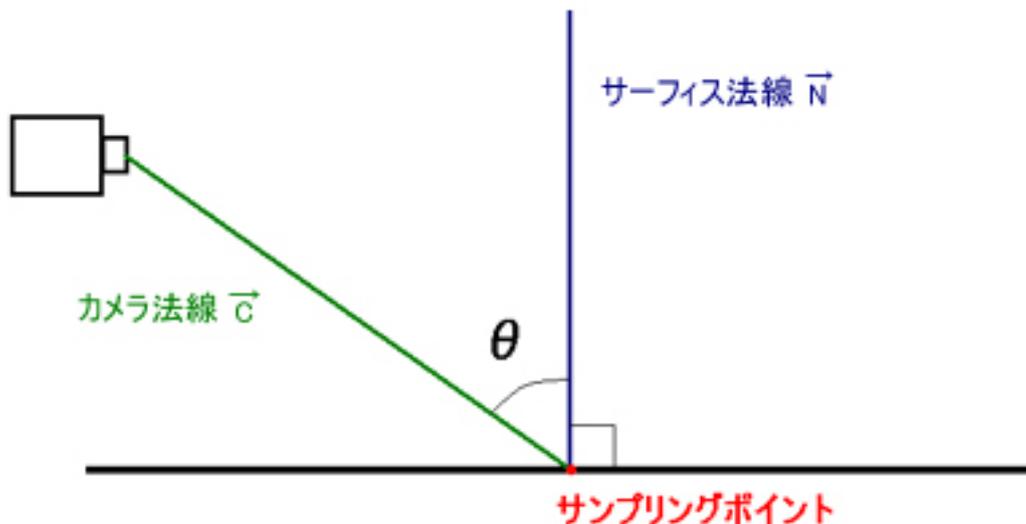


図 2.1: カメラ法線とサーフィス法線の内角

$\cos \theta = 1$ の時、 $\theta = 0^\circ$ となり、カメラはサンプリングポイントの属するサーフィスに対して正面を向いていることになる。一方 $\cos \theta = 0$ の時、 $\theta = 90^\circ$ となり、カメラはサンプリングポイントの属するサーフィスに対して平行となる角度を向いていることになる。カメラとの角度が大きくなるにつれて、処理画素数と出力画素数の差異が大きくなるので、模様のコントラストの強さを Y とすると、 θ が ($0^\circ \leq \theta \leq 90^\circ$) の範囲で、 $Y = \cos \theta$ の式を満たす Y の値の変化をグラフで示したものが図 2.2 であるが、このように模様のコントラストの強さ Y を、 $\cos \theta$ の値に付随する形で変化させてゆくことで、モアレパターンを効果的に軽減させることができる。尚、 θ が ($0^\circ \leq \theta \leq 90^\circ$) の範囲外の値をとる場合は、そのサンプリングポイントはカメラからは写らない角度にある為に考慮しないものとする。

2.2 出力する模様のコントラストに関して

2.2.1 出力する模様のコントラストに関する問題点

本手法では前節で述べたように、カメラ法線とサンプリングポイントの属するサーフィス法線との内積の値に伴い模様のコントラストを小さくするように RGB

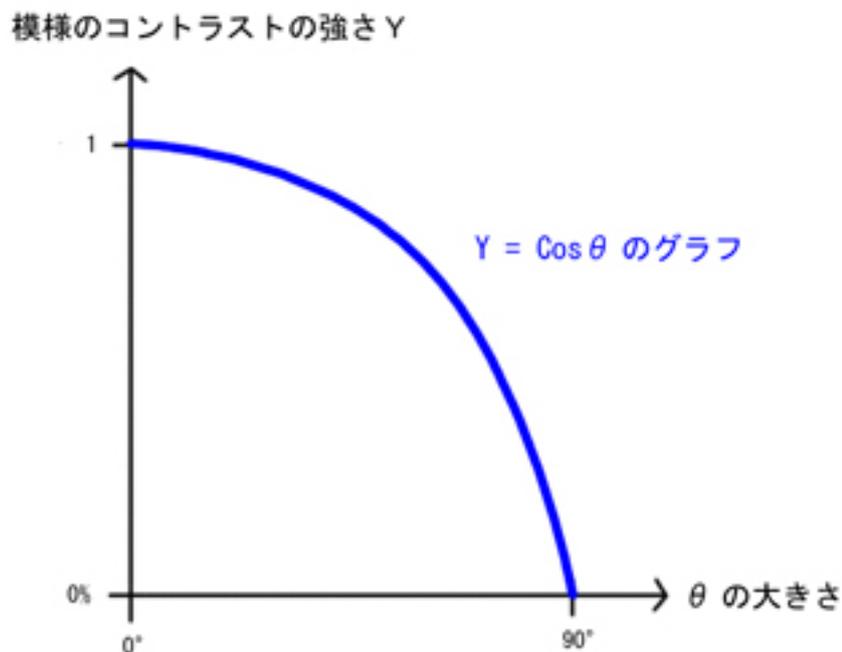


図 2.2: $Y = \cos \theta$ のグラフ

の出力値を操作する為、スフィア形状のオブジェクトのように、ある程度メッシュが細かく、カメラとの角度が1つのオブジェクト内で0度から90度まで徐々に変化する形状のオブジェクトに対しては有効に作用する。しかし、プレーン形状のオブジェクトのように、カメラとの角度があまり変化しない直線的なオブジェクトでは、模様のコントラストを操作できる範囲が狭まる為、処理画素数と出力画素数の動的な変化に合わせた効果的なモアレパターンの軽減処理を行うことが出来ない。図 2.3、図 2.4、図 2.5 にその具体例を示した。

図 2.3 は、プレーン形状のオブジェクトに一樣なタイル模様のバンプマップを適応した状態のオブジェクトを、比較の為、画面左側には何も処理を行っていない状態でレンダリングしたものを表示し、画面右側には、単純に内積の値に伴って模様のコントラストを操作する処理を適応してレンダリングしたものである。画面左側では手前に位置する図中の A 点から、奥に位置する B 点に近づくに従ってモアレパターンが発生してきている。一方画面右側では、モアレパターンの軽減は

確認できるものの、A点とB点との間でのコントラストの変化量が少ない。また、処理画素数と出力画素数の差が小さく、モアレパターンが発生しないA点においても、模様コントラストが下がってしまっている。モアレパターンの発生しない箇所にまでコントラストを小さくする効果が及んでしまえば、意図した効果を再現する事が出来ない為に好ましくない。これらの問題はプレーンの形状とスケールに起因するものである。

図2.4は、図2.3のオブジェクトとカメラ、及びA・B地点との位置関係を、水平真横側からの視点で示したものである。この図から分かる通り、A点でのカメラ法線とサーフィス法線との角度を θ_1 、B点でのカメラ法線とサーフィス法線との角度を θ_2 とした時、この形状のオブジェクトでは θ_1 と θ_2 の変化量、つまりコントラストの変化量と同義である $|\theta_1 - \theta_2|$ の値が小さい。しかし、それに反してオブジェクトのスケールが非常に大きい為に、A点とB点において、処理画素数と出力画素数の差異が大きく異なっている。よって、コントラスト操作の処理によるコントラストの変化量と、処理画素数と出力画素数の差異の大きさのバランスが崩れ、効率的なモアレパターンの軽減が出来ない。

また、任意のカメラ法線とサーフィス法線との角度を $\theta(0^\circ \leq \theta \leq 90^\circ)$ とし、模様コントラストの強さを Y とした、 $Y = \cos \theta$ の式をグラフで表した場合、このオブジェクトに大して適用されるコントラスト操作の処理の範囲は、図2.5の θ_1 から θ_2 間の赤色にマークされた範囲となる。このグラフからも分るように、画面手前のA点で既にコントラストが少し下げられてしまっている状態であり、画面奥のB点では逆にコントラストが下がりきっていない状態となっている。

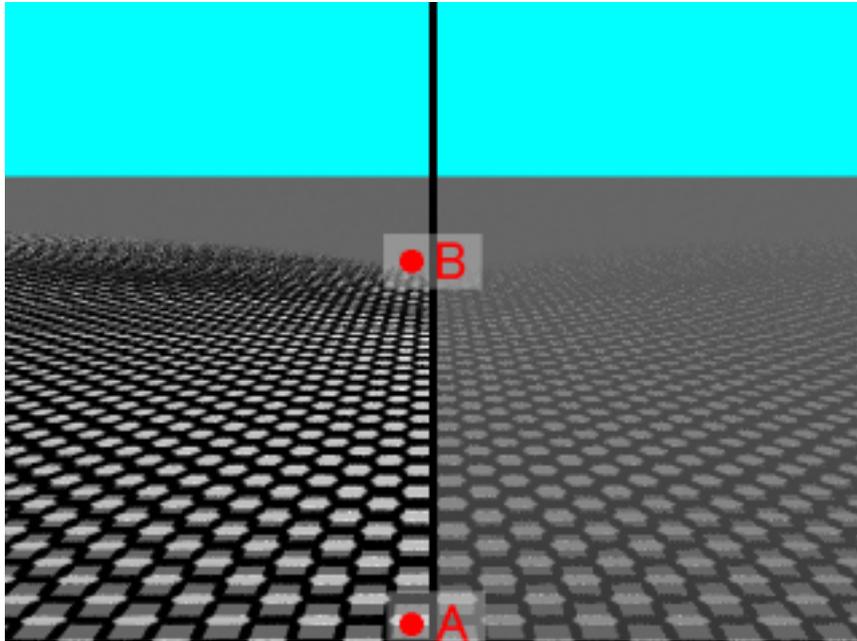


図 2.3: 内積値に正比例させコントラストを下げる処理を施した状態 (右側)

