

時系列ボリュームを高速レンダリングするための 2段階データ圧縮方法の性能および画質評価

永安大輔[†] 伊野文彦[†] 萩原兼一[†]

本稿では、GPUを用いた時系列ボリュームレンダリングを高速化するための2段階圧縮方法を提案する。提案方法は転送時間を短縮することを目的として、ディスクからCPUだけでなくCPUからGPUへも圧縮データを転送する。また、レンダリング時においてCPUがGPUの展開を補助する。そのために、GPUで動作する圧縮方法(PVTC)とCPUで動作する圧縮方法(LZO)を組み合わせてデータを2段階に圧縮する。提案方法を5種類のデータセットに適用した結果、1段階圧縮のみの場合と比較して秒間あたりの描画フレーム数を最良で30%向上できた。その際、画質に関してはPSNR値で30~50dB程度であり、圧縮に伴う画質劣化を許容範囲に収めることができた。

Performance and Image Quality Evaluation of Two-Stage Compression Method for Accelerating GPU-Based Volume Rendering of Time-Varying Data

DAISUKE NAGAYASU,[†] FUMIHIKO INO[†] and KENICHI HAGIHARA[†]

This paper presents a two-stage compression method for accelerating GPU-based volume rendering of time-varying data. In order to reduce transfer time, our method compress not only the data transmitted from the storage to the CPU but also that from the CPU to the GPU. Our method also makes the CPU assist the GPU in run-time data decompression. To achieve this two-stage compression method, we combine packed volume texture compression (PVTC) and Lempel-Ziv-Oberhumer (LZO) compression, each running on the GPU and the CPU, respectively. We also present evaluation results using five datasets. At the best case, the number of frames per second is increased by 30%, as compared to a single-stage compression method. With regard to the quality of images, we obtain good quality ranging from 30 to 50 dB in terms of PSNR.

1. はじめに

時系列VR (Volume Rendering) は、時系列の3次元データ (時系列ボリューム) を順に可視化する技術である。時系列VRは時間経過に着目したデータ解析を支援でき、3次元データの時間的な変化を直感的に観察する際に有用であるため、複雑なシミュレーション結果の解析に利用されている。

時系列ボリュームは4次元の情報であるため、情報量が多い (数GB~数TB)。ゆえに、時系列ボリュームの全体を主記憶上に保持することは容易ではなく、ディスクなどの2次記憶でデータを保持し、必要に応じて主記憶へ転送することになる。

主記憶量不足に陥るような時系列ボリュームに対し、

高速な可視化を実現するために解決すべき課題は次の2点である。(1) 高速なVRおよび(2) ディスクなどの2次記憶からの高速なデータ転送。これらの課題に対し、既存研究^{1),2)}ではGPUによるVRの高速処理、およびデータ圧縮による高速なデータ転送を図っている。これらの研究では、圧縮済のデータをGPUへ転送し、それらをGPUで展開する。しかし、GPUのみを展開に用いる場合、GPUに読み書き兼用メモリが存在しない、GPUにおける条件分岐処理が高速でないなどの理由により、高圧縮かつ短時間展開可能な圧縮方法の実現は容易ではない。一方、CPUのみを展開に用いる場合、非圧縮データを主記憶からGPUへ転送するため、このための転送時間が性能向上を妨げる。我々の知る限りCPUおよびGPUの双方を用いてデータ圧縮・展開機構を開発した試みはない。

本稿では、課題(1)および(2)を解決し、高速な可視化を実現するために、GPUで動作する圧縮方法

[†] 大阪大学大学院情報科学研究科コンピュータサイエンス専攻
Department of Computer Science, Graduate School of
Information Science and Technology, Osaka University

