

Height Fieldで表現された水面の屈折現象の高速レンダリング手法

5 N-7

徳山 哲朗

北海道大学大学院工学研究科

山本 強 高井 昌彰

北海道大学大型計算機センター

1 はじめに

本研究は、波の破碎を含む水流の挙動をモデル化し、並列機などの高価な機械を用いずに一般のPC上でリアルタイムに三次元CGによってレンダリングを行う事を目的としている。

前回までは、二次元においてHeightField上の速度差に着目して、破碎を行う手法を提案した。今回は、三次元化をする際に可視化のために必要となる、HeightFieldモデル上の水面のレンダリング手法について述べる。

2 水面による屈折現象

現実の水面では屈折現象により水面下の地面が歪んで見える。また、水面に入射する光も屈折されるため水面下の地面に複雑な光の模様が発生する。これは、光の屈折により、水面下の地面に届く光の分布に粗密が生じるためである。それゆえ、光が多く集まる部分は明るくなり、光が粗になる部分は暗くなる。

水面下の地面に現れるこのような光の模様を正確にレンダリングするためには、地面上の一点に対して、水面全体からの入射光を考慮する必要があり、非常に計算コストが高くなってしまふ。

そこで、本論文では次のような手法を利用する。まず、地面に発生する光の模様をアルファブレンディングを用いてレンダリングする。次に、得られた光の模様によってテクスチャに陰影づけをおこなう。そして、このテクスチャを水面に張り付けることにより屈折の表現を行い、水面そのもののレンダリングを行う。

3 水面下の光の模様の生成

それでは、水面下の地面に発生する光の模様のレンダリング手法を説明する。まず、HeightFieldモデルか

ら水面を構成するポリゴンを作成する。次に、そのポリゴンの各頂点に、光源からの光を入射させ、光の屈折方向を求める。屈折した方向にHeightFieldを探索して行き、屈折した光と地面との交点を算出する。そして、交点からなるポリゴンを生成する。交点では地面の法線を求め、輝度を次式によって決定する。

$$(\overrightarrow{L_{refracted}} \cdot \overrightarrow{N_{ground}}) \left\{ \frac{S}{S'} (\overrightarrow{L} \cdot \overrightarrow{N_{polygon}}) \right\}$$

ここで、 $\overrightarrow{L_{refracted}}$ は水面で屈折された光線ベクトル、 $\overrightarrow{N_{ground}}$ は地面と屈折した光線ベクトルとの交点における地面の法線ベクトル、 S は水面のポリゴンの面積、 S' は地面に生成されたポリゴンの面積、 \overrightarrow{L} は水面のポリゴンに入り込む光線ベクトル、 $\overrightarrow{N_{polygon}}$ は水面のポリゴンの法線である。

式の前半は、地面に届く光の拡散反射による光の強度を表しており、後半部分は、地面上のポリゴン全体に届く光の強さを表している。

レンダリングを行う際には、ポリゴンの色は白とし、先ほど交点で算出した輝度値をアルファ値としてポリゴン内で補間しながらアルファブレンディングを行い、輝度値をフレームバッファに加算して行く。ブレンディングに用いた式は以下の通りである。ただし、 α はアルファ値、 C_{white} は白色を、 $C_{framebuffer}$ はフレームバッファ上の色を表している。

$$C_{framebuffer} = \alpha \times C_{white} + C_{framebuffer}$$

このようにすることで、近似的に光の粗密を実現することができる。ここで生成した画像によって、地面のテクスチャに陰影を付け、レンダリングの際に利用する。Fig.2にこの手法によって生成した画像を示す。

4 水面のレンダリング

水面のレンダリングの際にも、水面下の光の模様の生成と同様な手法を利用する。ただし、入射するのは光源からの光ではなく、視点からのレイを入射する事になる。

A rapid rendering method for the light refraction on the water surface in the height field model.