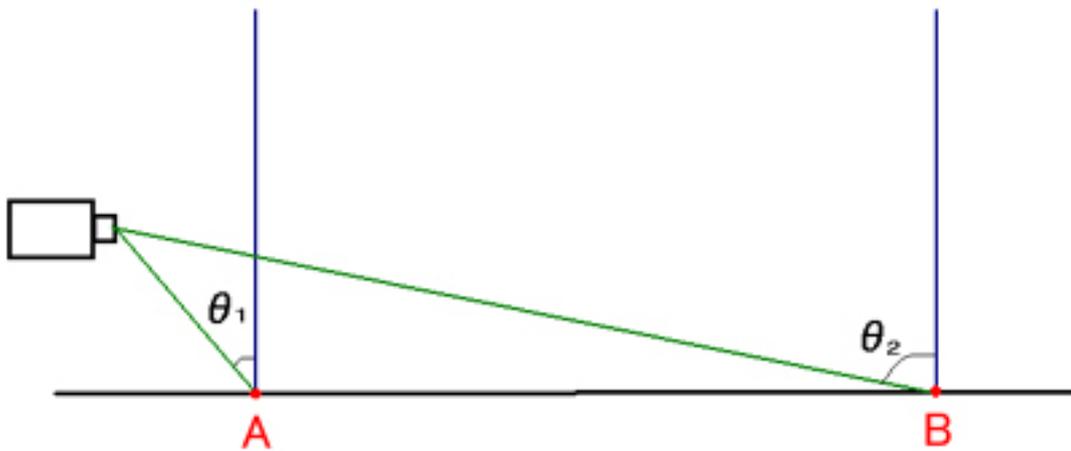


図 2.4: カメラ及び A 点、B 点を真横から見た状態

模様のコントラストの強さ Y

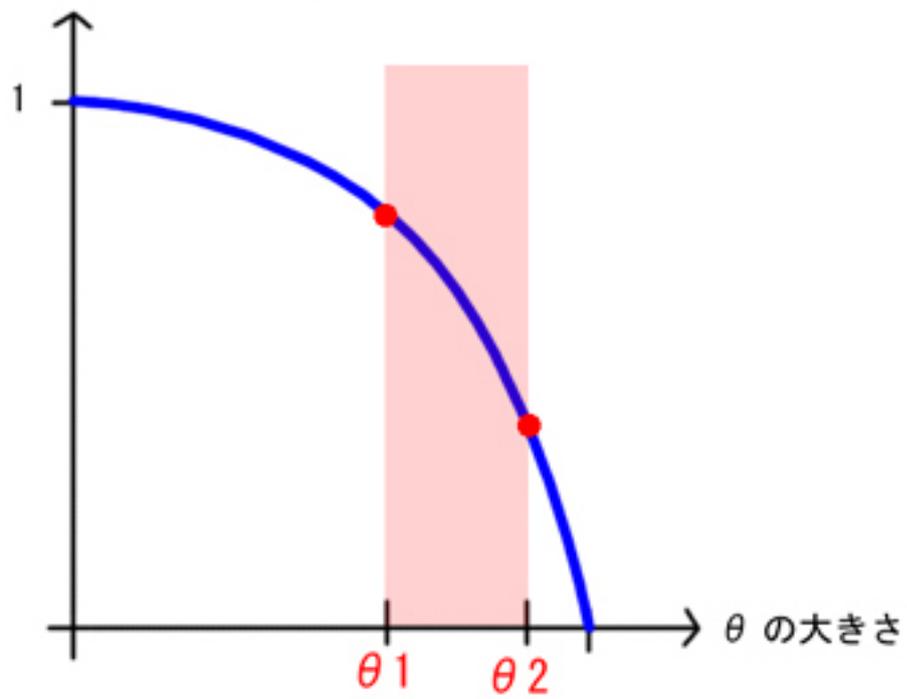


図 2.5: Y の値がとり得る範囲

2.2.2 問題への対処法

これらの問題への対処として、 θ_1 と θ_2 間に於いて、コントラストの強さ Y が 0 から 1 まで変化するように線形変換を用いることとした。

θ_1 の時のコントラストの強さ Y は $\cos \theta_1$ であり、同様に θ_2 の時の Y は $\cos \theta_2$ である。この時、任意の角度 θ ($\theta_1 \leq \theta \leq \theta_2$) の時の Y を $\cos \theta$ とすると、以下の式 2.4 を適用することで、図 2.6 のように、 Y の値が θ_1 から θ_2 の間で 0 から 1 まで変化する。尚、任意の角度 θ が θ_1 より小さい場合、 Y の値は常に 1、また θ_2 より大きい場合には、 Y の値は常に 0 をとるものとする。

$$Y = \frac{(\cos \theta - \cos \theta_2)}{(\cos \theta_1 - \cos \theta_2)} \quad (2.4)$$

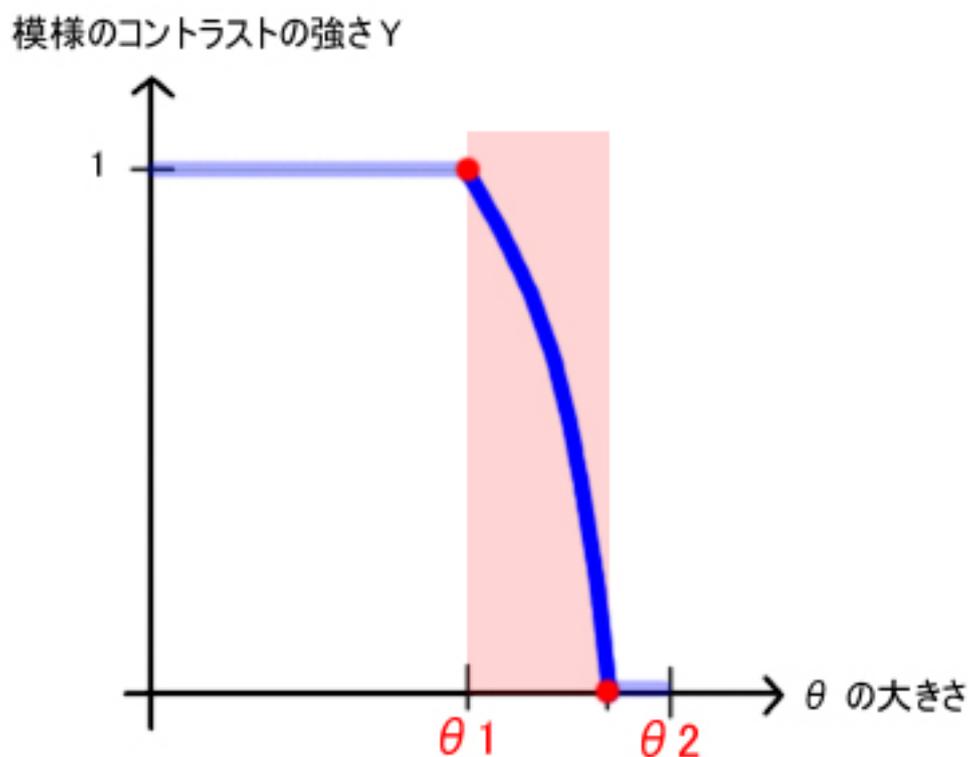


図 2.6: 線形変換を施した $Y = \cos \theta$ のグラフ

線形変換を行う際の θ_1 及び θ_2 の値は、モアレパターンを軽減する為の処理の、開始角度と終了角度として、対象となるオブジェクトの形状によってユーザーが任意に変更できるものとする。そうすることで、さまざまな形状のオブジェクトに対応し、同時に処理範囲を指定できることから演出性の面でも汎用性が高まる。

これらの処理を適応してレンダリングしたものが図 2.7 である。比較の為に、図の中央より左側に線形変換の処理を適応していない状態でのレンダリング結果を表示してある。左側と比べ線形変換の処理を行った右側では、A 地点に於いて模様のコントラストが高い状態で保たれており、B 地点に近づくに従ってコントラストが小さくなってゆき、効率的にモアレパターンを軽減しているのが見て取れる。

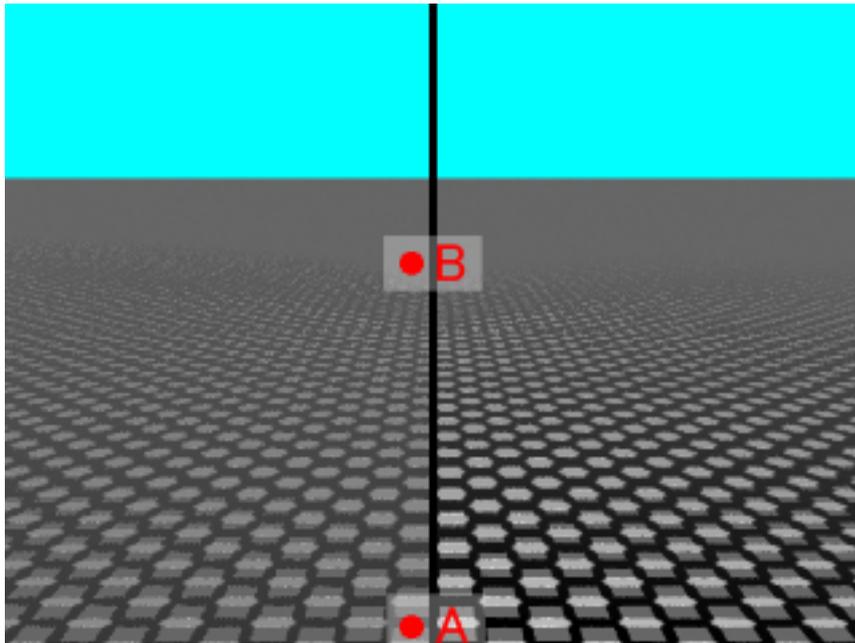


図 2.7: 線形変換を適用した内積値に比例させコントラストを下げる処理を施した状態（右側）

第 3 章

MEL による実装

3.1 本手法を実装する際のキーワード

本手法の有用性を検証する為に、本研究では Maya の MEL を用いて本手法を実装することを試みた。本手法の実装に於いてキーワードとなるのは、以下の要素である。

- 任意の処理開始角度と終了角度の入力
- カメラ法線とサンプリングポイントの属するサーフィス法線との内積計算
- 線形変換された内積値を基に模様コントラストを小さくするよう RGB の出力値を操作