(以下 MG) と CPU で動作する圧縮方法 (以下 MC) を組み合わせる 2 段階データ圧縮方法を提案する。圧縮方法としては、MG には GPU がハードウェアとして備える圧縮機能 VTC (Volume-Texture Compression)³⁾ を用いる。一方、MC には実時間展開を目的とする LZO (Lempel-Ziv-Oberhumer)⁴⁾ を用いる。前者については、VTC に RGB チャネルへのデータ詰込を組み合わせることでより高い圧縮率を達成する PVTC を提案する。さらに、PVTC (Packed VTC) について、時間類似性および空間類似性 (2 章) のそれぞれを重視して圧縮をする PVTC-TE (PVTC for Temporal Encoding) および PVTC-SE (PVTC for Spatial Encoding) を提案する。提案方法は、ディスク・主記憶間だけでなく、主記憶・GPU 間においても圧縮データを転送する。また、CPU による展開が GPU を補助することで、性能の向上を図る。

以降では、まず 2 章で GPU を用いた既存の時系列 VR および圧縮方法を示す。次に 3 章および 4 章で提案方法およびその評価実験の結果を示す。最後に 5 章で本稿をまとめる。

2. GPU を用いた時系列 VR

GPU による VR では、TBVR (Texture-Based VR)^{5),6)} を用いることが多い。TBVR は、ボリュームをテクスチャとして保持し、その断面を視線方向の奥から順に作成し、それらを半透明合成する。TBVR は、ボリュームの断面図作成・半透明合成に GPU がハードウェアとして備える機能 (テクスチャマッピング機能・アルファブレンディング機能) を使用できるため、高速に処理できる。

しかし、時系列ボリュームは情報量が多いため、時系列 VR を高速に処理するためには、高速な VR に加えて転送時間の短縮も必要である。そこで、既存研究^{1),2)} は、時系列ボリュームが持ち得る以下の類似性に着目してデータを圧縮している。

時間類似性 同一座標において、時間が近いボクセル同士が近い数値を持つ

空間類似性 同一時刻において、空間的に隣接するボクセル同士が近い数値を持つ

Lum ら¹⁾ は、時間類似性に着目して圧縮をしている。具体的には、数時刻分のボクセル値をすべて保存するのではなく、平均値および平均値との差の一部のみを保存する。そのために数時刻分のボクセル値に対して DCT (Discrete Cosine Transform) を適用して複数の周波数成分に分解し、分解した周波数成分から低周波成分のみを抽出し、抽出した成分を量子化して

1 時刻分のデータ領域に詰込む。

Binotto ら²⁾ は、時間類似性および空間類似性に着目し、重複する値を代表値に置き換えることによりデータサイズを削減する。時間類似性に対しては、全時刻のボリュームをブロックに分割し、内容が同じブロックを 1 個のブロックに置き換える。空間類似性に対しては、ブロック中の全ボクセル値が同じ値である場合、そのブロックを 1 個の代表値に置き換える。

3. 提案する 2 段階圧縮

本章では、提案する 2 段階圧縮方法について述べる。まず、2 段階圧縮の設計の指針について述べ、次に PVTC および LZO を用いる際の圧縮・展開について述べる。

3.1 設計の指針

提案方法は、MC と MG を組み合わせる。以降では、2 段階圧縮により高速な時系列 VR を実現するために MC および MG が満たすべき条件を示す。

まず、2 段階圧縮が否かに関わらず、転送時間を短縮するために MC および MG が満たすべき条件は以下である。

(I) MC および MG が、展開時間に見合うだけの圧縮率を達成すること

次に、MC および MG を 2 段階圧縮で用いる際に満たすべき条件は以下である。

(II) MC および MG が互いに異なる類似性に着目してデータを圧縮すること

(III) MG が生成する圧縮済データに MC が着目する類似性が存在すること

条件 (II) は、同じ類似性に着目した圧縮を繰り返しても、効率のよい圧縮とは言えないことを意味する。この条件を満たす例としては、MC が時間類似性に着目し、MG が空間類似性に着目する圧縮などが挙げられる。一方、条件 (III) は MC および MG が条件 (I) および (II) を独立に満たすような圧縮方法であったとしても、それらの相性が悪い場合、よい 2 段階圧縮とは言えないことを意味する。

