

視線と光線を考慮した手描き風 3D アニメーションのための ハイライト表現と制御の手法

田中遼平^{†1} 森本有紀^{†1} 高橋時市郎^{†2}

概要：この論文では、3DCG によるアニメーションにおいて、手書きアニメで描かれるデフォルメされたハイライトを表現する手法を提案する。本手法では任意形状のハイライトを適用するために、3D モデルの法線の変更を提案する。また、ユーザが意図した表現を直感的に実現するための UI を提案する。

キーワード：3DCG, システム, アニメーション, セルシェーディング

1. はじめに

セルシェーディングは、3D モデルを使用したマンガ調アニメーションを手描きアニメ風に変換する方法として広く使用されている。しかし、3D モデルに対する光の反射によるハイライトの物理的挙動は、デフォルメされ物理的に正しくないアニメ風のハイライトを表現するには、適していない。アニメ風のハイライトを表現するためには、テクスチャマッピングを行う方法があるが、その形状と挙動は固定されてしまう。我々は、アニメ風のハイライトをレンダリングし、制御する手法を提案する。提案手法は環境マッピング[4]を拡張し、ユーザの好みに従ってハイライトの位置を調整し、視線と光線の角度を考慮した補間曲線によって制御することにより、ハイライトの位置と形状を変更する。これによって、ユーザの意図に従ったハイライトの挙動を制御し、アニメ風のハイライト表現が可能である。もう一つの提案手法の特徴は、3D モデルの法線をモデル本来のものとは別に作成する方法であり、これはモデル本来の法線の値によってハイライト形状が散らないように、法線を単純化する。ユーザは手法を使用して作成した法線を使用するか、プリミティブに近い単純な形状の法線をハイライトが適用される部分に転写することによって法線を単純化したモデルを使用する。以下、提案手法の法線を使用する前提として手法を示す。図 1 に提案手法の結果を表す。

2. 関連研究

3D モデルに任意のハイライトを生成する方法はこれまでにいくつか提案されている[1][2][3]。Sloan et al. [4]は、リットスフィアシェーディングを提案した。これはスクリーン空間の法線に対応する二次元球体上の陰影表現をモデルにマップする方法である。Todo et al. [5]は、Blinn-Phong モデルをリットスフィアに適用し、ライティングやシェーディング等を行うことで、ハイライトや陰影を自動計算し、かつインタラクティブに使用できるように拡張した。これらの手法は、レンダリングの計算量を大幅に減らし、かつ

幅広い非写実的表現を容易に達成できる。しかし、いずれの方法でも、ハイライトの形状はモデルの形状によって著しく変形するので制御することはできない。

環境マッピングは Blinn et al. [6]が提案した周囲環境の画像（環境マップ）を用いて簡易的にモデルの金属反射等を表現する事ができる方法である。また、Ritschel et al. [7]は、視線の反射ベクトルをインタラクティブに操作できる手法とインタフェースを提案した。

本手法では、環境マッピングとハイライトテクスチャの移動・変形を組み合わせることによって、アニメ風ハイライトの表現を行う。

3. 提案手法

提案手法の全体の概要を説明する(図 2)。最初に、ユーザが、2D テクスチャとしてハイライトの形状を指定することによって、ハイライトの形状を設計する。次に、以下の過程を繰り返す。

- I. ユーザが指定したハイライトの位置に対して、そのシーンの光線ベクトルと視線ベクトルの角度をキーとして使用し、提案手法によって指定した制御点の位置を更新する(図 2a)。
- II. I の結果を用いて、キューブ環境マップを作成する(図 2b,c)。
- III. II の結果を用いて 3D モデルに変形したハイライトを環境マッピングする。このとき、擬似法線(図 2d)を用いて法線の単純化を行う。

提案手法は、動的な環境マッピングに基づいている。また、これらの過程はシンプルなテクスチャ変形とマッピングによってリアルタイムに実行することができる。ユーザは滑らかな球の法線である擬似法線を用いて、モデル本来の散らばった法線を単純化し、ハイライトを表現することができる。ユーザは擬似法線の中心 O を設定する。これはモデルのサーフェスを近似するように、モデルの中心付近