まず、開始角度と終了角度の指定に関しては GUI を設け対応した。また、内積計算は Maya の Utility の機能によって計算した。そして RGB の出力値の操作に関しては、通常のシェーディング計算を行わず、カスタムシェーダーを作る際に用いられる `surfaceShaderNode` を土台として、パラメーターの異なる 2 つのベースマテリアルの RGB の出力値を、線形変換された内積値から算出した比率でブレンドすることで実現した。ユーザーは、モアレパターンの原因となる模様コントラストが、高い状態のマテリアルと低い状態のマテリアルの 2 つを準備しなければならない。その為、作業はやや煩雑なものとなるが、通常の Maya の操作でユー

ザーが任意にベースマテリアルを設定できるこの手法では、カラーマップやアルファマップ、グロウマップ、バンプマップ等、Maya でマテリアルに設定することのできる、多くのマップに起因するモアレパターンへの対処を可能としている。ただし、ディスプレイメントマップ [15] に関しては、モデルの形状自体が変化してしまう為に非対応である。

3.2 ソース解析

MEL による実装時に用いた、RPG 出力値のブレンドアルゴリズムを解説する。ベースとなるマテリアルの RGB 出力値をそれぞれ、 A, B とし、ブレンドされたマテリアルの RGB 出力値を C とする。尚、 A はモアレパターンの原因となる模様のコントラストが最大値をとるマテリアルであり、 B は最小値をとるマテリアルであるものとする。そして、カメラ法線とサンプリングポイントの属するサーフィス法線との内角を θ 、処理開始時及び終了時の角度をそれぞれ θ_1, θ_2 とする。この時、処理画素数と出力画素数の差異に合わせて模様のコントラストを小さくするよう、RGB 出力値をブレンドし、効率的にモアレパターンを減少させるための処理は以下のアルゴリズムによって実現できる。

```
P = cos θ
if (P > cos θ1){
    C = A
}else if(P < cos θ2){
    C = B
}else{
    C = A * ((P - cos θ2)/(cos θ1 - cos θ2))
      + B * (1 - ((P - cos θ)/(cos θ1 - cos θ2)))
}
```

\cos の値は MEL では、*samplerInfoNode* の *facingRatio* によって得ることができる。IF 文の条件式の中では、カメラ法線とサンプリングポイントの属するサーフィス法線との内角 θ が、設定した処理の開始角度及び終了角度の範囲内かどうか

を判別している。 θ が θ_1 より小さければ、処理が開始される前の角度であるので、最終的な RGB の出力値である C の値には常に、コントラストが最大の状態である A の値を適用する。同様に θ が θ_2 より大きければ、処理が終了した後の角度であるので、 C の値には常にコントラストが最小である B の値を適用する。そして、 θ が θ_1 と θ_2 の間の角度である時、 C の値は P に線形変換を適用して得られた値を A と B のブレンド比として用いることで、 θ の値が大きくなるにつれて、 C の値を A の値から B の値へと動的に変化させてゆくことを可能としている。尚、RGB の出力値である C の値は、Maya では *outColor*・*outGlowColor*・*outTransparency*・*outMatteOpacity* のそれぞれのアトリビュートに値を設定する必要があるので、同様のアルゴリズムを用いたエクスペッションをそれぞれに接続する。

これらの処理を行うことで、2つのベースマテリアルを、カメラ法線とサンプリングポイントの属するサーフィス法線との内角によって、動的にブレンドした値を出力するマテリアルが作成できる。その2つのベースマテリアルに、模様のコントラストが高い状態と低い状態を設定することで、ブレンドされたマテリアルが、処理画素数と出力画素数の差異に起因するモアレパターンを効率的に軽減するマテリアルとなるのである。

本研究で、MEL によって提案手法を実装したモアレパターン軽減の為のシェーダーを、「MoireDecrementShader」と呼ぶこととする。

3.3 「MoireDecrementShader」の使い方

本研究に於いて実装したシェーダーである「MoireDecrementShader」の、Maya 上での使い方を説明する。

1. MoireDecrementShader の設定ウインドウ起動

図 3.1 のように、Maya の CustomShader の MDS アイコンをクリック、もしくはスクリプトエディターにプログラムを入力して右 Enter キーを押し、MoireDecrementShader の設定ウインドウを起動。

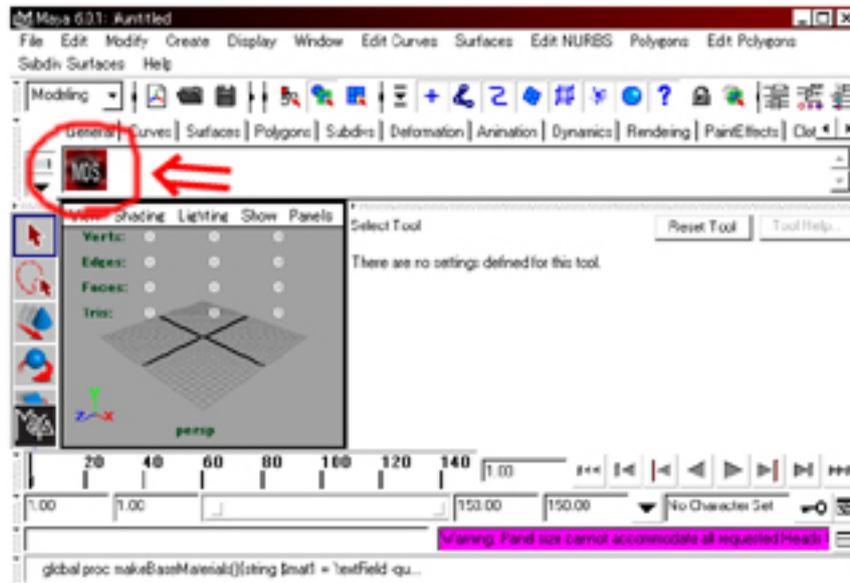


図 3.1: ツールの起動

2. ベース材料の生成

2つ立ち上がるウインドウのうち、「set Material」ウインドウに於いて、ベースとなる2つの材料のノード名をそれぞれ入力し、「Decision」ボタンを押す。すると図 3.2 のように、設定した名前の材料とセットをシェーディンググループに接続した状態で生成し、Hypershade 上でも確認・編集が行えるようになる。

3. ベース材料の設定

生成されたベース材料は通常の Maya の操作で自由にパラメーターを設定することができるので、AttributeEditor や HyperGraph を用いて材料の設定を行う。この時、材料の生成の際に「1st material」として設定した材料には、モアレパターンを引き起こす要因となる模様のコントラストが最大値をとる状態で設定し、「2nd material」として設定した材料には、逆に最低値をとる状態で設定する。図 3.3 では、生成した「1st material」に、Hypershade 上でバンプマップを適用した様子である。

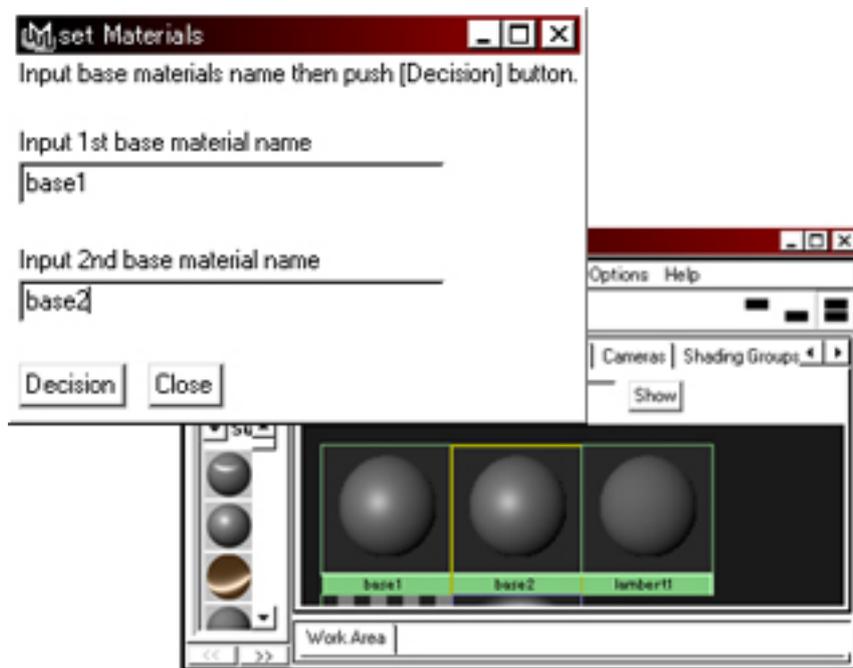


図 3.2: ベースマテリアルの生成

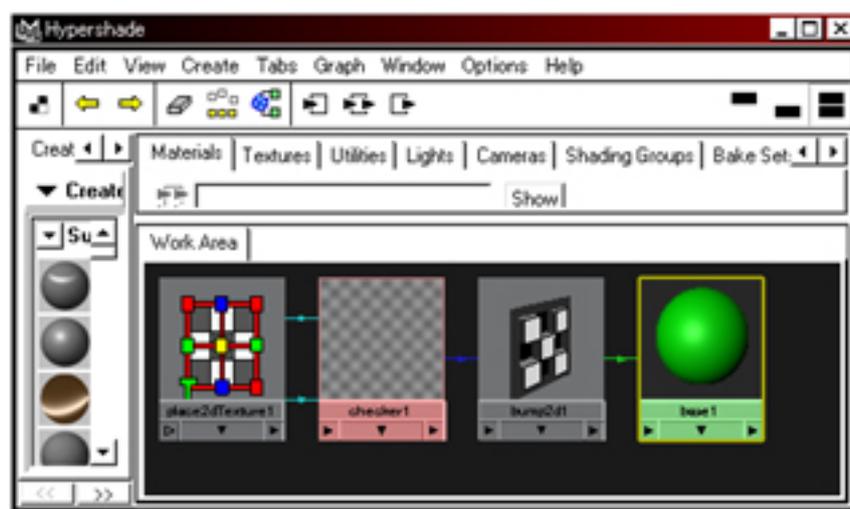


図 3.3: ベースマテリアルの設定

4. 処理の開始角度と終了角度の設定

「set MoireDecremetAngle」を立ち上げ、ベース材料をブレンドする処理の開始角度と終了角度を設定する。この時、開始角度は終了角度よりも小さい値を入力する。尚、開始角度に終了角度よりも大きい値が入力された場合、開始角度と終了角度の値を入れ替えて認識する仕様となっている。また、ベース材料をブレンドして生成する材料、つまり最終的にオブジェクトにアサインする材料のノード名を入力する。角度設定と名前が入力が終了したら「Decision」ボタンを押すと、設定した名前のブレンド処理の施された材料を生成する。図 3.4 は「set MoireDecrementAngle」へ値を入力した様子と、生成したブレンド材料の GraphNetwork を表示したものである。