提案方法では、条件 (I) ~ (III) を満たすものとして、MG として PVTC を用い、MC として LZO を用いる。PVTC は、RGB チャネルへの詰込と VTC³⁾ を組み合わせた圧縮方法であり (3.2 節)、LZO⁴⁾ は既存の圧縮方法である。

図 1 に提案方法の概要を示す。PVTC および LZO は、以下のような理由から高速に展開できるため、条件 (I) をみताす。

- VTC は GPU がハードウェアの機能として備え

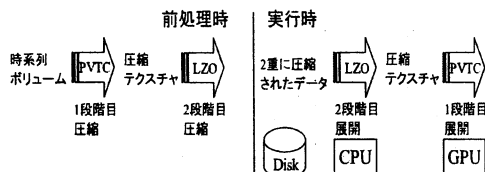


図 1 提案方法の概要

る圧縮方法である

- RGB チャンネルに詰込まれたデータは GPU 上で並列に取り出し可能
 - LZO の展開アルゴリズムは単純であり、かつ、展開時に CPU のキャッシュメモリを有効に利用する
- また、以下のように着目する類似性が異なり、条件 (II) をみす。

- PVTC はボリューム内部の微小ブロックがもつ時間類似性および空間類似性に着目する
 - LZO はより広い範囲に存在する類似性に着目する
- なお、条件 (III) を満たすことは後述する (3.2.3 節)。

3.2 圧縮

PVTC と LZO を用いた 2 段階圧縮は、以下に示す 3 つのステップからなる。

- (1) RGB チャンネルへの詰込み
- (2) VTC による圧縮
- (3) LZO による圧縮

(1) および (2) が PVTC による 1 段階目の圧縮である。なお、PVTC は非可逆圧縮であり、圧縮率は約 6 倍に固定である。また、LZO は可逆圧縮である。

PVTC について、圧縮する際に時間類似性を重視する PVTC-TE、および空間類似性を重視する PVTC-SE の 2 種類の実装を提案する。

PVTC-TE RGB チャンネルへ同一座標の連続する 3 時刻のデータを詰込む

PVTC-SE RGB チャンネルへ同一時刻の連続する 3 座標のデータを詰込む

3.2.1 PVTC-TE による圧縮

PVTC-TE は微小ブロック ($4 \times 4 \times 1$ ボクセル) における空間類似性および短時間 (3 時刻) における時間類似性を用いてデータを圧縮する。図 2 に圧縮の概要を示す。以下に圧縮の各ステップについて詳細を示す。

- (1) RGB チャンネルへの詰込み：

時間類似性を利用するために、連続する 3 時刻分の時系列ボリュームを RGB の 3 チャンネルをもつボリュームへ詰込み、詰込済ボリュームを作成する。すなわち、 $w[t](x, y, z)$ がもつ RGB の各チャンネルに $v[3t](x, y, z)$, $v[3t+1](x, y, z)$,

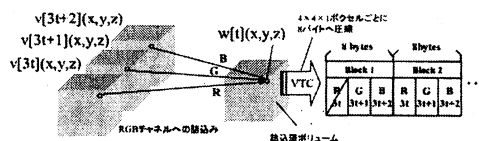


図 2 PVTC-TE を用いた圧縮

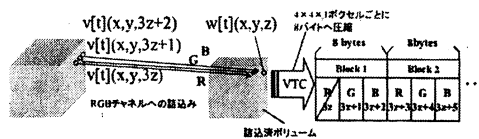


図 3 PVTC-SE を用いた圧縮

$v[3t+2](x, y, z)$ を保存する。ここで、 t 時刻目のボリュームおよび t 個目の詰込済ボリュームの (x, y, z) 座標のボクセルをそれぞれ $w[t](x, y, z)$ および $v[t](x, y, z)$ とおいた。