^{†1} 東京電機大学
Tokyo Denki University

^{†2} 東京電機大学 / UEI Research
Tokyo Denki University / UEI Research



図 1 本手法を適用したレンダリング結果と対応するハイライトのテクスチャの例



(a) ハイライトの制御 (b) 球面マッピング (c) 環境マップ (d) 擬似法線 (e) 結果

図 2 本手法のフロー図

に設定する。生成されたハイライトはセルシェーディングによって生成された拡散反射成分と組み合わされてレンダリングされる。

提案手法は、動的な環境マッピングに基づいている。また、これらの過程はシンプルなテクスチャ変形とマッピングによってリアルタイムに実行することができる。ユーザは滑らかな球の法線である擬似法線を用いて、モデル本来の散らばった法線を単純化し、ハイライトを表現することができる。ユーザは擬似法線の中心 O を設定する。これはモデルのサーフェスを近似するように、モデルの中心付近に設定する。生成されたハイライトはセルシェーディングによって生成された拡散反射成分と組み合わされてレンダリングされる。

3.1 視線ベクトルと光線ベクトルを考慮したハイライトの制御

視線ベクトルと光線ベクトルの角度によって、ハイライトの位置を制御する。ハイライトの形状は任意の 2D テクスチャによって表現される。このテクスチャを背景テクスチャ上で移動することによってユーザの意図するハイライトの位置を、視線ベクトルと光線ベクトルの角度によって補間する(図 1a, b, c)。

まず、視線ベクトル、光線ベクトルの角度の制御を回転行列によって行うものとする、視線ベクトル、光線ベクトルは共に Z 軸+方向を基準とし、X 軸回転、Y 軸回転の順に計算を行う。視線ベクトルに対する光線ベクトルの X 軸の相対的な回転量と Y 軸の相対的な回転量を取得する。ユーザは視線ベクトル・光線ベクトルの値・ハイライトのテクスチャの位置を、結果を表示させながら手動で意図する位置に調整し、ハイライトのテクスチャの X 座標に対して Y 軸の相対的な回転量、Y 座標に対して X 軸の相対的な回転量をそれぞれキーとして使用し、X 座標、Y 座標のそれぞれで補間して毎フレームのテクスチャの位置を決定する。ハイライトのテクスチャが背景のテクスチャより左右には

み出した場合、球面マッピング上の連結と同じく、図 1 のように、反対側の境界に連続して描画する。上下方向に関しては、頂点の移動領域を閾値処理で制限することができる。

3.2 擬似法線に基づいたハイライトの簡易化

ハイライトの簡略化は元の法線を平滑化する擬似法線に基づいている(図 1e)。擬似法線 N^* はモデル表面の位置 p を用いて $N^* = p - O$ とし、更に正規化したものとして定義する。環境マッピングの際、擬似法線とモデルの法線の割合 γ : $(1-\gamma)$ でブレンドした値を用いる。ここで、 γ は 0-1 の範囲内でユーザが指定する定数である。この手法により、意図したハイライト形状とモデルの法線を考慮した複雑なハイライト形状の間で、幅広い表現を選択することが可能である。

3.3 インタラクティブなハイライトの制御

本システムでは、ユーザがハイライトの結果を確認しながら制御するためのインタフェースを実装する。ユーザは 3.1, 3.2 を適用した結果を見ながら、ハイライトのテクスチャの移動、視線・光線の値の変更と、キーの追加を常に行うことが出来る。これによりユーザは挙動が補間されたハイライトが意図した表現であるかどうかを常に確認することが可能であり、必要に応じてキーの内容を変更・追加・消去することも可能である。