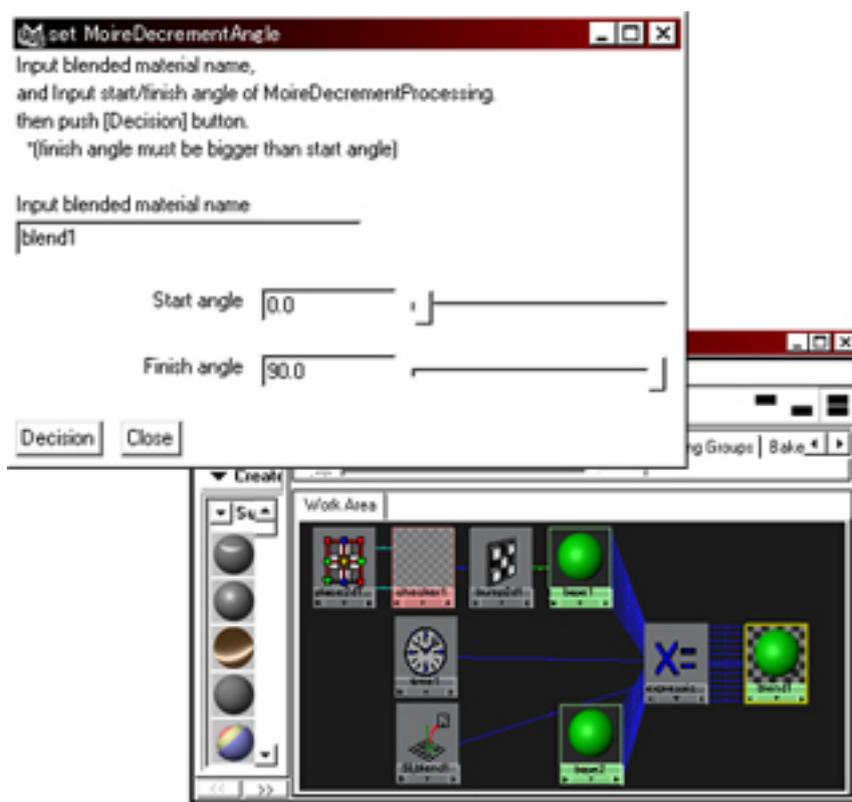


図 3.4: 処理範囲の設定

5. 生成したブレンドマテリアルについて

生成したブレンドマテリアルは、通常の Maya の操作でオブジェクトにアサインすることができる。尚、ブレンドマテリアルを直接編集する事はできないが、生成済みのベースマテリアルは直接編集することができ、ブレンドマテリアルに自動的に反映を行う。ただし、処理の開始角度及び終了角度を変更したい場合は、「set MoireDecrementAngle」を再び立ち上げ、角度とノード名を変更した上で「Decision」ボタンを押し、新しくマテリアルを生成しなおす必要がある。図 3.5 は、オブジェクトにブレンドマテリアルを適用した様子である。制作に使用した MAYA はカスタムシェーダーのプレビューには対応していないバージョンである為に、真っ黒な状態が表示されてしまいが、レンダリング時には正常に色が出力することができる。

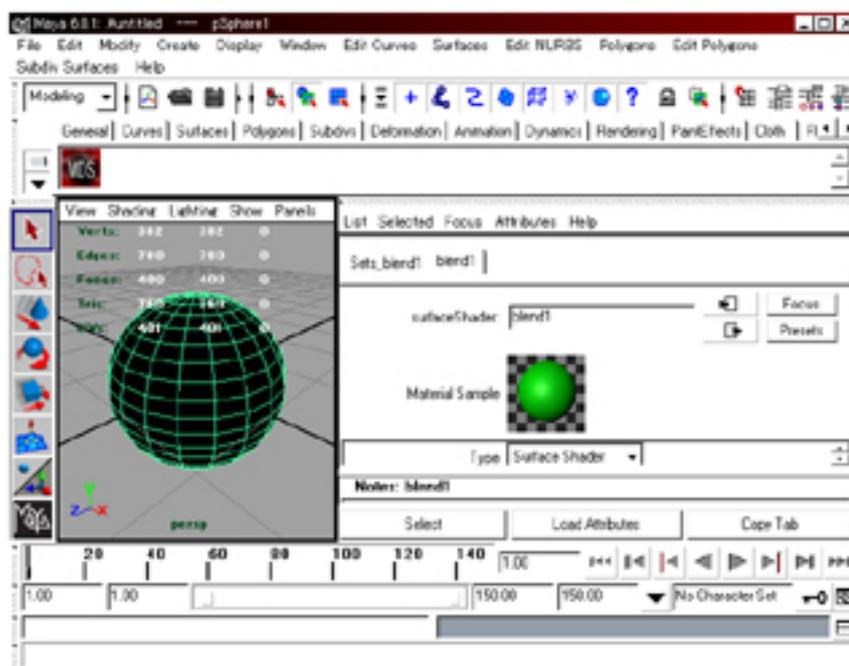


図 3.5: 処理範囲の設定

第 4 章

本手法の有用性及び比較による検証

4.1 有効性の検証

本研究で提案する手法を実装した「MoireDecrementShader」を用いて、本手法の有用性を検証した。

図 4.1 は、スフィア形状のオブジェクトに周期的な模様をもつカラーマップを適用し、カメラ空間座標の Z 軸方面に対して平行に DirectionallyLight を当て、レンダリングした結果である。オブジェクトの輪郭に近づくにつれて、オブジェクトのサーフィスのカメラに対する角度が大きくなってゆく為、1 ピクセル当たりに表示するサーフィスの面積が増えるので、処理画素数と出力画素数の差異が大きくなり、モアレパターンが発生しているのが見て取れた。それに対して図 4.2 は、同じ条件のオブジェクトに本手法を適用したものである。オブジェクトの輪郭に近づくにつれて模様のコントラストを小さくするよう、RGB の出力値が操作され、図 4.1 と比較し、モアレパターンが軽減されているのが確認できた。

図 4.3 では、カラーマップの例と同一の条件で、カラーマップの代わりにバンプマップを適用してレンダリングしたものであり、同様にオブジェクトの輪郭付近でモアレパターンが発生しているのが見て取れた。それに対して図 4.4 は、本手法を適応したものであり、カラーマップの例と同様にモアレパターンの軽減に効果的に作用しているのが確認できた。

