

MPEG等動画圧縮を考慮した Global Illumination(大域照明)モデルの開発

1. 背景

コンピューターグラフィックスが誕生して以来、長い間、三次元映像処理の研究がなされてきました。その中でも Global Illumination(大域照明)は写実的な映像を作るための手法として、映画や広告、ゲームのオフラインムービー等多様な分野で使用されてきました。それ以来20年経過しコンピューターの速度が早くなっているにもかかわらず、式も同様に複雑化しているので、計算時間が減る事は余りありませんでした。

アニメーションは連続した静止画の集合によって生成可能となり、それに伴う計算時間はフレーム数に比例して莫大な物となりました。これまで、時間軸を考慮して計算時間や質を向上する研究はありましたが、速度向上は限られた物でした。

2. 目的

本プロジェクトでは、従来までに開発されていた高速の大域照明法を動画圧縮に最適化しつつ組み込む事によって、従来まで別処理として分けられていたすべての工程を単一のフレームワークに収める事です。今回、私は通常ならば画像生成の後に行われるポストプロセスという工程に注目を置きました。ポストプロセスでは画像生成で作られた連続静止画を動画ファイルに落とし込む作業が行われます。この際、多くの圧縮がかけられせっかく長時間をかけて生成した画像の質が落ちるという事がありました(図1参照)。そこで我々はあらかじめ動画圧縮でどの程度の情報が失われるかを予測しておき、それを基に高速に画像を描画しようという方法を提案する事にしました(図2参照)。

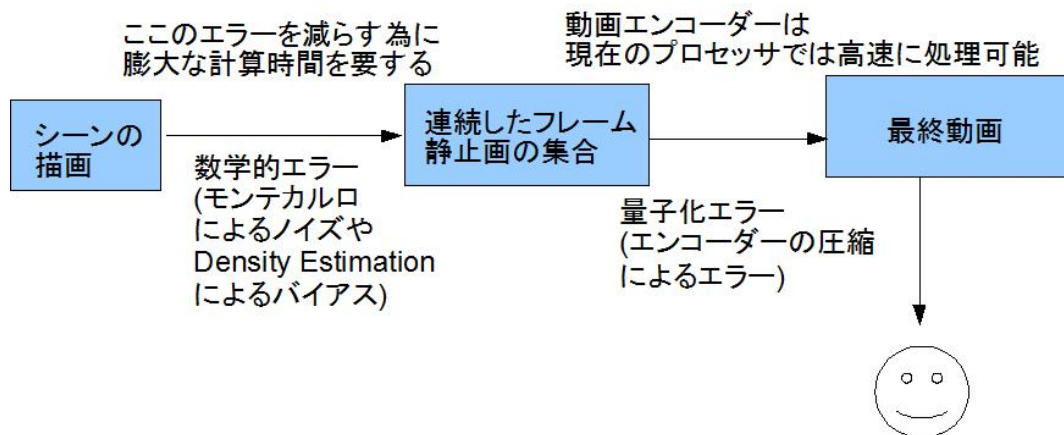


図1: 従来までのフレームワーク

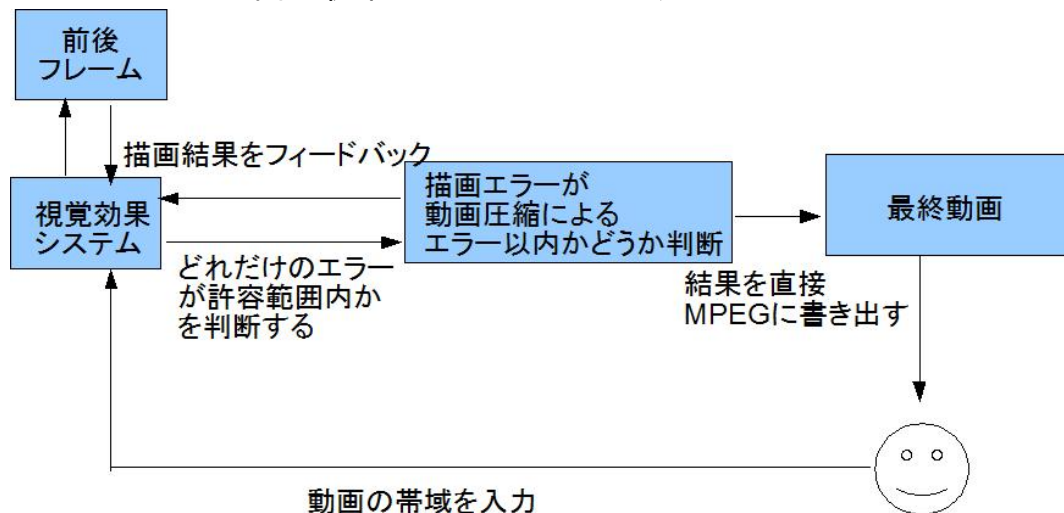


図2: 提案する新しいフレームワーク

3. 開発の内容

動作環境:

- Windows 2000, XP, Vista
- 2G 以上のメモリ推奨
- Pixel shader 3.0 以上をサポートするGPU が搭載

今回、私が開発したのは、MPEG2 のエンコーダーのルーチンを従来のGlobal Illuminationのレンダリングアルゴリズムに組み込むという物です。レンダリングアルゴリズムにはSiggraph 2006、2007 でコーネル大学から発表された「Light Cut」を動画圧縮に向けてカスタマイズしました。