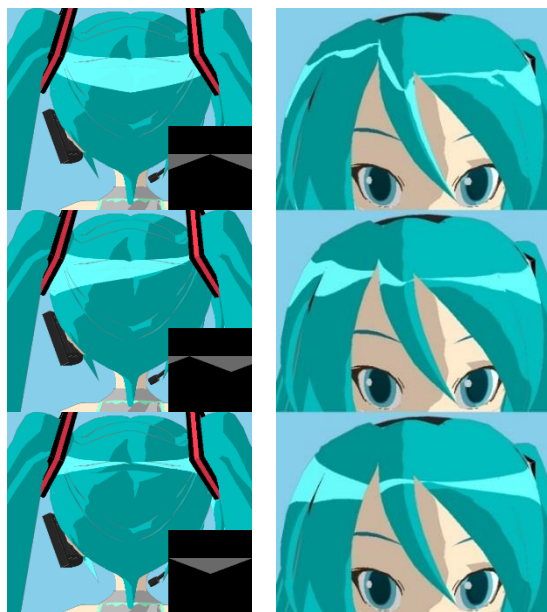


図 4 補間による結果の比較(左)と擬似法線のパラメータによる結果の比較 (右). 左列上より補間なし, 任意のパターン 1 とパターン 2, 右列上より, $\gamma=0.0,0.9,1.0$.

4. 結果

提案手法の開発・実験には Unity5, PC は Intel(R) Core(TM) i7-3770, NVIDIA Quadro 600, メモリ 16GB, Windows 10 Professional 64bit を用いた. 擬似法線は, 描画されたピクセル毎に求め, 正規化したものを用いた. 図 1 は, 任意にテクスチャの位置を設定した結果画像である. 結果画像からは視点の違いによってハイライトの位置を任意に調整できることが確認できる. 図 4 左列では, 任意の値に調整することによってハイライトの位置を決められることがわかる. 図 4 右列では, 提案した擬似法線によりモデルの凹凸によるハイライトの局所的な変形を防ぎ, 意図した形状を保っていることと, 既存のモデルの法線と擬似法線を組み合わせることによって, 幅広い表現を生成できることがわかる. また, ハイライトにテクスチャを使用することによって任意の表現が可能である(図 5).



図 5 様々なハイライトの表現

5. むすび

本研究では視線・光線をキーとして, そして法線や新た

に設定した擬似法線を用いて, ユーザの意図した形状のハイライトを表現するためにパラメータ制御できる手法を提案した. 本手法によって, 3DCG におけるリアルタイムなアニメ風ハイライトの表現が可能である.

今後の課題として, 頭部以外への手法の応用が挙げられる. 本手法では球以外で近似すべき形状のハイライト表現に適していない為, 他の物体による近似を検討し, 対応できるハイライトの形状の幅を広げる必要がある.

また, ハイライトの位置・形状をユーザが直感的に設定できるインターフェースの開発を行い, 更に幅広い表現を目指す.

謝辞 本研究にあたり, 主に初期の段階で適切な助言を下された東京大学の藤堂英樹博士に感謝の意を表します.

参考文献

- 1) Choi, J. J. and Lee, H. J.: Rendering stylized highlights using projective textures. The Visual Computer 22(9-11), pp.805-813, Springer-Verlag, New York (2006).
- 2) Anjyo, K. et al.: Tweakable light and shade for cartoon animation, In Proceedings of the 4th international symposium on Non-photorealistic animation and rendering (NPAR '06). ACM New York, pp.133-139 (2006).
- 3) Sakai, T. and Savchenko, V.: Skeleton-based cartoon hair modeling using blobby model. In SIGGRAPH Asia 2013 Posters, ACM New York, 2013.
- 4) Sloan, P. P. et al.: The Lit Sphere: A Model for Capturing NPR Shading from Art. Graphics Interface, pp.143-150, Canadian Information Processing Society Toronto (2001).
- 5) Todo, H. et al.: Lit-Sphere extension for artistic rendering. The Visual Computer 29(6-8), pp.473-480, Springer-Verlag New York (2013).
- 6) Blinn J. and Newell, M.: Texture and reflection in computer generated images. Commun, ACM 19, pp.542-547, ACM New York (1976).
- 7) Ritschel, T. et al.: Interactive reflection editing. In SIGGRAPH Asia 2009 papers, ACM, New York (2009).