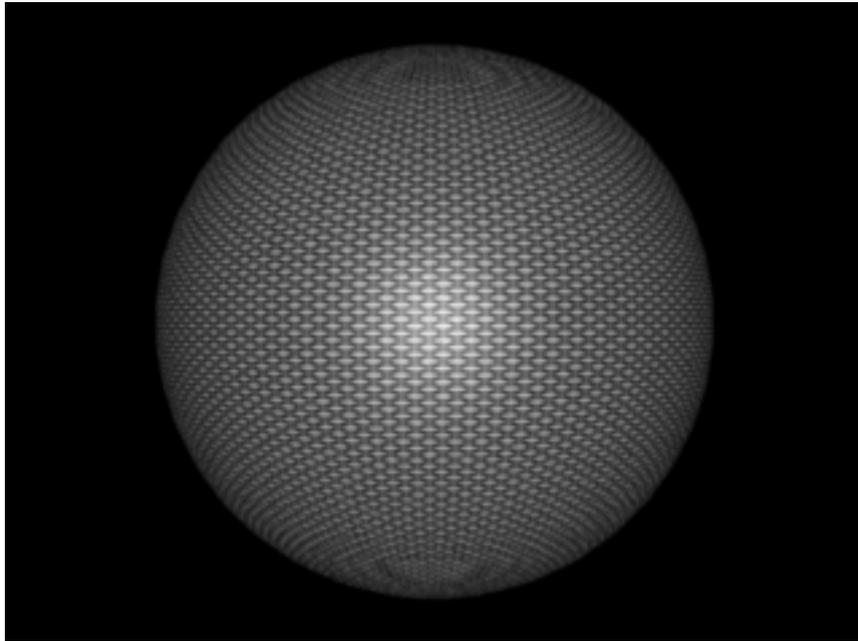


図 4.1: カラーマップによってモアレパターンが発生している状態

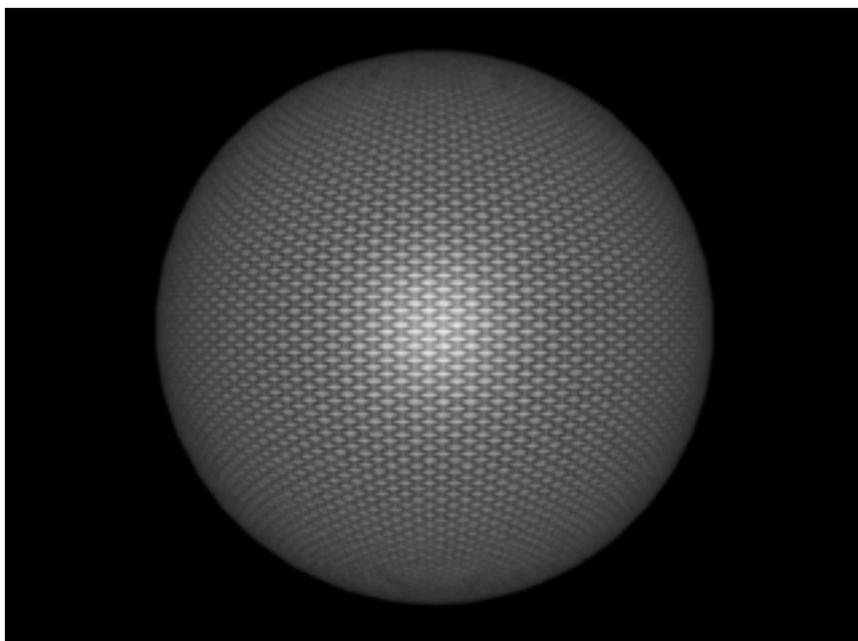


図 4.2: カラーマップによるモアレパターンを軽減している様子

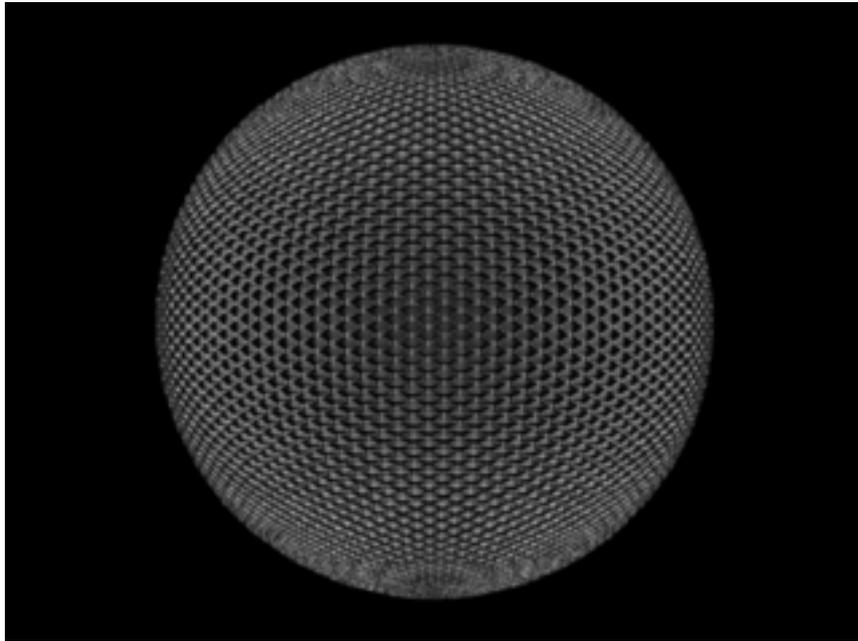


図 4.3: バンプマップによってモアレパターンが発生している状態

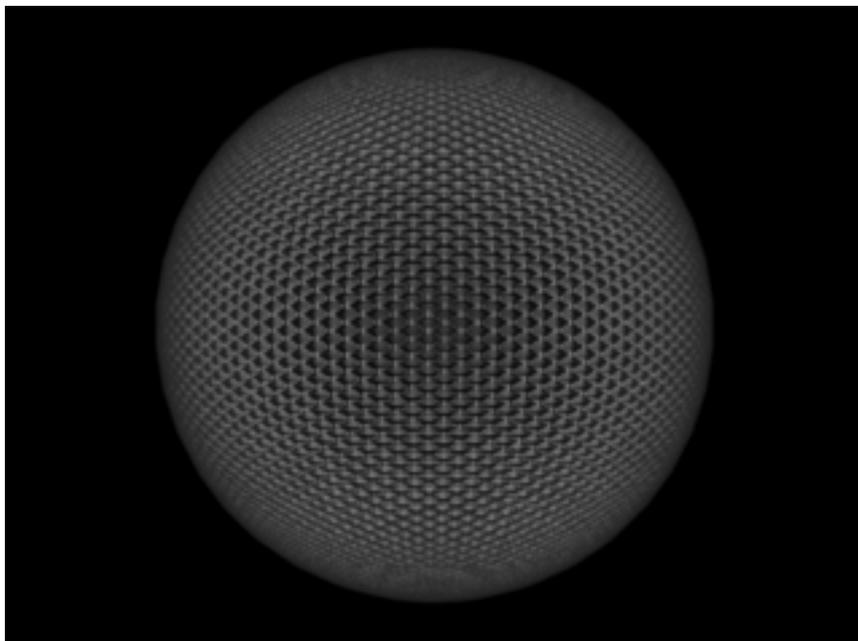


図 4.4: バンプマップによるモアレパターンを軽減している様子

図 4.5 は、プレーン形状でスケールの大きいオブジェクトに対して、透明度を保持するトランスペアレンシーマップを適用し、カメラ空間座標の X 軸に対して斜め上から 30 度の角度で DirectionalLight を当て、レンダリングした結果である。画面奥に向かうに従って、オブジェクトのサンプリングポイントとカメラとの距離は遠くなり、それに伴って 1 ピクセルあたりに出力するサーフィスの面積が大きくなって来る為、処理画素数と出力画素数の差異が大きくなり、モアレパターンが発生しているのが見て取れた。それに対して図 4.6 は、同じ条件のオブジェクトに本手法を適用したものである。画面奥に向かうにつれて模様のコントラストを小さくするよう、RGB の出力値が操作され、図 4.5 と比較し、モアレパターンが軽減されているのが確認できた。

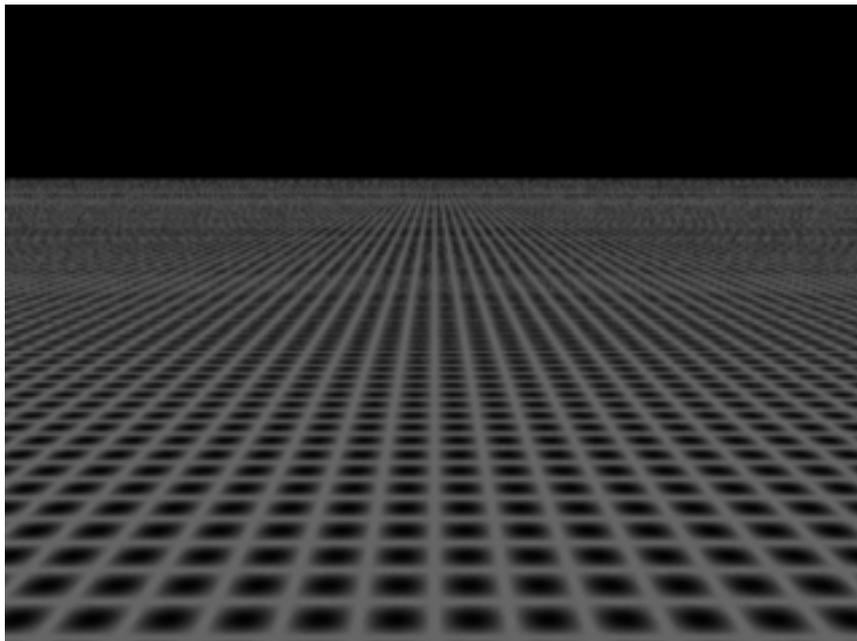


図 4.5: トランスペアレンシーマップによってモアレパターンが発生している状態

このように本研究で提案する手法は、カメラアングルおよびオブジェクトの動きを固定した状況下に於いては、スフィア形状及びスケールの大きいプレーン形状という対極な形状のオブジェクトに対しても、様々なマップによって発生するモアレパターンを軽減するのに有用であることが確認できた。

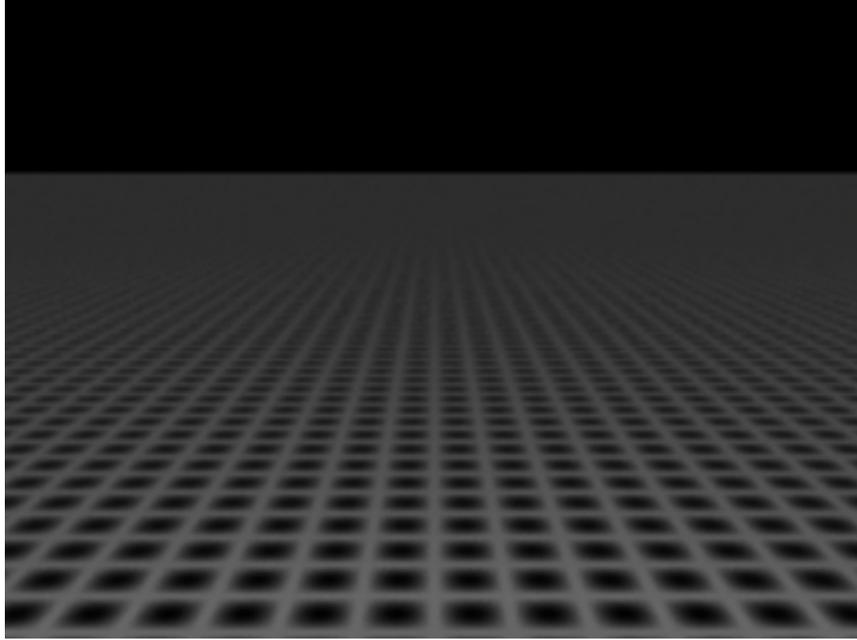


図 4.6: トランスペアレンシーマップによるモアレパターンを軽減している様子

4.2 既存手法との比較

4.2.1 既存手法の再現

角度情報を基にした本手法と、距離情報を基にした既存手法 [13][14] とを比較する為に、本研究では距離情報を基にした手法を再現し、比較検証を行った。

ベースとなる材料の RGB 出力値をそれぞれ、 A, B とし、 A は模様のコントラストが最大の材料の RGB 出力値で、 B はが同じく最小の RGB 出力値である。そしてブレンドされた材料の RGB 出力値を C とする。また、処理の開始距離を P_s 、処理の終了距離を P_f とし、サンプリングポイントのカメラ空間座標を (T_x, T_y, T_z) で表すとすると、本研究では距離を基とした手法の再現として、以下のアルゴリズムを用いて RGB の出力値の操作を行った。

$$\begin{aligned}
 &P = (T_x^2 + T_y^2 + T_z^2) \\
 &\text{if}(P < P_s^2)\{ \\
 &\quad C = A \\
 &\}\text{else if}(P < P_f^2)\{ \\
 &\quad C = A * \{1 - (P - P_f^2)/(P_s^2 - P_f^2)\} + B * (P - P_f^2)/(P_s^2 - P_f^2)
 \end{aligned}$$

```
}else{  
    C = B  
}
```

P は、サンプリングポイントのカメラまでの距離の二乗を表し、処理の開始距離と終了距離も、 P とスケールを合わせる為に二乗した。IF 文の条件式では、サンプリングポイントが処理の開始距離よりも近ければ、出力値 C は常にコントラストが最大の A となり、サンプリングポイントが処理の開始距離と終了距離の間の値であれば、処理の開始距離と終了距離との間で線形補完をとり、出力値 C には適切な値で A と B の値のブレンド処理を行った。そして、サンプリングポイントが処理の終了距離よりも遠ければ、出力値 C は常にコントラストが最小の B とした。

