LC-009

GPUによる計算機合成ホログラムの高速化

Computer-Generated Hologram by Graphics Processing Unit

阿部 幸男† 田中 喬† 白木 厚司† 市橋 保之† 増田 信之† 伊藤 智義†
Yukio Abe Takashi Tanaka Atsushi Shiraki Yasuyuki Ichihashi Nobuyuki Masuda Tomoyoshi Ito

1. まえがき

ホログラフィは物体光の波面をそのまま記録・再生できる唯一知られた技術であり，究極の三次元映像技術であるともいわれている．ホログラフィでは三次元の情報をホログラムに記憶するが，ホログラムは光の干渉を数値シミュレーションで求めることも可能であり，計算機合成ホログラム（Computer-Generated Hologram ：CGH）と呼ばれている．しかしホログラムの持つ情報量が膨大なため，CGH の作成には相当な時間を要する．これがホログラフィを用いた三次元動画像システムの大きな問題点の一つとなっている[1].

計算高速化の一手法として，近年，性能の向上が著しく進んでいる GPU（Graphics Processing Unit）の利用が考えられている[2][3]．GPU は内部にシェーディング処理を行うパイプラインを持ち，並列処理を行うことで，グラフィックス処理においては，CPU をはるかに上回る高い演算能力を得ている．近年，シェーダはプログラム可能になり，32bit 浮動小数点精度の 4 次元ベクトルプロセッサとして，数値計算にも応用できる構造になっている．

本研究では，最新の GPU の一つである GeForce8800 GTX（nVIDIA 社）に CGH 計算のアルゴリズムを実装して並列計算を行った．シェーダ数は 128 で，1.35 GHz で動作する．CPU（Central Processing Unit）単独での計算と比べて，約 2,000 倍の高速化に成功したので，報告する．

2．計算機合成ホログラムの作成

三次元物体を構成する点数を N_{obj} とすると，ホログラム上の点（x_α, y_α）において，以下の式を計算することで CGH を求めることができる．

$$I(x_\alpha, y_\alpha) = \sum_{j=1}^{N_{obj}} A_j \cos\theta \qquad (1)$$

$$\theta = \frac{2\pi}{\lambda}\sqrt{(x_\alpha - x_j)^2 + (y_\alpha - y_j)^2 + z_j^2} \qquad (2)$$

変数 α はホログラム点を，j は物体点を表わす．A_j は物体点の輝度であり，λ は三次元情報の記録・再生時に使われる参照光の波長である．

ここで，物体の z 座標を x, y 座標に比べて十分大きくとると，（2）式の代わりに（3）式のフレネル近似式が使えて計算負荷を軽減できる．本研究では（3）式を用いた．

$$\theta = \frac{\pi}{\lambda z_j}\{(x_\alpha - x_j)^2 + (y_\alpha - y_j)^2\} \qquad (3)$$

ホログラム上の点数を N_{hol} とすると，1 枚の CGH を作成するためには，（1）及び（3）式を α=1 から N_{obj} まで N_{hol} 回繰り返して計算する．したがって，計算量は $N_{obj} \times N_{obj}$ に比例し，膨大なものになる．ただし，（1）式の計算では各 α 間に相関はなく，独立に行うことができる．そのため，並列計算が有効である．

3. シェーダ・プログラミング

GPU に CGH 計算のアルゴリズを実装する手順を図 2 に示す．まず，頂点シェーダにおいて，4 つの頂点をスクリーンの端にくるようにセットする．次に，ラスタライザの自動補間機能を用いて，4 角形ポリゴンを描画するという命令を発行すると，その頂点に囲まれたピクセルが生成される．そして，ピクセルシェーダにおいて，補間されたテクスチャ座標データをホログラム面の座標データとして利用して，CGH 計算を行う．

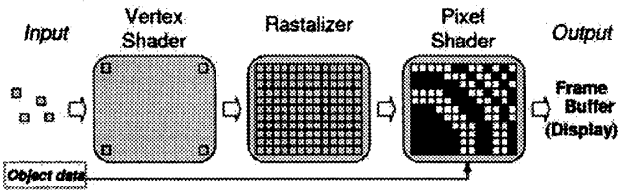


図 2．CGH アルゴリズム実装のフロー・チャート

ピクセルシェーダでのプログラムは以下の通りである．

```
float4 PS (VS_OUTPUT In) : COLOR
{
   float4 Xaj = 0.0f, Yaj = 0.0f, I = 0.0f;
   for (int j=0; j<DATA; j++) {
      Xaj = In.Text0.x – model_x[j];
      Yaj = In.Text0.y – model_y[j];
      I += cos(Z[j]*(Xaj*Xaj + Yaj*Yaj));
   }
   return I.x + I.y + I.z + I.w
}
```

4. マルチパス・レンダリング

CGH 計算を並列に行うためには，物体点データはピクセルシェーダ内のレジスタに送る．GeForce8800 シリーズ以前の GPU では，全ての物体点データを送ってから計算を始める方法（シングルパス・レンダリング）がもっとも高速だった．ところが GeForce8800GTX では，分割してデータを送って計算する方法（マルチパス・レンダリング）の方が高速な結果を示した．

具体的に示すために，N_{hol} を n の倍数として（1）式の右辺を（4）式に書き換え，n を変化させたときの計算速度を図 3 に示す．縦軸は 1 秒間に計算できる物体点数で，数値が大きいほど計算性能が高いことを示している．

† 千葉大学大学院工学研究科
Graduate School of Engineering, Chiba University

$$\sum_{j=1}^{n} A_j \cos\theta + \sum_{j=n+1}^{2n} A_j \cos\theta + \cdots + \sum_{j=N_{obj}-n+1}^{N_{obj}} A_j \cos\theta \quad (4)$$

