

大規模ボリュームレンダリング処理における 擬似透視投影法に関する考察

高山 征大 , 森 眞一郎

津邑 公暁, 五島 正裕, 中島 康彦, 富田 眞治 (京都大学)

Consideration to A Pseudo-Perspective Projection in Large Scale Volume Rendering

MOTOHIRO TAKAYAMA, SHIN-ICHIRO MORI, TOMOAKI TSUMURA,
MASAHIRO GOSHIMA, YASUHIKO NAKASHIMA and SHINJI TOMITA

ABSTRACT

In a large scale visualization with a wide screen, parallel projection becomes no more acceptable. However, real-time rendering using perspective projection is relatively expensive to support due to increasing irregularity in memory access patterns. So, we have been investigating a technique which we call *Pseudo-Perspective Projection*. This method simulates a perspective projection by a collection of multiple parallel projections. In this paper, we have examined the quality of images rendered with the pseudo-perspective projection.

Keywords: Volume Rendering, Perspective Projection, Large-Scale Data Set

1. 背景

近年の計算機処理能力の向上に伴い、より大規模なデータを可視化することが求められている。こうした可視化においては相応の大きさを持つスクリーンが必要となるが、そうしたスクリーンに対して平行投影を行うと、現実とかけ離れた画像が出力されてしまう。またこれまでのように、小さなスクリーンで見ていたものでも、プロジェクタなどにより大きなスクリーンに投影した場合には相当な違和感を感じる。

例として、Fig 1 に、 $256 \times 256 \times 128$ の大きさを持つ盆栽のボリュームデータを平行投影、透視投影を用いて 256×256 のスクリーンに可視化したものを示す。目とスクリーン、スクリーンと物体の距離の比は 20:1 程度である。下部にある、木とは異なる物体 X に注目すると、平行投影においては木に近いところに描画されているのに対し、透視投影では木から離れている。これは、実際には奥行き方向に関して X が木より手前にあるためであり、平行投影では奥のものも手前のもも同じように投影することに起因している。

このように、 256×256 程度の大きさのスクリーンにおいてさえ、平行投影による実像との歪みが大きいことは明らかである。

2. 擬似透視投影

このように透視投影の必要性が高まっているが、一方で透視投影は平行投影と比べた場合に計算量が大きいという問題がある。そこで本稿では、平行投影によって透視投影を擬似的に表現する、擬似透視投影法を提案する。

アルゴリズム

まず、物体を投影するスクリーン S (以下では便宜上、一辺が S_x の正方スクリーンとする) を、 N^2 (N : 一辺の分割数) 個のサブスクリーン $s_{i,j}$ ($i, j: [1..N]$) に等分割する。次に、各 $s_{i,j}$ における代表的な視線ベクトルとして、視点から各 $s_{i,j}$ の中心へのベクトル $v_{i,j}$ を計算する。 $s_{i,j}$ の各点に対して、得られた $v_{i,j}$ と平行なベクトルを用い、平行投影によってボリュームレンダリングを行う。最後に、以上のようにして得られた各 $s_{i,j}$ の画像を一つに合成することにより、全体 S の画像を擬似的な透視投影の結果として得る。この方法では、分割数 N が 1 の場合が並行投影に相当し、 $N = S_x$ の場合が透視投影に相当する。