Teturo Tokuyama, Tsuyoshi Yamamoto, Yoshiaki Takai
Hokkaido University

水面に入射したレイを屈折させ地面との交点を求める。先ほどは交点で輝度値を算出したが、ここでは交点の座標をテクスチャ座標に変換する。

水面をレンダリングする際には、ここで求めたテクスチャ座標を利用して、水面下の光の模様生成の際に作成したテクスチャをポリゴンに張り付ける事によってレンダリングを行う。

また、水面では反射も起こるので、周囲の景色をテクスチャで用意し、環境マッピングを施す事で水面への景色の写り込みを実現することができる。Fig.3に屈折の様子を示す。Fig.3には空の画像の環境マッピングも行っている。

- [3] 徳山哲朗、山本強、高井昌彰 「水流のしぶき発生に関する考察」 55回情報処理学会 全国大会講演論文集（分冊4），pp.4-309



Fig.1 使用した地面用のテクスチャ

5 実験結果

本論文で提案した手法を使って、レンダリング速度の計測を行った。実験環境は、IBM-PC/AT 互換機で、Pentium 166 MHz、メモリ64 MB、セカンドキャッシュ256 KB。3DアクセラレータはVoodoo2チップを搭載したもので、ライブラリーはGlideを使用した。表示サイズは256×256で行った。なお、ここに示した値はレンダリングのみの時間であって、水の移動計算の時間は含まれない。

Height Field の大きさ	FPS
16 × 16	25.027
32 × 32	7.971
64 × 64	2.059
128 × 128	0.511

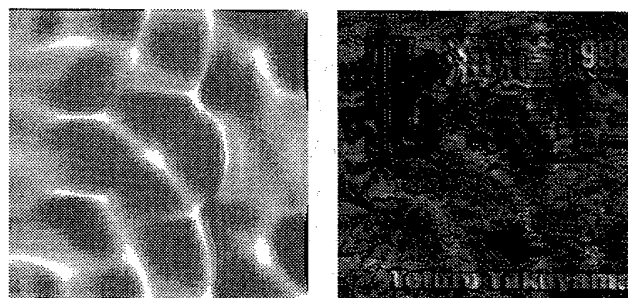


Fig.2 生成した光の模様（左）と、光の模様によって陰影づけられたテクスチャ（右）

6 まとめ

今回は、水の深さによる光の減衰などは考慮しなかったが実装することは容易であろう。また、一時屈折しか考慮していないので、水面に入ったレイが再び水面から出て行く場合などには不都合が生じるなどの問題点が存在する。

参考文献

- [1] 徳山哲朗、山本強、「水流のモデリングとレンダリングに関する研究」情報処理北海道シンポジウム'97 講演論文集 pp.37-39
- [2] Michael Kass and Gavin Miller "Rapid, Stable Fluid Dynamics for Computer Graphics" Proc. SIGGRAPH'90(August 1990), pp.49-57

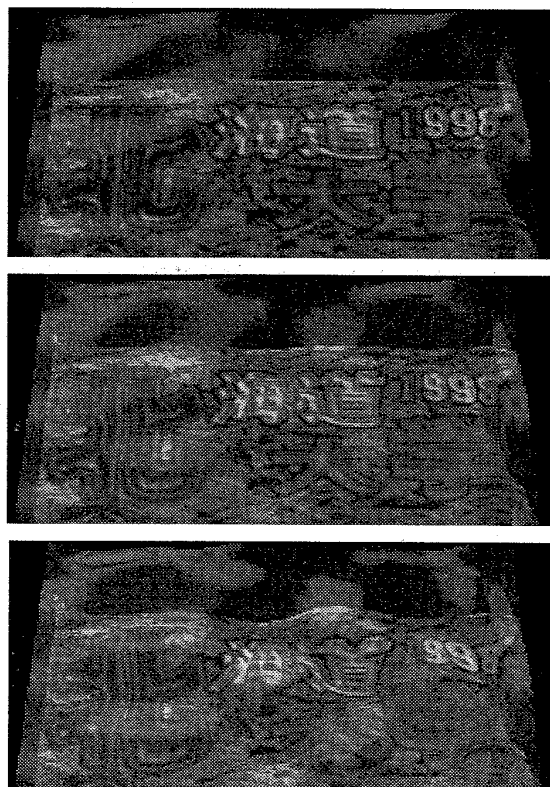


Fig.3 屈折の様子