- (2) VTC による圧縮：

空間類似性を利用するために、各詰込済ボリュームをそれぞれ $4 \times 4 \times 1$ ボクセルからなるブロックの集合に分割する。続いて、各ブロック (48 バイト) を 8 バイトからなるデータに圧縮する。この 8 バイトのデータは、それぞれ圧縮前のブロックを代表する 2 つの値 (各 2 バイト) と、ブロック内のボクセル値の分布状況 (4 バイト) からなる。以後、圧縮した詰込済ボリュームを圧縮テクスチャと呼ぶ。

3.2.2 PVTC-SE による圧縮

PVTC-TE は微小ブロック ($4 \times 4 \times 3$ ボクセル) における空間類似性を用いてデータを圧縮する。図 3 に圧縮の概要を示す。以下に圧縮の各ステップについて詳細を示す。

- (1) RGB チャンネルへの詰込み：

空間類似性を利用するために、時系列ボリュームの各ボリュームについて、奥行方向に並ぶ 3 ボクセルを RGB の 3 チャンネルへ詰込み、詰込済ボリュームを作成する。すなわち、 $w[t](x, y, z)$ がもつ RGB の各チャンネルに $v[t](x, y, 3z)$, $v[t](x, y, 3z+1)$, $v[t](x, y, 3z+2)$ を保存する。

- (2) VTC による圧縮：

PVTC-TE の場合と同様にして圧縮する。

3.2.3 LZO による圧縮

LZO は、入力されたデータを 1 次元データとして端から参照し、互いに重複する箇所に着目し、それらを削減することによりデータサイズを減らす圧縮方法である。

圧縮テクスチャは詰込済ボリュームの微小ブロックの代表値・分布状況を表す値が連続する1次元データである。詰込済ボリュームの微小ブロックに互いに等しいもの、および非常に類似するものが存在する場合、圧縮テクスチャに重複した値が存在する。2段階目の圧縮では、この重複値に着目し、LZOによりさらにデータサイズを削減する。

3.3 展 開

提案方法の展開は、以下の5つのステップからなる。

- (1) 2重に圧縮されたデータをディスクから主記憶へ転送
- (2) CPUがLZOにより主記憶上のデータを展開
- (3) 圧縮テクスチャを主記憶からGPUへ転送
- (4) GPUがVTCにより圧縮テクスチャを展開
- (5) 適切なRGBチャンネルを選択し、ボクセル値を取得

4. 適用実験

提案方法を評価するために、レンダリングの性能および画質を測定した。

実験環境として用いたPCは、Pentium 4 3-GHzのCPU、2GBの主記憶、Serial ATA 150GBのディスクおよびGPUとしてnVIDIA GeForce 6800 GTO（ビデオメモリ容量256MB）を備える。提案手法の実装にはC++、OpenGLおよびCg（C for graphics）⁷⁾を用いる。また、VRの実装にはTBVRを用いる。

表1に、使用した5つのデータセットD1～D5の詳細を示す。図4にD1、D2、D3のレンダリング結果を示す。D1は肺が変形する過程を記録したものである。時刻間の変形が微小であることから、時間類似性が高い。また、肺の隣り合うボクセルは値が類似しており、さらに透明領域が多数存在することから、空間類似性も高い。D2は乱流状態にある渦がボリューム内を移動する様子を示す。渦が一点にとどまらないため、時間類似性は低い。また、隣り合うボクセルとの値も類似しておらず、空間類似性も低い。D3は乱流状態にある噴流が、ボリューム端の狭い領域から対面に向けて噴射する様子を示す。噴流が移動するため、時間類似性は低い。一方で、透明領域が多数存在するため、空間類似性は高い。D4およびD5は各々D2およびD3を3次元補間することで各辺を2倍に拡大したデータセットである。