例として、Fig 3 に $N = 2$ 、すなわち 4 分割の場合を示す。

利点

この手法による利点は二つある。

まず、視線に沿ったボクセルをメモリから読み込む際、隣

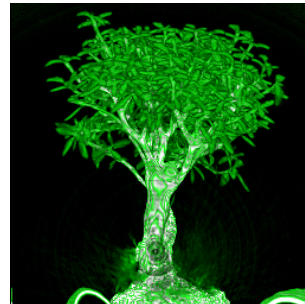


Fig. 1 平行投影

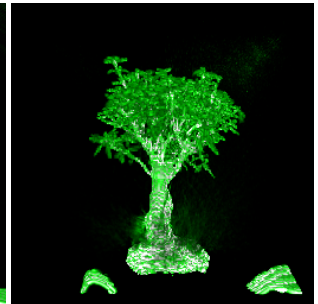


Fig. 2 透視投影

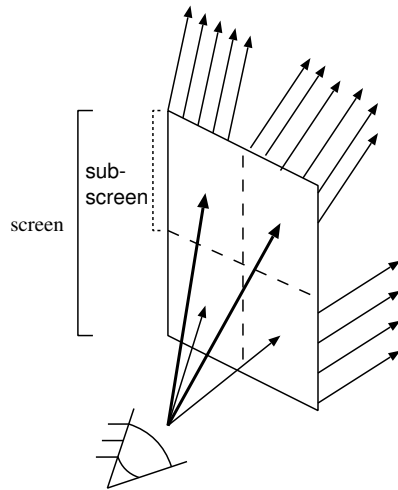


Fig. 3 擬似透視投影

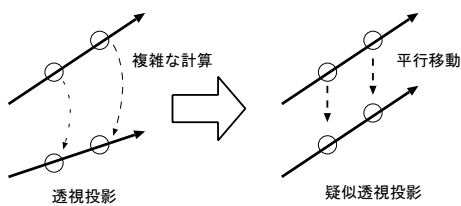


Fig. 4 擬似透視投影 - メモリアクセス

接するピクセル間で視線ベクトルはほぼ同じであることに着目する。このとき、透視投影の場合、視線ベクトルはスクリーン上の各ピクセル毎に異なるため、その都度視線ベクトルを再計算する必要があり、計算コストがかかる。また、視線が奥に進むにつれ視線間の間隔が広がり、メモリアクセスパターンの均質性が乱れる。このため、メモリへのアクセス時間の増加をもたらす (Fig 4)。

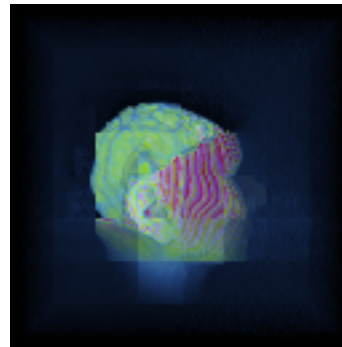
それに対し擬似透視投影の場合、サブスクリーンにおける視線ベクトルは全て同じであるので、隣接するピクセルに関するメモリアドレス計算は単純に平行移動を行うだけで求まる。このことにより、レンダリングハードウェアの実装を容易にできる。

また、高速なボリュームレンダリングのソフトウェア¹⁾や、実時間可視化が可能なハードウェア²⁾は、投影方法を平行投影のみに限っているものが多い。擬似透視投影を用いることにより、このような既存の技術を利用しつつ、透視投影による画像を得ることが可能であり、両者の長所を活かすことができる。

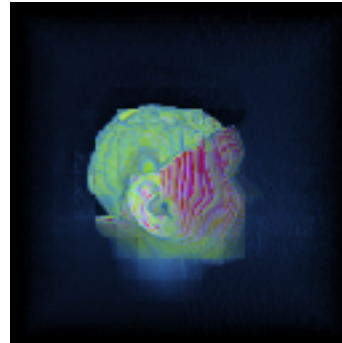
3. 評価, 考察

以上のような擬似透視投影法によって生成した画像と、透視投影による画像との画質の評価を行った。評価方法として、データとスクリーンの大きさ、分割数を変化させた場合について、直感的な観点、輝度差による統計的な観点、数値的な観点から考察する。

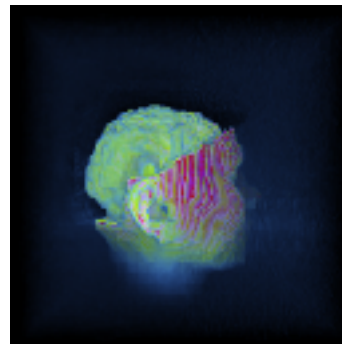
対象とするデータは、人間の頭部 (128 × 128 × 84)、盆栽 (256 × 256 × 128) のボリュームデータであり、それぞれ



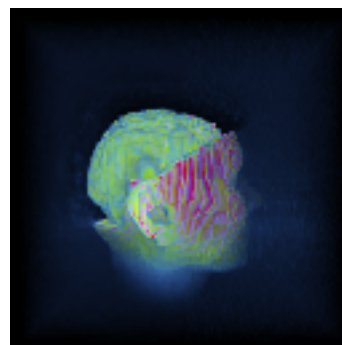
$N = 8$
($s = 16 \times 16$)



$N = 16$
($s = 8 \times 8$)



$N = 32$
($s = 4 \times 4$)



透視投影

Fig. 5 人の頭部 (データ : 128 × 128 × 84, スクリーン : 128 × 128)