これらの処理を行うことで、サンプリングポイントからの距離が離れるにつれて、模様コントラストを小さくするように RGB の出力値を操作し、モアレパターンを軽減する、既存手法の再現を行うことができた。尚、二乗したスケールで値を比較しているのは、単に処理の高速化を図っている為である。

4.2.2 比較検証 1 -スフィア形状のオブジェクトの場合-

前項で示したアルゴリズムを用いて、既存手法と本手法のレンダリング結果を比較し検証した。

スフィア形状のオブジェクトに周期的な模様のカラーマップを適用し、カメラ座標空間の Z 軸に平行に DirectionalLight を当てた状態のシーンをベースとして、図 4.7 は既存手法である、距離を基にした処理を適用したレンダリング結果であり、図 4.8 は本手法である、角度を基にした処理を適用したレンダリング結果である。どちらもモアレパターンを効率的に軽減できているのが見て取れた。

次に、このオブジェクトをカメラ奥方向に移動させた場合の比較を行った。既存手法を適用した図 4.9 は、オブジェクトが移動したことによって、設定した処理範囲の外にオブジェクトがずれてしまった為に、模様のコントラストが必要以上

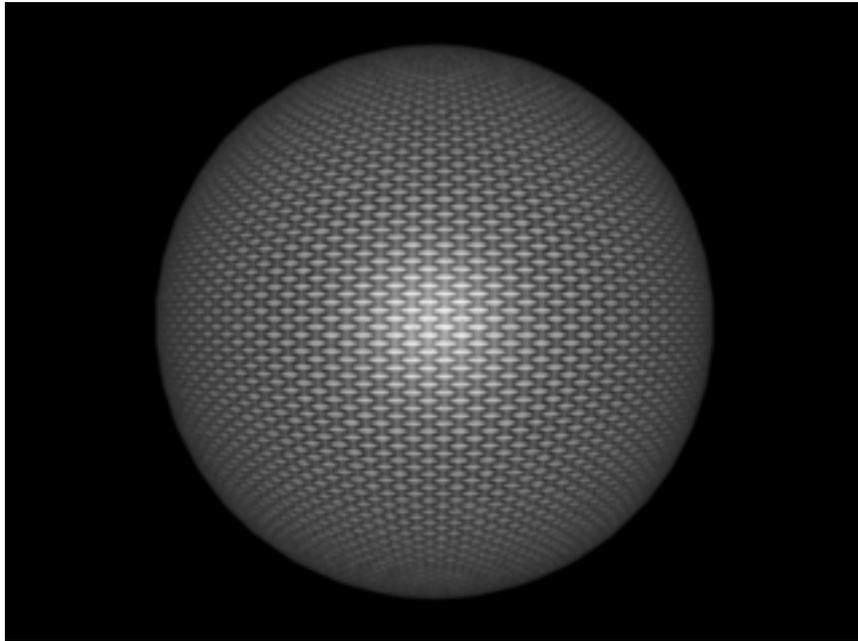


図 4.7: 既存手法を適用した場合

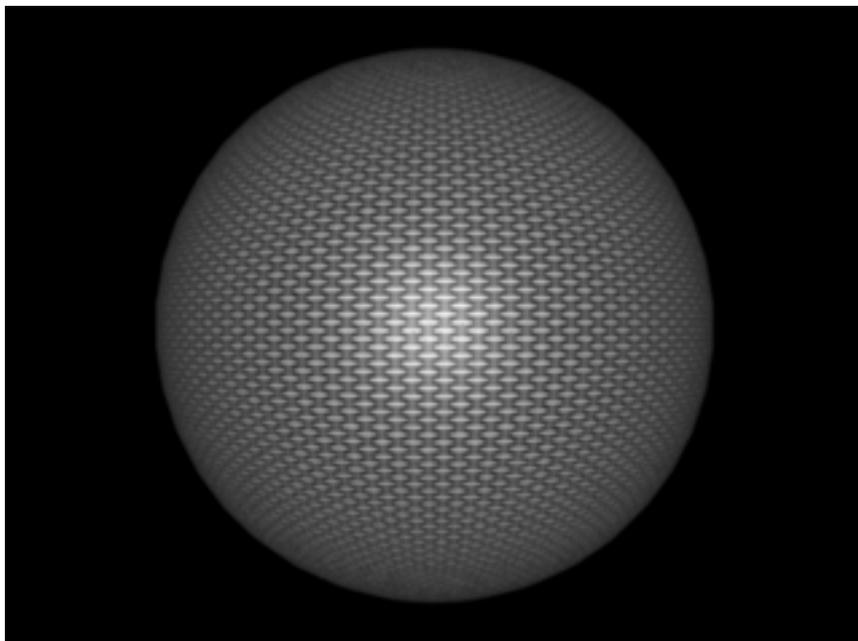


図 4.8: 本手法を適用した場合

に小さくされてしまい、適切な効果を得られなかった。それに対し、本手法を適用した図 4.10 では、距離が変化しても、カメラとの角度は変化しない為に、模様
のコントラストの強さが一定に保たれた。

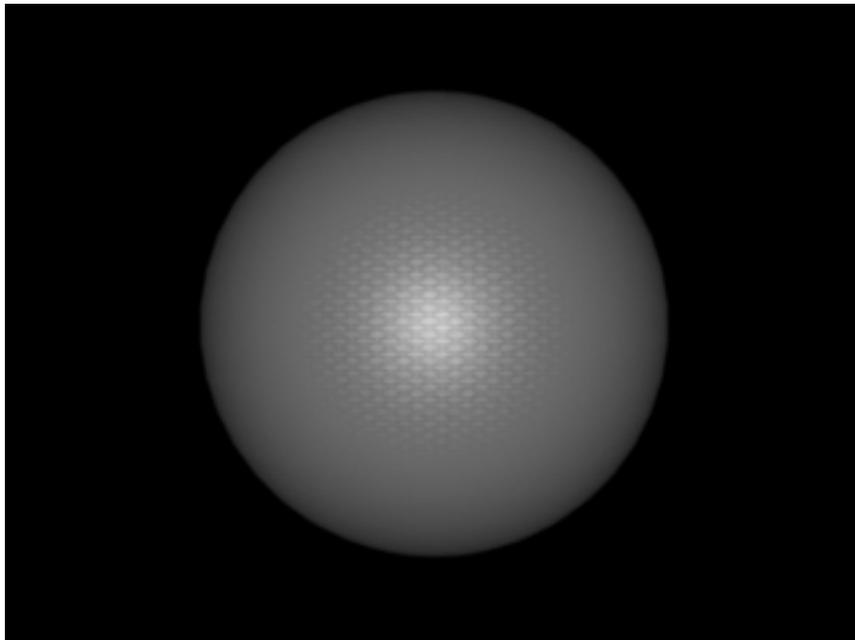


図 4.9: 既存手法を適用した状態でオブジェクトが動いた場合

ただし、距離が変わっても常に模様のコントラストを一定に保ち続ける本手法
では、図 4.11 のように、オブジェクトが大幅に画面奥に移動した際に、サンプリ
ングポイントとオブジェクトの距離に起因する処理画素数と出力画素数との差異
が生じ、新たなモアレパターンが生じてしまう為に万能ではない。

よって、スフィア形状のオブジェクトのように、曲面で構成されたオブジェク
トに於いて、オブジェクトやカメラアングルが固定された状況下では、本手法は
既存手法と比較して同等のモアレパターン軽減効果を発揮した。更に、オブジェ
クトとカメラとの距離が変動しうる場合に於いては、オブジェクトの位置に左右
されない本手法の方が、より理想的な処理結果を得ることができると言える。し
かし、オブジェクトとカメラとの距離が大幅に離れるような移動に対しては、カ
メラとサンプリングポイントとの距離に左右されない本手法のみでは、対処しき