4.1 レンダリング性能

D1、D4およびD5については 512^2 ピクセルからなる投影面を使用し、D2およびD3については 256^2 ピクセルからなる投影面を使用した。また、ボリューム

表1 実験で用いたデータセット

データセット	サイズ	時刻数	時間類似性	空間類似性
D1	256 × 256 × 148	81	高	高
D2	128 × 128 × 128	99	低	低
D3	129 × 129 × 104	99	低	高
D4	256 × 256 × 256	99	低	低
D5	258 × 258 × 208	99	低	高

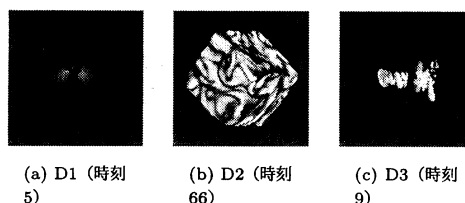


図4 レンダリング結果の例

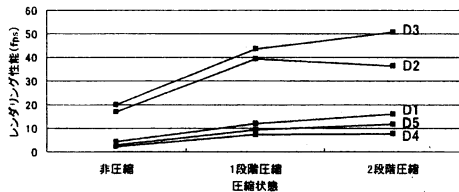
ムを次の時刻のものへ切り替えるごとに、視点をx軸、y軸の周りに2度ずつ回転させた。ここで、初期状態の視線方向をz軸としている。

図5に、秒間に描画できるフレーム数の変動を示す。また、表2に、各圧縮状態において、1ボリュームあたりの処理に要する時間を示す。

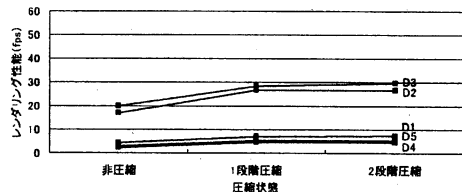
まず、PVTC-TEについて評価する。図5(a)より、2段階圧縮によって、全データセットについて非圧縮の場合と比べて3倍程度性能を向上できている。表2で、全体の処理時間 T の短縮時間が、ディスクからの読み込み時間 T_1 の短縮とほぼ等しいことから、2段階圧縮による性能向上はディスクからの読み込み時間の短縮が要因である。

D2を除く全データについて、2段階圧縮がPVTC-TEの単一よりも高い性能を示しているが、D2については2段階圧縮によりわずかに性能が低下している。この原因は、LZOによるデータサイズ削減がわずかであったため、 T_1 の短縮量がCPUにおけるLZOの展開時間 T_2 よりも小さいことが原因である。このことから、時系列ボリュームの時間類似性および空間類似性が小さく、LZOによるデータサイズの削減が小さい場合には、2段階圧縮を使用せず、PVTC-TEのみを使用して圧縮するほうがよい。

次に、PVTC-SEについて評価する。図5(b)より、2段階圧縮を使用しても、PVTC-SE単一と比べて性能を向上できていない。また、表2で、2段階圧縮によりディスクからの読み込み時間 T_1 を短縮できているが、その短縮が全体の処理時間 T の短縮につながって



(a) PVTC-TE による 2 段階圧縮



(b) PVTC-SE による 2 段階圧縮

図 5 提案方法によるレンダリング性能の変化

表 2 PVTC を用いた際のレンダリング実行時間 (各データは 1 時刻あたりの平均)

データセット	非圧縮		PVTC-TE						PVTC-SE					
			1 段階圧縮		2 段階圧縮				1 段階圧縮		2 段階圧縮			
	T		T		T				T		T			
	T ₁		T ₁		T ₁	T ₂			T ₁		T ₁	T ₂		
D1	201	223	33.9	84	4.5	3.3	63		45.0	141	13.9	4.2	136	
D2	52	59	11.0	25	10.8	2.5	28		15.8	38	14.9	2.0	38	
D3	41	50	8.9	23	4.4	2.2	20		15.8	35	3.7	1.7	34	
D4	351	404	56.6	138	31.8	14.4	128		76.9	214	47.0	20.1	215	
D5	291	319	49.3	108	15.3	10.0	85		63.2	186	29.4	7.8	186	