128 × 128, 256 × 256 の大きさのスクリーン (S) に投影する。それぞれを S の分割数を変化させて擬似透視投影したものと、透視投影によってレンダリングしたものを Fig 5, Fig 6 に示す。分割数はどちらも上から順に、 $N = 8, 16, 32$ であり、それぞれのサブスクリーン (s) の大きさは図中の括弧内に示す通りである。以下では、サブスクリーンの大きさに着目して比較を行う。

主観的考察

まず、頭部の擬似透視投影について、透視投影の画像と比較する。

$s = 16 \times 16$ では、各 s において明らかに違和感を感じる。これは、 S と s の面積比が 8 : 1 程度という具合に、一つの s が全体に与える影響が大きいためだと考えられる。しかし $s = 8 \times 8$ では、首周りの階調具合や上部の曲線あたりに違和感を感じるものの、 16×16 のときと比べ、かなり改善されているのが分かる。 $s = 4 \times 4$ に至っては、透視投影の画像と見分けをつけるのが非常に困難である。

次に、盆栽の擬似透視投影についても同様に比較する。

$s = 32 \times 32$ では、中心からやや左下の領域について、大きな差が見られる。これも頭部の場合と同様に、 s が全体に与える影響が大きいためだと考えられる。次の $s = 16 \times 16$ では、幹の輪郭、色に大きな差を感じるが、 $s = 8 \times 8$ では、幹のあたりを除いては透視投影に遜色無い画像が得られている。

次に、擬似透視投影と透視投影の画像の各画素における RGB 値の差の絶対値を、そのまま画像として可視化する (Fig 7)。この図は、点が少なければ少ないほど、同じ対象に対して相対的に透視投影に近いことを表しており、元の画像同士による比較よりもさらに、画像同士の差を視覚的に理解できる。実際に、分割数を増やしサブスクリーンを小さくするに伴って、段階的に擬似透視投影が透視投影に近づいていることが見てとれる。

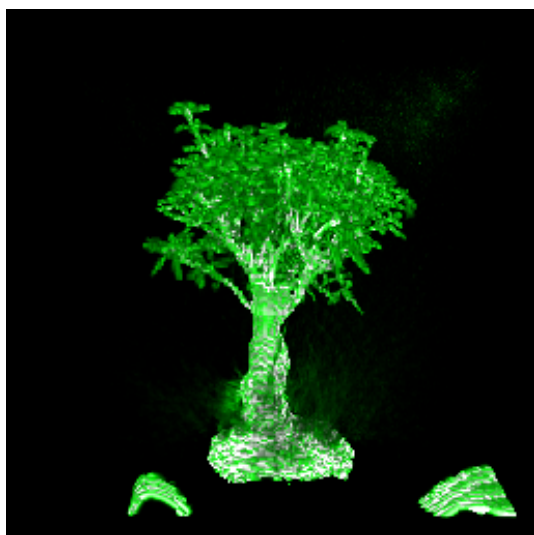
以上二通りのデータ、スクリーンの組についての目で見えた印象による比較から、擬似透視投影の効果はスクリーンが大きいほどに効果が出るのではないかと推測される。また、人間が擬似透視投影と透視投影の区別がつかなくなるのは、サブスクリーンの大きさが $4 \times 4 \sim 8 \times 8$ 程度だと分かる。

