

J-034

GPU による計算機合成ホログラムの高速化

Computer Generated Hologram using Graphics Processing Unit

増田信之† 田中喬† 白木厚司† 伊藤智義†*

Nobuyuki Masuda Takashi Tanaka Atsushi Shiraki Tomoyoshi Ito Takashige Sugie

1. まえがき

近年、3次元動画の表示技術の研究開発がさかんに行われており、数値シミュレーション結果の可視化や医療分野における人体情報の可視化など様々な分野での応用が期待されている。本研究では物体の光の干渉と回折を利用して3次元動画を記録再生するホログラフィ技術に焦点をあてる。

ホログラフィでは3次元の情報をホログラムに記憶するが、そのホログラムは光の干渉をシミュレートすることにより計算機で作成することができる。そのようにして作成されたホログラムは計算機合成ホログラム(CGH: Computer Generated Hologram)と呼ばれる。しかしCGHの作成にはその膨大な演算量のために相当な時間がかかってしまう。これがホログラフィを用いた3次元動画システムの大きな問題点の一つとなっている。

現在までに我々の研究室では、この問題を解消するためにFPGA(Field Programmable Logic Array)を用いた専用計算機(HORN(HOolographic ReconstructioN))を開発してきた。

本研究では近年、CPU以上に高速化のスピードが上がっているGPU(Graphics Processing Unit)に着目し、CGH計算の高速化を目的としてCGHの計算をGPUを用いて行い、CPUとその計算速度を比較することでGPUによるCGH計算の有効性を検証することを目的としている。

2. 計算機合成ホログラムの作成

記録する3次元物体がN点で構成されているとすると、式(1)を計算することによってCGHを作成することができる。

$$I(x_\alpha, y_\alpha) = \sum_j^N \cos(2\pi\theta) \quad (1)$$

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

I はホログラム面上 (x_α, y_α) での光の強度、 p はホログラムの画素間隔、 λ は参照光の波長、 (x_j, y_j, z_j) は物体点の座標である。また、物体のz座標がx,y座標に比べて十分大きくなるように配置することで式(2)を式(3)のように近似することができ、こうすることでルート計算をなくす

ことが出来、計算負荷を軽減できる。

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

ここで式(3)の第1項を省略してもホログラムを作ることができる。本研究では式(1)と式(3)の第1項を除いた式を用いてCGH計算を行った。

3. GPU

コンピュータグラフィックス(CG)の分野では非常に多くの浮動小数点演算が求められることが多く、CPUではリアルタイム処理に追いつくことが不可能である。そこでGPUのようなグラフィックス処理に特化した専用の演算装置が用意されている。

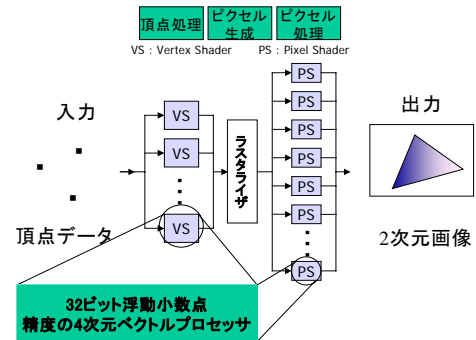


Fig.1 : GPU パイプライン概略図

GPUは固定された内部処理のパイプラインを持ち、複数のベクトル演算装置で並列処理を行うことでCPUを上回る高い演算能力を得ている。Fig.1にそのパイプラインの概略を示す。GPUのパイプラインに入力された頂点のデータは頂点処理を行う「Vertex Shader」に送られ、その後、ピクセル生成を行う「ラスタライザ」に送られ、さらにピクセル処理を行う「Pixel Shader」で処理され、2次元画像としてモニタなどの画面に出力される。そして近年、パイプラインの一部を自由にプログラム可能にしたGPUが登場した。パイプライン中で自由にプログラム可能になったのは、Vertex ShaderとPixel Shaderである。

それぞれ、32bit 浮動小数点精度の4次元ベクトルプロセッサが並列に組み込まれている。本研究ではこれらのプログラム可能なプロセッサを用いて、GPUによるCGH計算を行い、その高速化を評価した。

4. 結果

70点で構成された物体(星型)から48万画素(800×600)のホログラムをCPUおよびGPUを用いて計算したときの計算時間をTable1に示す。また作成されたCGHを用いて、ホログラム再生用の光学系で再生した再生像をFig.2に示す。

本研究で用いたGPUはnVIDIA社 GeForce 6600で、Vertex Shaderが3並列、Pixel Shaderが8並列されている。前節で述べたように、GPUでは、Vertex ShaderとPixel Shaderの両方がプログラム可能であるが、この計算では、より並列度の高いPixel Shaderを用いてCGH計算を行っている。また、GPUの計算においては、グラフィックスAPIにDirectX 9.0cを使用し、上位レベルシェーダ言語として、HLSL(High Level Shader Language)を使用した。

Table1: CGH 計算時間

CPU		GPU	
2980	[ms]	67	[ms]

また、計算に用いられたCPU、GPUは以下のとおりである。

Table 2 : CPU 及び GPU

CPU	Intel Pentium 4 3.2GHz
Memory	2GB
OS	Linux
Compiler	Intel C++ Compiler
GPU	nVIDIA GeForce 6600

CGH作成の計算時間を比べるとGPUを用いた場合、CPUを用いた場合より約45倍の計算速度が得られた。これはGPUの並列性と高い演算能力をCGH計算に応用できたことと結果といえる。



[] CPU [] GPU

Fig.2 再生像

5. まとめ

本研究ではGPU上でCGH計算(物体点数70点,ホログラム画素数800×600点)を実装することでCPUの約45倍の高速化を図ることに成功した。GPUは並列ベクトルプロセッサであり、その高い並列性を活かすことができるような計算にはCPUを上回る高速な処理ができると言える。CGH計算を考えると、各ピクセルにホログラムの座標データをセットすることでその並列性を活かすことができ、さらにすべてのピクセルで同じ計算を行うようになっている。つまりCGH計算はGPUの構造に適していると言える。しかし今回開発したシステムでは一度にGPU内に保持できる物体点数が100に制限されてしまうという問題点を抱えている。これはGPU内に搭載されているレジスタの数によるものであるが、このことに関しては、物体点をいくつかのグループに分け、その結果をグラフィックカード上のVRAMに一時的に保存することで、解決することができる。ただ、VRAMからのデータの読み込みには、約2msの時間がかかり(ホログラムの画素数が512×512の時)、このことが、速度向上の妨げになるが、さらに性能が向上したGPUを使用することで、計算の高速化が期待できる。現在のハイエンドなGPUには、Vertex Shaderが6並列、Pixel Shaderが16並列されている。また、物体点数やホログラムの画素数に制限はあるが、秒間30枚のホログラムの作成が可能であり(物体点47点,画素数800×600)、実際に研究室にあるホログラム再生用の光学系を用いることで、3次元動画のリアルタイム処理が可能になっている。

6. 文献

- [1] 辻内順平：“ホログラフィー”，裳華房(1997)
- [2] T. Ito, N. Masuda, K. Yoshimura, A. Shiraki, T. Shimobaba and T. Sugie “A special-purpose computer for electroholography HORN-5 to realize a real-time reconstruction,” Optics Express, Vol.13, pp1923-1932(2005)
- [3] Randima Fernando, (監訳)中本 浩：“GPU Gems 日本語版”，ポーンデジタル(2004)
- [4] 今給黎 隆：“DirectX 9シェーダプログラミングブック”，毎日コミュニケーションズ(2004)