今回開発したソフトウェアは大域照明のレンダリング部分と動画圧縮ルーチンを持っています。まず、画像がどれくらいのエラーを許容出来るかのエラーマップを計算します(図3)。その許容エラー量を元に貴重な光源を順番に評価していき、エラー許容範囲に入ったら画像を生成します。このプロセスは従来までの大域照明の複雑な工程を必要とせず、すべて自動で行われ、最終の動画が生成されます(図4)。



図4: 生成された動画(ウェブページで見る事が出来ます)

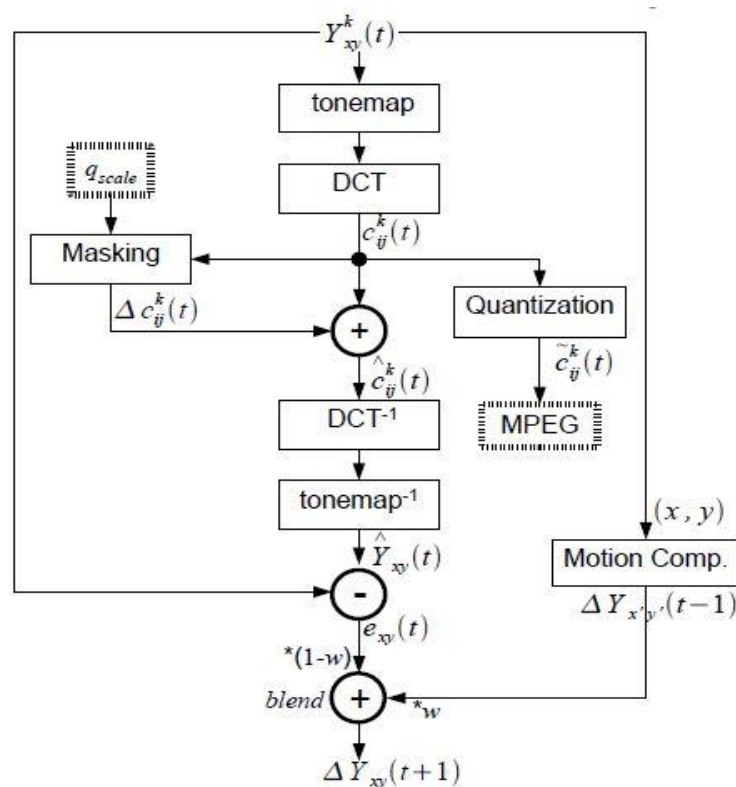


図3: MPEGの機能からエラーマップを導出する

4. 従来の技術(または機能)との相違

今までの静止画だけで判断する方法では、1%以内の輝度値の違いは人間には認識出来ないという事でそれ以下のエラーを打ち切る事が可能でした。しかし、今回はMPEGの圧縮工程まで含めてエラーを計算するのでより多くのエラーを受け入れられるという事が可能となり、それにより計算時間も大幅に向上します。光源の評価数は役半分から五分の一ぐらいにまで減らす事が可能となります。

した(図5)。

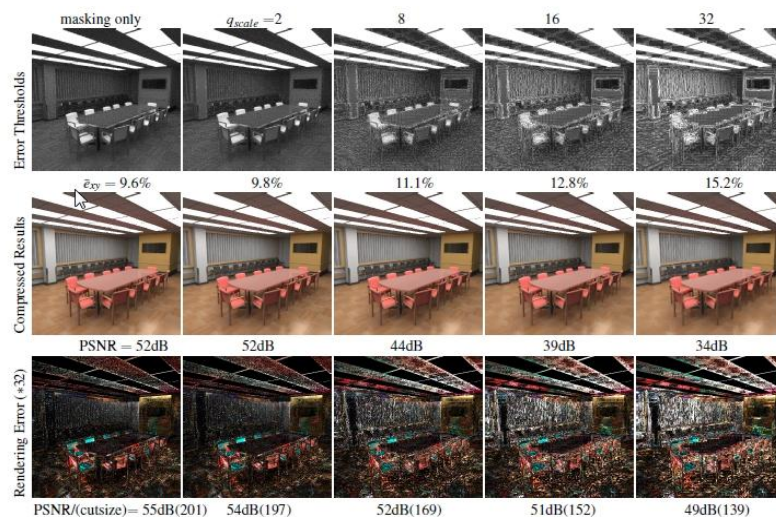


図5:MPEGの機能からエラーマップを導出する

5. 期待される効果

映画業界では Pixar 等が既に映画 Cars で時間軸に対する画像のフィルタリングを行っております。動画CGを扱う多くの業界で、計算速度を向上の為に我々の提案したエラーメトリックスが使用される事が期待されます。

また、モデリングソフトさえ持っていれば何十万もするレンダラーを購入する必要もなく、このソフトウェアを使えばレンダリングが可能となります。

6. 普及(または活用)の見通し

作成しソフトウェアの技術部分はすべて英語のドキュメントとして公開されます。10ページに及ぶ論文に加えてメインルーチンの部分はホームページにてソースコードとして公開します。また今後、英語のテクニカルレポートも作成する予定です。学会誌にも投稿していますのでCGを扱う世界中の人がこの手法に興味を持ってそして利用する事が出来ます。

動画CGの生成は国内産業だけではなく、ハリウッドから個人までの多くの分野

で使用されています。今回提案したソフトウェア、また手法はそれぞれの動画圧縮ファイルに組み込まれ多くの計算時間が省略する事が可能となるでしょう。

7. 開発者名(所属)

独ザーランド大学修士課程二年

マックスプランク工科研究所 コンピューターグラフィックスチーム

絹脇 伸一

(参考)<http://www.mpi-inf.mpg.de/resources/anim/EG08/>