数値的考察

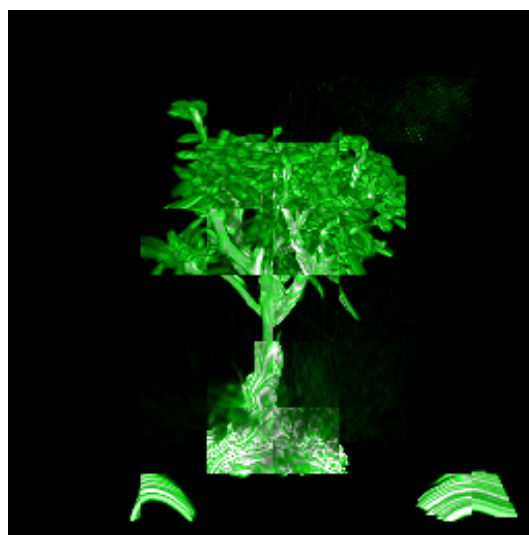
次に、両方のデータ、スクリーンについて数値的な比較を行う。

まず、各画像の画素において、RGB の値を輝度 Y に変換する。次に、透視投影に対する、各分割数における擬似透視投影の輝度の SN 比

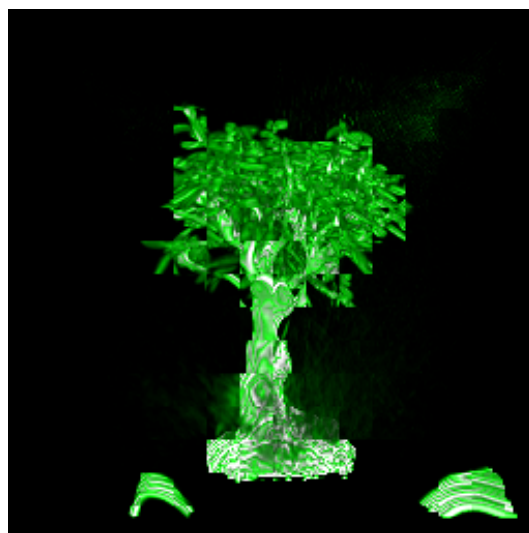
$$X = 10 \times \log \frac{\sum Y_{pers}^2}{\sum (Y_{pseudo} - Y_{pers})^2} \quad [dB]$$



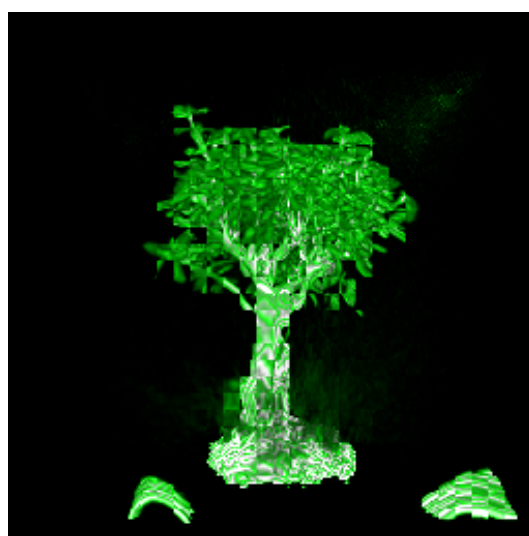
透視投影



N = 8
(s = 32 × 32)



N = 16
(s = 16 × 16)



N = 32
(s = 8 × 8)

Fig. 6 盆栽 (データ : 256 × 256 × 128, スクリーン : 256 × 256)