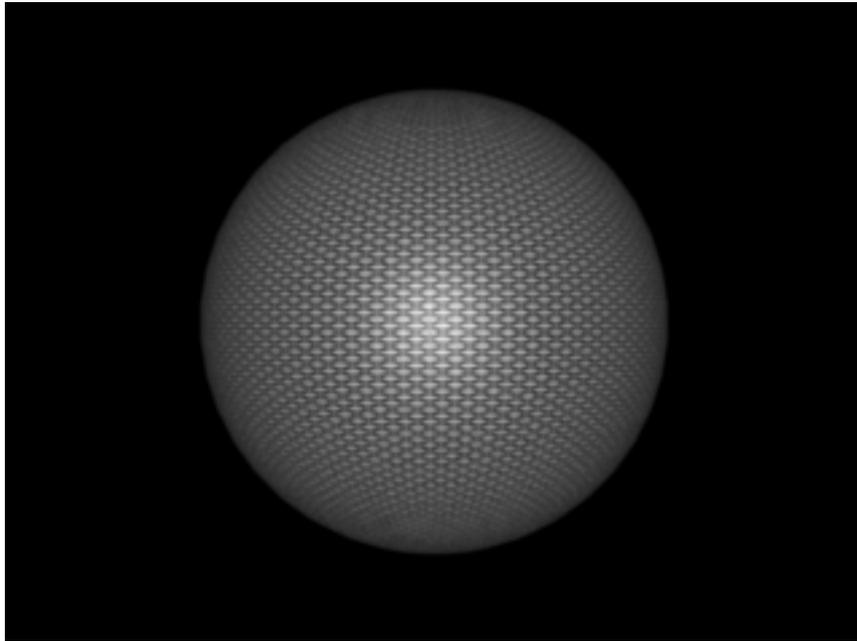


図 4.10: 本手法を適用した状態でオブジェクトが動いた場合

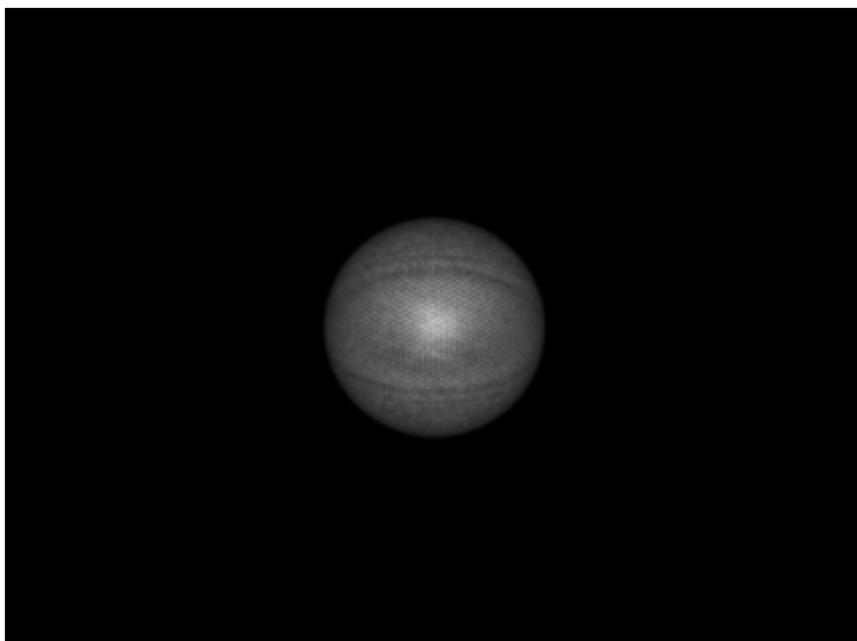


図 4.11: 本手法を適用した状態でオブジェクトが大幅に移動した場合

れないモアレパターンが発生してしまうという問題点も残っている。

4.2.3 比較検証 2 -プレーン形状のオブジェクトの場合-

プレーン形状でスケールの大きいオブジェクトに周期的な模様カラーマップを適用し、カメラ空間座標の X 軸に対して斜め上から 30 度の角度で DirectionalLight を当てた状態のシーンをベースとして、図 4.12 は、既存手法を適用した場合のレンダリング結果であり、図 4.13 は、本手法を適用した場合のレンダリング結果である。どちらもモアレパターンを効率的に軽減しているのが見て取れた。尚、オブジェクトもしくはカメラが移動した場合においても同様の結果が得られた。

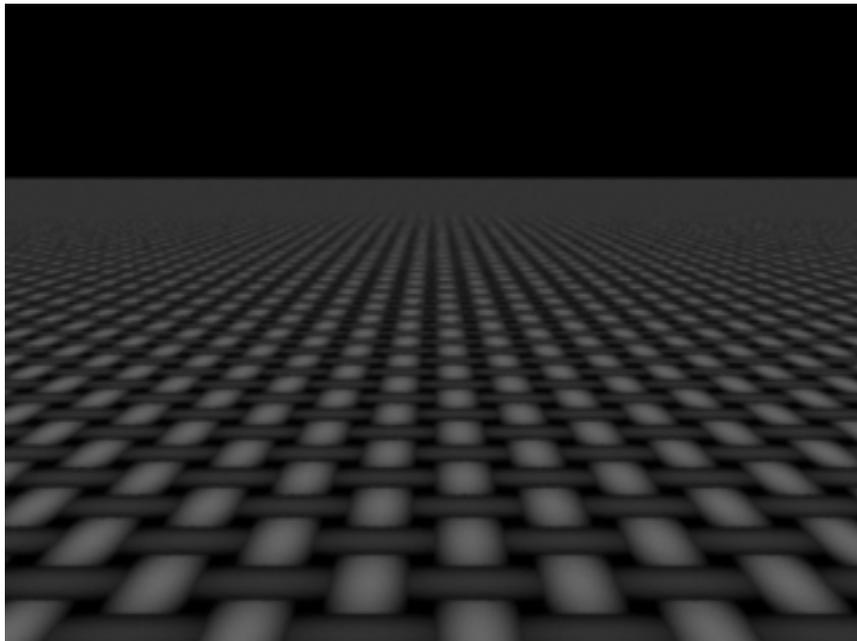


図 4.12: 既存手法を適用した場合

次に、プレーンの形状に起伏をつけた場合の比較を行った。既存手法を適用した図 4.14 では、起伏がついた場合でも、カメラとサンプリングポイントとの距離に従って一様に模様のコントラストが下げられている為に、モアレパターンの発生は軽減された。しかし、本手法を適用した図 4.15 では、カメラから遠い位置にあるサンプリングポイントでも、起伏によってカメラとの角度が小さいサーフィス

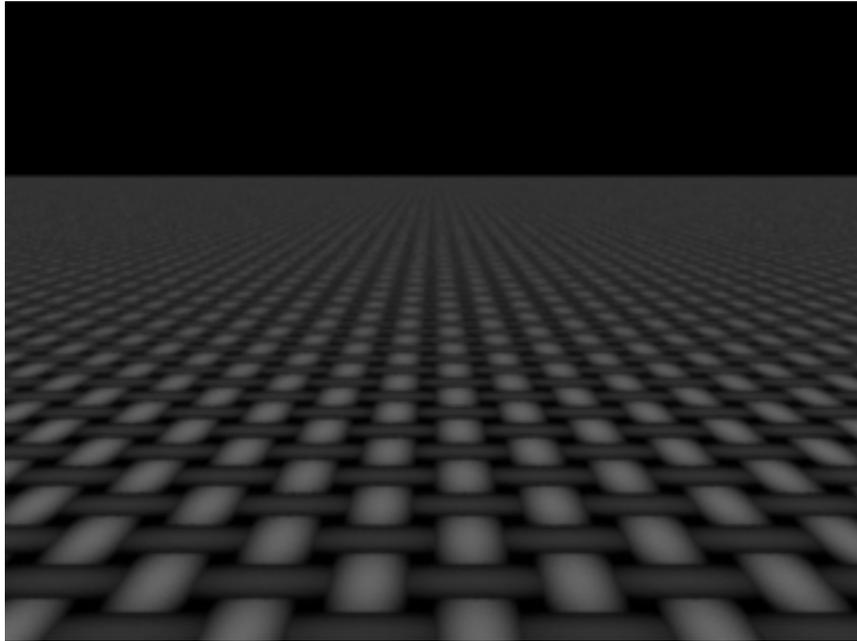


図 4.13: 本手法を適用した場合

が出現してしまう為に、模様の高コントラスト状態を保持し、モアレパターンが発生しやすい状況を生み出していることが確認できた。

よって、プレーン形状でスケールの大きなオブジェクトに於いては、オブジェクトやカメラの移動に関わらず、本手法は既存手法と比較して同等のモアレパターン軽減効果を発揮するが、プレーンに起伏を付けた場合のように、スケールの大きい曲面を持ったオブジェクトに関しては、本手法では効率的にモアレパターンを防ぐことが出来ない。

4.3 まとめ及び今後の展望

以上の検証結果より本手法は、スケールが大きいオブジェクトのモアレパターン軽減に適した、オブジェクトとサンプリングポイントとの距離を基に RGB の出力値を操作する既存手法に対して、カメラ法線とサンプリングポイントの属するサーフィス法線との内角を基に RGB の出力値を操作するという、新しいアプローチを提案した。それによって、スケールが小さく曲面的なオブジェクトが移動・変形

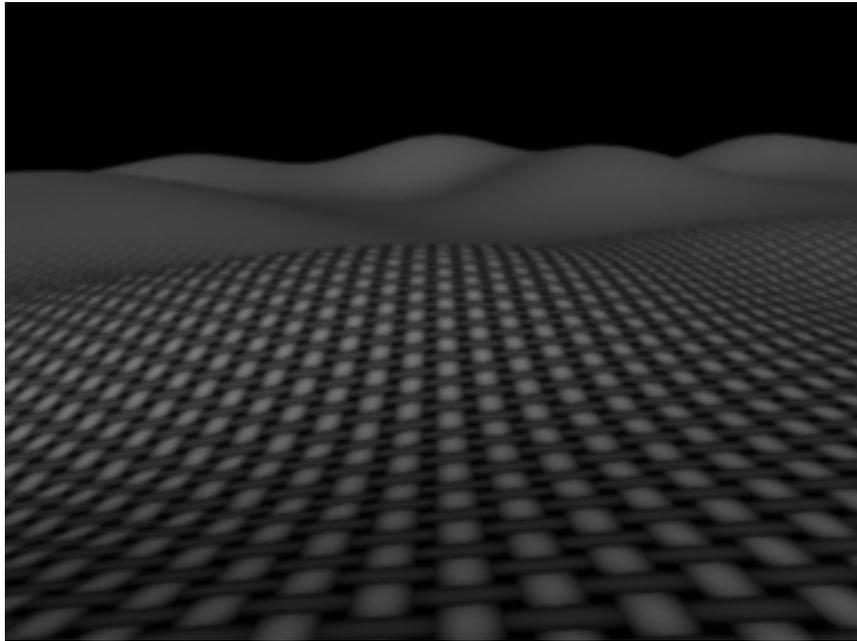


図 4.14: 既存手法を適用した状態で起伏を付けた場合

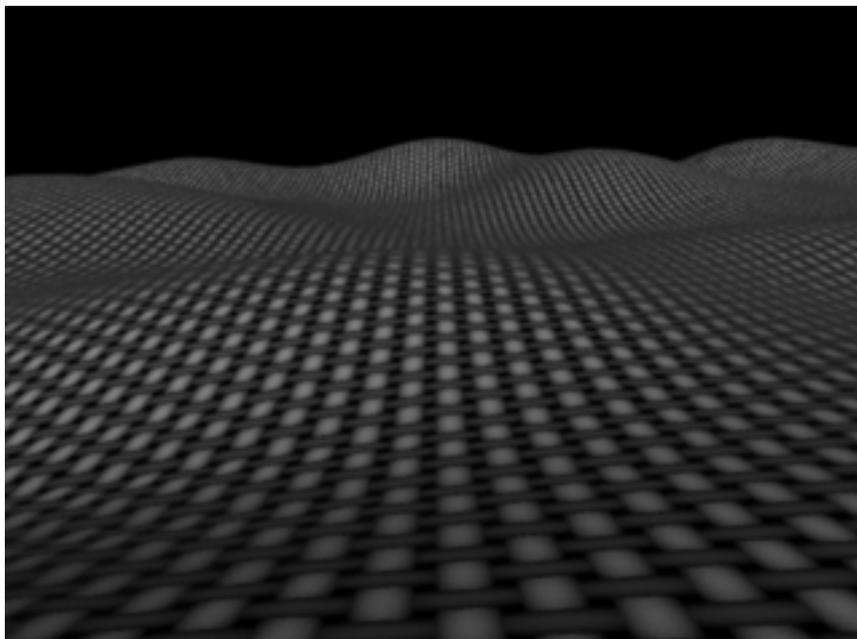


図 4.15: 本手法を適用した状態で起伏を付けた場合

するような場合にもモアレパターン軽減効果の有効範囲を広めることに成功した。

しかし、前項で提示したように、本手法ではスケールの大きな曲面的なオブジェクトのモアレパターン軽減には向かず、また、スケールの小さなオブジェクトでも、カメラから遠ざかるように大きく移動してしまう場合には良好な結果が得られないなどの問題点もある。