T : 1 ボリュームあたりの処理時間 (ミリ秒)

T₁ : ディスクからの読み込み時間 (ミリ秒)

T₂ : LZO 展開時間 (ミリ秒)

いない。読み込み時間を短縮できているにもかかわらず全体の性能が向上できていない理由は、PVTC-SE の展開アルゴリズムが PVTC-TE よりもやや複雑であるため、GPU における展開処理がボトルネックになっているためである。

予備実験により、PVTC-SE の GPU における展開処理が PVTC-TE と同程度の負荷で実現できれば、PVTC-SE においても PVTC-TE と同程度の性能向上を得られることを確認している。このことから、高速なレンダリングを実現するためには、GPU における展開処理の負荷が小さくなるように設計することが重要である。

4.2 圧縮率

表 3 に圧縮結果を示す。PVTC-TE、PVTC-SE 単一では約 6 倍にしか圧縮できないが、2 段階圧縮により、全データセットに対してより高い圧縮率を達成できている。

PVTC-TE を使用した 2 段階圧縮の場合、D1、D5 のように、1 段階の場合と比べた場合の圧縮率の向上が 3.7 倍以上である場合 (表 3) には、1 段階の場合よりもレンダリング速度を 1.3 倍程度向上できており (表 2)、2 段階圧縮が有用であるといえる。圧縮率の向上が 2 倍程度 (表 3) の D3、D4 に関しては、レンダリング速度はさほど向上しないが (表 2)、必要な

表 3 提案方法の圧縮率 (各データは 1 時刻あたりの平均、ファイルサイズの単位は KB)

データ セット	非圧縮 ファイル サイズ	PVTC-TE		PVTC-SE	
		1 段階	2 段階	1 段階	2 段階
		圧縮率	圧縮率	圧縮率	圧縮率
D1	9473	6.0	65.8	5.9	58.1
D2	2049		6.5	5.9	6.3
D3	1691		12.0	5.9	16.7
D4	16355		12.0	5.9	8.6
D5	13521		22.5	6.0	28.1

ディスク容量を 1/2 程度に削減できるため、2 段階圧縮は有用であるといえる。圧縮率の向上が 1.1 倍に満たない (表 3) D2 に関しては、性能が低下し (表 2)、必要なディスク容量の削減も十分でないため、2 段階圧縮は有用でない。

PVTC-SE については、展開時間が長いため、レンダリングの性能は向上できない。しかし、D3 および D5 に関しては PVTC-SE が PVTC-TE よりも高い圧縮率を達成する (表 3)。これは PVTC-SE が空間類似性を重視するからである。このことから、PVTC-SE の展開時間を短縮できれば、時間類似性が低く空間類似性が高い D3 および D5 のようなデータに関しては、PVTC-SE が有効であるといえる。

表 4 レンダリング結果画像の画質 (PSNR)

データセット	PVTC-TE (dB)	PVTC-SE (dB)
D1	47.7	44.6
D2	43.5	28.4
D3	31.5	31.8
D4	49.5	35.6
D5	35.7	39.9

4.3 画質評価

非可逆圧縮である PVTC-TE および PVTC-SE を用いた際の画質を評価するために、得られた画像について PSNR を測定した。PSNR は、圧縮前後での画質の差を評価するために広く用いられている指標で、値が高いほど画質劣化が小さい。なお、2 段階目に用いた LZO は可逆圧縮であるため、その部分の劣化はない。

表 4 に測定結果を示す。全データに対して PSNR 値で 30~50dB 程度を得ている。このことより、PVTC による画質劣化は許容範囲に収まっている。また、得られた PSNR 値は既存研究¹⁾と比較しても遜色がない。