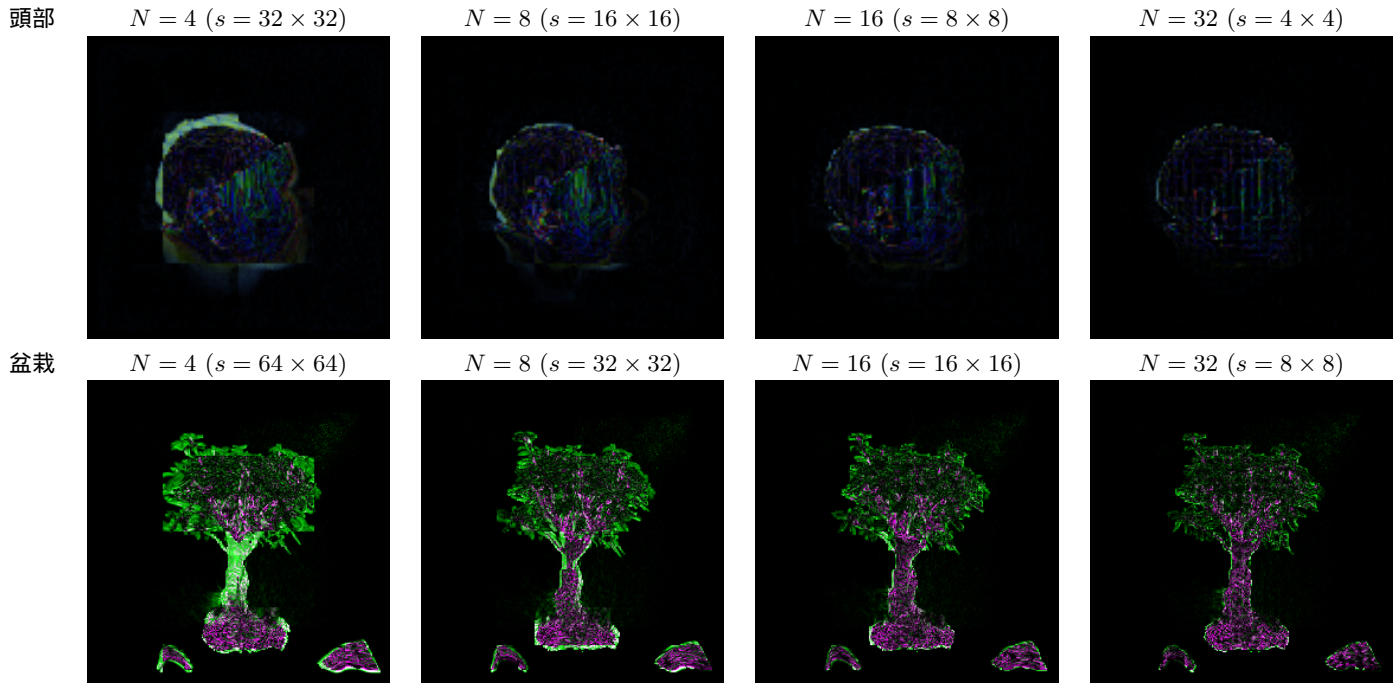


Fig. 7 擬似透視投影と透視投影における RGB 値の差

を求める。Fig 8 に分割数に伴う SN 比の変化を示す。(横軸:分割数の常用対数, 縦軸:SN 比)

どちらの組に対しても, 分割数の増加に伴って類似性が増加しており, 主観的な比較を裏付ける結果となっている。また, このグラフにおいて増加の様子は線形, つまり元の分割数に対しては対数的であることから, ある程度の分割数を越えるとそれ以後に大きな画質の向上はないということが分かる。

ここでもう一度 Fig 7 の盆栽の図を見ると, 目で見た感覚では $N = 16$, $N = 32$ はさほど差がないように思われるが, 実際には着実に透視投影に近付いていることが, 盆栽のグラフが線形に増加していることから分かる。

主観的な観点からの考察と合わせて考えると, 今後の大規模ボリュームレンダリングにおける大きなスクリーンにおいて, 擬似透視投影の効果はより高くなると考えられる。

4. ま と め

本稿では擬似透視投影の提案をし, 主観的, 数値的な評価

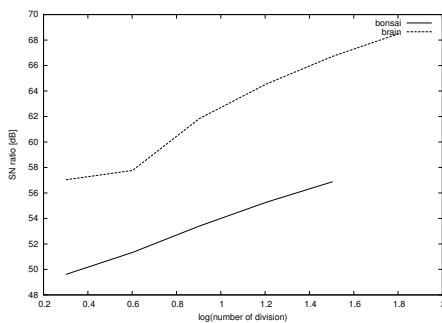


Fig. 8 透視投影画像に対する擬似透視投影画像の SN 比 (盆栽の $N = 64$ での値は未測定)

を行った。その結果, サブスクリーンの大きさとして $4 \times 4 \sim 8 \times 8$ 程度が目安であること, スクリーンが大きくなるにつれて擬似透視投影の効果が向上することを示した。

今後の課題としては, より大きなスクリーンにおける擬似透視投影の効果の検証, このアルゴリズムを用いて, 我々が取り組んでいる並列可視化環境のハードウェア³⁾の実装を進めることが挙げられる。

謝 辞

本研究の一部は文部省科学研究費補助金 (基盤研究 (B)(2) 課題番号 13480083 ならびに 特定領域研究 (C)(2) 「情報学」 課題番号 13224050) による。

また, 本稿で用いたポリウムデータは, 盆栽: Computer Graphics Group, Computer Science Department, University of Erlangen-Nuremberg, Germany. 頭部: The VolPack¹⁾ を利用させて頂いた。

参 考 文 献

- 1) The VolPack Volume Rendering Library, Stanford Computer Graphics Laboratory, 1994
- 2) The VolumePro Real-Time Ray-casting System, Hanspeter Pfister and Jan Hardenbergh and Jim Knittel and Hugh Lauer and Larry Seiler, Siggraph 1999, Computer Graphics Proceedings
- 3) 数値シミュレーション過程の実時間可視化を支援するハードウェア, 原瀬 史靖, 森 真一郎, 富田 真治他, 可視化情報学会第 30 回可視化情報シンポジウム, pp.119-122 (2002)