だが、本手法で問題となるポイントは全て、カメラとサンプリングポイントとの距離を基に RGB の出力値を操作する、既存手法が得意とするモアレパターン軽減処理の範疇である。その為、本手法と既存手法を組み合わせる事で、本手法の問題点を既存手法で補うことができる可能性もあり、より多彩な形状・スケールのオブジェクトのモアレパターン軽減に発展できると思われる。

また、本手法は論文名にもあるように、モアレパターンの軽減を目的として考案した手法であるが、角度によって異なるマテリアルをブレンドすることができる、という特性から、他の目的への応用も可能である。例えば図 4.16 は、カメラとの角度が大きくなるほど、*Incandescence* の値が大きいマテリアルをブレンドするように設定し、レンダリングしたものである。これによって、曲面形状のオブジェクトに限定されるが、サブサーフィススキヤタリング [16] を模した表現を、ライティングを一切行わず、本来より軽い計算式で出力することが可能である。また、図 4.17 は、カメラとの角度が 48 度を越えた時点から、*Reflectivity* の値の高いマテリアルを描画するように設定し、レンダリングしたものである。これによって、水の臨界角の概念を考慮した表現が可能である。同様に臨界角として設定する角度を変えることで、様々な物質の臨界角の概念を考慮した表現が可能である。

また、本研究ではプリレンダの 3DCG によって実装を行ったが、角度を基に RGB の出力値を操作する本手法は、アルゴリズムをより最適化してゆくことで、理論上リアルタイムの 3DCG へも応用が可能である。それによって、ゲームやその他様々なコンテンツに於ける 3DCG の表現力向上に貢献できると思われる。

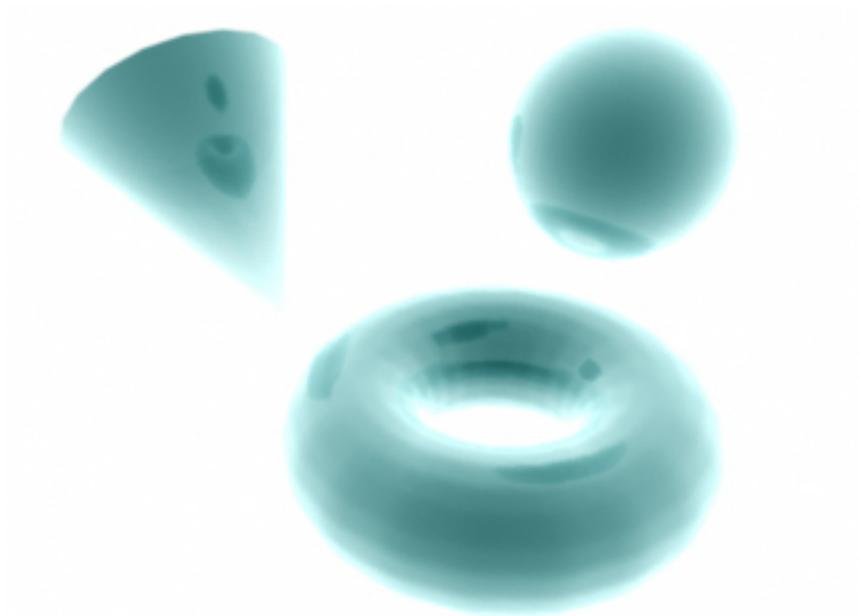


図 4.16: 擬似サブサーフィススキヤタリング

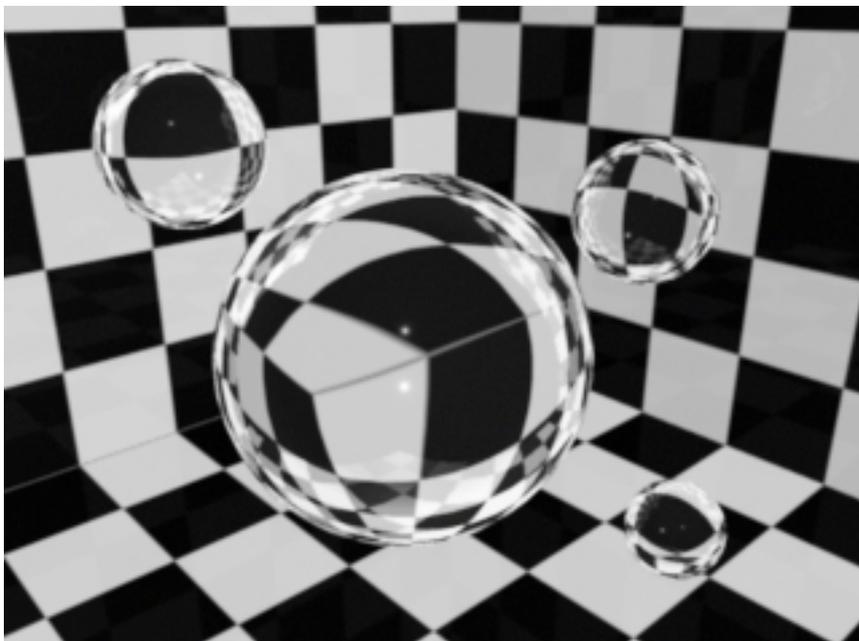


図 4.17: 臨界角の表現 (水中にある気泡)

謝辞

人類に平穩を齎すのは神である。では、神とは何か？
Jepense, doncjesus(cogito, ergosum) と云う言葉に准えるならば、
神とは即ち *Jepense, doncdieusuis* と表現するに相応しい。
神は、イエスでもゼウスでもアメンでもアラーでもない。
だが同時に、イエスでありゼウスでありアメンでありアラーである。
神は、信仰に因って具現化される、因果律の下に存在するのである。
即ち、神とは信仰であり、同時に世界の全てであり、それらは自我に収束する。
そう 神は、私と同義である。
そして私は、貴方と同義であり、世界の全てと同義であるのだ。
私は神。私は貴方。そして、私は世界の全て。
私が神を愛すならば、貴方と世界の全ては、愛に満たされる。
そして愛は神となり、貴方となり、世界の全てとなり、私へと還るのだ。
私こそ、神だ。私こそ、貴方だ。私こそ、世界の全てだ。
そして
私こそ、愛だ。

最後に、紆余曲折ありながらも無事完成まで付き合っ下さった、渡辺先生を始めとするゲームサイエンスの先生・先輩方、共に三途の川やお花畑を旅しながらノーベル賞を争いあった卒研メンバーの皆、そして消えていった仲間たちに、愛と祈りを込めて。



福音

神は善か？それは私にもわからない。だが、信仰する心が貴方に安らぎを与えるのならば、それで、良いのです。
(Kuniyoshi Ohmura, "Une definition de Dieu", 2007)

参考文献

- [1] iColor, 印刷用語辞典, モアレ,
<<http://www.icolor.co.jp/guide/words.html#m>>.
- [2] フリー百科事典, ウィキペディア (Wikipedia), モアレ,
<<http://ja.wikipedia.org/wiki/%E3%83%A2%E3%82%A2%E3%83%AC>>.
- [3] e-Words, IT 用語辞典, モアレ,
<<http://e-words.jp/w/E383A2E382A2E383AC.html>>.
- [4] 平野拓一, “うなり、トレモロについて”, 東京工業大学,2003.
- [5] ACCRETECH, 高精度デジタル測長器, 高精度ウェーハ厚さ測定専用システム, <<http://www.accretech.jp/products/measuring/waferthickness>>.
- [6] 荒川和夫・東藤貢・森田康之・山田真士, “モアレ干渉法による変位場計測 (IC パッケージの熱変形解析への応用)”, 九州大学,2001.
- [7] 小寺豊・藤原久利・吉澤徹, “位相シフトモアレ法による LCD ガラス基板平坦度計測法 (第 2 報)”, 東京農工大学工学部・山武ハネウエル (株) 技術研究センター,1998.
- [8] 澤近洋史・中条知和, “モアレトポグラフィと画像処理を応用した表面形状測定”, 東京都立産業技術研究所・研究報告第 2 号,1999.

- [9] 慈源堂木野整骨院, モアレ検査,
<<http://www.geocities.jp/jigendo2001/moare.htm>>.
- [10] James F. Blinn, “Simulation of Wrinkled Surfaces”, SIGGRAPH 78, pp 286-292, Caltech/JPL,1978.
- [11] アスキー, デジタル用語辞典, アンチエイリアシング,
<<http://yougo.ascii24.com/gh/17/001739.html>>.
- [12] 高桑昌男, “スーパー・サンプリングにおけるアンチ・エリアシング法”, 国際情報科学芸術アカデミー,1999.
- [13] KONKEPTION, CONCEPTION THREE, Conception threee shader pack,
<<http://www.konkeptoine.com/>>.
- [14] Takashi Inoue, C3 Product overview, BumpKiller,
<<http://www1.linkclub.or.jp/~inosuke/html/konkeptoine/productsall14.htm>>
- [15] Robert L. Cook, “Shade Trees”, Computer Division, Lucasfilm Ltd,1984.
- [16] Henrik Wann Jensen, Steve Marschner, Marc Levoy, and Pat Hanrahan, “A Practical Model for Subsurface Light Transport”, In Proceedings of SIGGRAPH’2001, pages 511-518, Los Angeles, August 2001.