空間類似性および時間類似性がともに高い D1 においては PVTC-TE および PVTC-SE 両方について 40dB 以上の PSNR 値を示し、画像劣化が最も小さい。また、空間類似性が低い D2 および D4 よりも、空間類似性が高い D3 および D5 が高い画質を示す。このことから、類似性が高いほど画質劣化を小さくできる。

D2 と D4、D3 と D5 を比べると、サイズの大きい D4、D5 がサイズの小さい D2、D3 よりも高い画質を示す。これは、サイズが大きくなるにつれて、圧縮する際に着目する $4 \times 4 \times 1$ 個のボクセルが全体に占める割合が小さくなり、誤差が小さくなるためである。

PVTC-TE と PVTC-SE を比較すると、空間類似性が高く時間類似性が低い D3 および D5 においては PVTC-SE が、それ以外のデータについては PVTC-TE がより良い画質を示している。このことから、一般的に、空間類似性が高く時間類似性が低い場合は PVTC-SE がより高い画質を示し、その他の場合は PVTC-TE がより高い画質を示す可能性が高い。

5. 結 論

本稿では、GPU を用いた時系列 VR を高速化するために PVTC と LZO を用いた 2 段階圧縮方法を提案した。結果、PVTC-TE と LZO を併用した場合には、非圧縮の場合と比べて時系列 VR に要する時間を 3 倍程度に高速化でき、PVTC-TE のみを用いる場合と比較して秒間あたりの描画フレームを最大で 1.3 倍

程度に増やせた。画質劣化に関しては、PSNR 値で 30~50dB 程度に抑えることができた。また、以下の 2 つの知見が得られた。

- 時系列ボリュームの時間類似性および空間類似性が小さい場合は、PVTC のみを使用して圧縮するほうがよい
- 高速なレンダリングを実現するためには、GPU における展開処理の負荷が小さくなるように設計することが重要である

今後はより大規模なデータに対する高速化に取り組んでいきたい。

謝辞 本研究の一部は、科学研究費補助金特定領域研究 (16035209) および (17032007) の補助による。また、データセットの一部には <http://www.cs.ucdavis.edu/~ma/ITR/tvdr.html> を使用した。深く感謝いたします。

参 考 文 献

- 1) Lum, E. B., Ma, K.-L. and Clyne, J.: A Hardware-Assisted Scalable Solution for Interactive Volume Rendering of Time-Varying Data, *IEEE Trans. Visualization and Computer Graphics*, Vol.8, No.3, pp.286-301 (2002).
- 2) Binotto, A. P. D., Comba, J. and Freitas, C. M. D. S.: Real-Time Volume Rendering of Time-Varying Data Using a Fragment-Shader Compression Approach, *Proc. IEEE Symp. Parallel and Large-Data Visualization and Graphics*, pp.69-76 (2003).
- 3) NVIDIA Corporation: NV texture compression VTC. http://oss.sgi.com/projects/ogl-sample/registry/NV/texture_compression_vtc.txt.
- 4) Oberhumer, M. F.: LZO data compression library. <http://www.oberhumer.com/opensource/lzo/>.
- 5) Cabral, B., Cam, N. and Foran, J.: Accelerated Volume Rendering and Tomographic Reconstruction Using Texture Mapping Hardware, *Proc. 1994 Symp. Volume Visualization*, pp.91-98 (1994).
- 6) Kniss, J., Engel, K., Hadweiger, M. and Rezk-Salama, C.: High-Quality Volume Graphics on Consumer PC Hardware, *SIGGRAPH 2002, Course Notes 42* (2002).
- 7) Mark, W. R., Glanville, R. S., Akeley, K. and Kilgard, M. J.: Cg: A System for Programming Graphics Hardware in a C-like Language, *ACM Trans. Graph.*, Vol.22, No.3, pp.896-907 (2003).