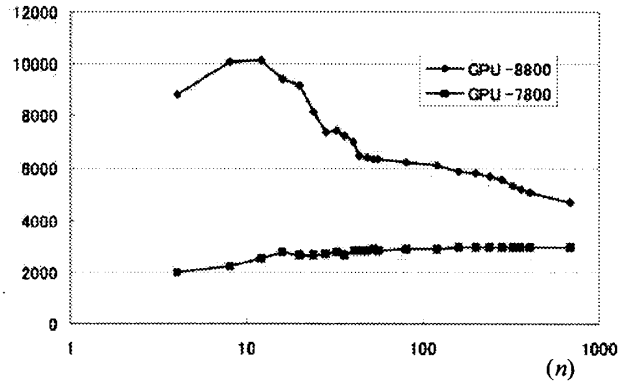


図 3．マルチパス・レンダリングの計算性能

比較のために，前シリーズの GeForce7800GTX の計測結果も図示した．GeForce7800GTX では，シングルパス・レンダリングがもっとも高速なのに対して，GeForce8800 GTX では n=12 がもっとも高速であり，シングルパス・レンダリングに比べて 2 倍近い速度比を示した．そこで本研究では，n=12 のマルチパス・レンダリングを採用した．

5. 結果

CPU 単独計算と GPU（＋CPU）計算との比較を表 1 に示す．解像度 1,920×1,080 の CGH を 1 枚作成する時間で比較している．使用したシステムは以下の通りである．

CPU: Intel Pentium 4 3.4-GHz, Memory: 2GB
OS: Linux Fedora Core 6 kernel-2.6.19
Compiler: gcc 4.0 (option -3)
GPU: Geforce8800GTX
Graphics API: OpenGL, Shading Language: Cg

表 1．CPU 計算と GPU 計算の比較

物体点数	CPU（秒）	GPU（秒）	速度比
96	16.39	0.0087	1880
288	47.09	0.0259	1820
480	77.98	0.0421	1850
672	108.76	0.0587	1850
960	184.98	0.0832	2220
1920	360.45	0.1651	2180
4800	880.93	0.4131	2130
9600	1748.07	0.8258	2120

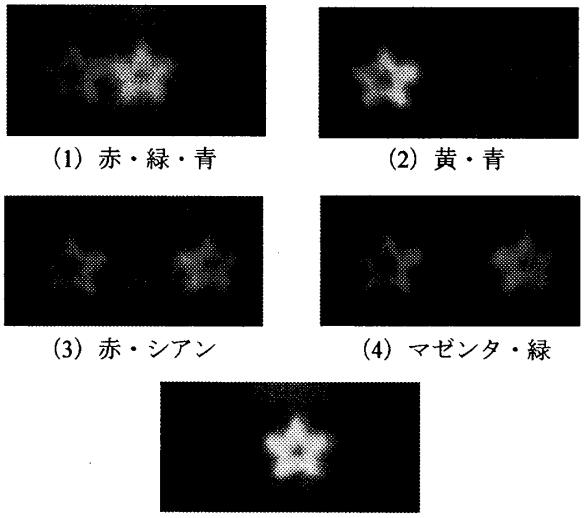
計算結果は，GPU（＋CPU）による計算が CPU 単独の場合と比較して，CGH 計算を 1,800～2,200 倍高速化することを示している．

この要因として，GPU はコサイン計算を高速に行うアーキテクチャを内包していることが考えられる．コサインのみの計算速度を実測したところ，GPU の計算速度は CPU に比べて 1,600 倍高速だった．

また，表 1 より，GPU を用いることで，物体点数が 500 点程度であれば，ビデオ・レート（30 フレーム／秒）に近いリアルタイムの電子ホログラフィ再生が可能なことがわかる．

図 4 は，GPU による計算システムに光学系を組み合わせて行った電子ホログラフィ再生の例で，三原色（赤・緑・青）の発光ダイオード（LED）を参照光源に用いたリアルタイムのカラー動画のスナップショットである．

はじめに，三原色それぞれの星を表示する（1）．一つの星は 64 点で構成されており，物体点の総数は 192 である．中央の緑の星を左に動かし，赤の星と重なると，混色の黄色に変わり（2），逆に動かして緑と青を重ねるとシアンになり（3），赤と青を重ねるとマゼンタになる（4）．最後に，赤，緑，青の星を全て重ねると白の星に変化する．簡単なシステムでありながらも，電子ホログラフィのカラー再生をきれいに行うことに成功した．



（1）赤・緑・青　（2）黄・青

（3）赤・シアン　（4）マゼンタ・緑

（5）白

図 4．電子ホログラフィ再生例

6. まとめ

GPU の特性を考慮して CGH 計算式を実装した結果，CPU のみの計算に比べて，約 2,000 倍の高速化を実現した．

GPU は安価で一般的なリソースであることから，本研究結果は，電子ホログラフィの研究分野において，有用な知見を与え得るものと考えている．

参考文献

[1] 本田捷夫他, ":高度立体動画像通信プロジェクト最終成果報告書", 通信・放送機構 (1997).

[2] N. Masuda, T. Ito, T. Tanaka, A. Shiraki and T. Sugie, "Computer generated holography using a graphics processing unit", Opt. Express, 14, 603-608 (2006)

[3] 白木厚司, 阿部幸男, 田中喬, 根尾敦, 増田信之, 伊藤智義, "GPU とプロジェクタを用いた簡易なリアルタイム電子ホログラフィ再生システム", 映像情報メディア学会誌, Vol.61, No.4, pp.1-6 (2007)

[4] "NVIDIA CUDA Compute Unified Device Architecture, Programming Guide, Version 0.8" (2007)