

# グリッドコンピューティングを用いた ボリュームレンダリング手法の開発

---

広島大学 大学院 工学研究科

岡部 憲史

檜垣 徹

金田 和文

玉木 徹

# 背景

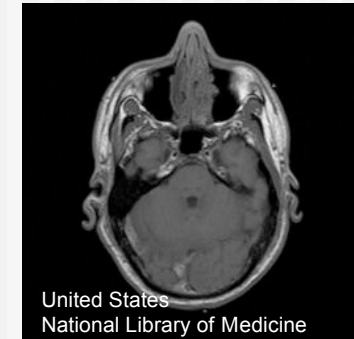
CTやMRIから出力された断面画像

- ・・・直観的な観察が困難
- 高精度な3次元表示の要求

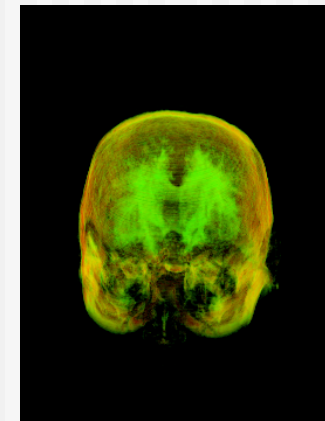
ボリューム  
レンダリング

- ・3次元表示により直観的・内部まで観察が可能
- ・データ量増大により高計算コスト

医療機関では計算機を多数導入  
空いている計算資源を有効利用  
グリッドコンピューティング



United States  
National Library of Medicine  
頭部断面画像



頭部MRIデータ

# 提案手法の要点

## ■ レンダリング手法の選択

レイキャスティング法 [Levoy '88]

・・・ 全てのセル値が画像に反映されるとは限らない

スプラッティング法 [Westover '90]

・・・ 全てのセル値が画像に反映され **高精度**

## ■ 品質係数の導入

**品質係数**に基づく閾値によりセルをグループ化

**ユーザーの求める品質の画像を高速に生成**

## ■ 可視性の優先順位を考慮した分割レンダリング・合成

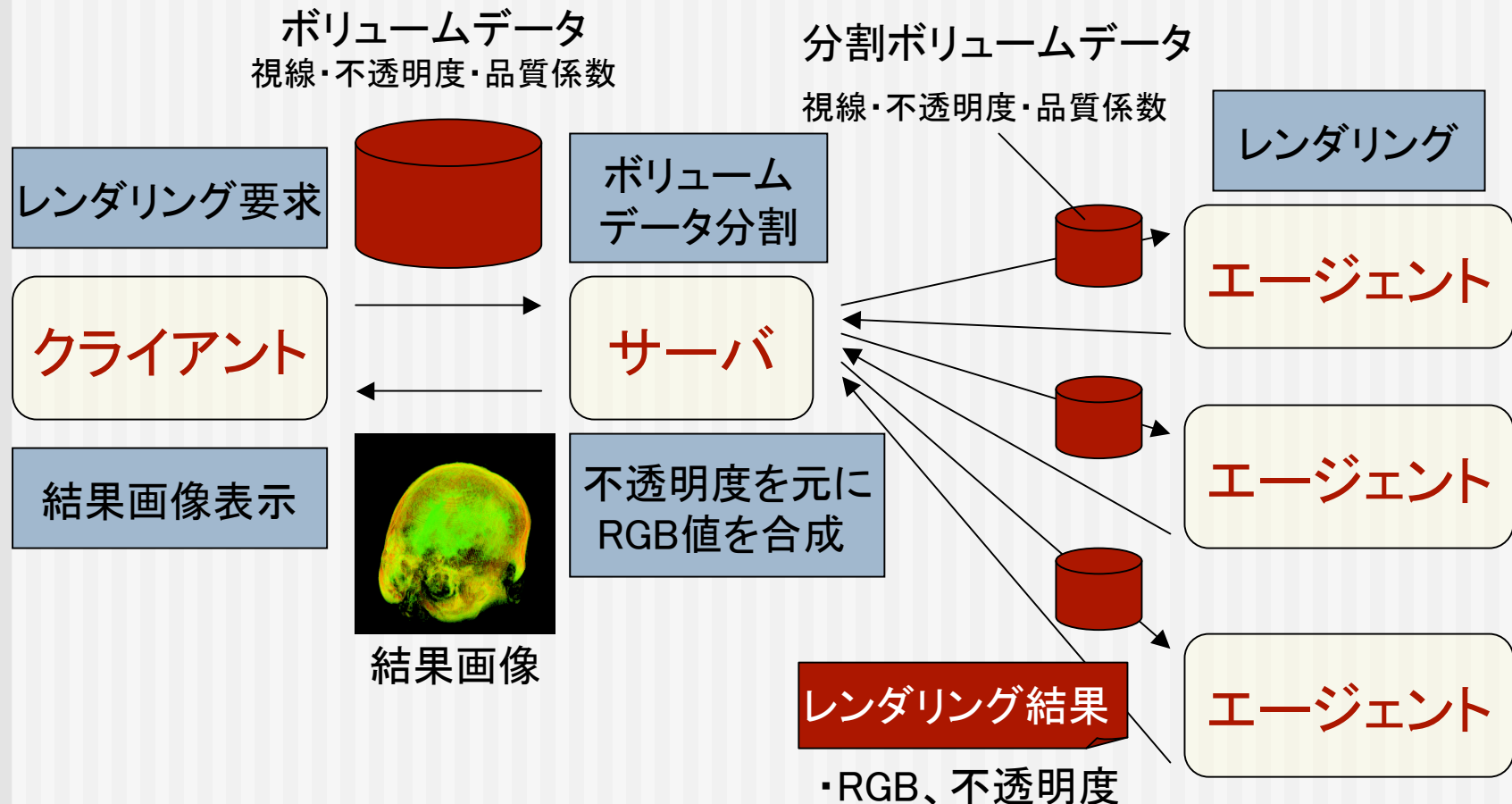
**オブスタクルフラグ**を使用し **可視性の優先順位**を決定

**可視性の優先順位**を考慮した順序で

ボリュームデータを分割・個別にレンダリング

不透明度をもとにそれぞれのレンダリング結果を合成

# 全体の構成・流れ



⋮

# 提案手法の要点

## ■ レンダリング手法の選択

レイキャスティング法[Levoy '88]

・・・ 全てのセル値が画像に反映されるとは限らない

スプラッチング法[Westover '90]

・・・ 全てのセル値が画像に反映され**高精度**

## ■ 品質係数の導入

**品質係数**に基づく閾値によりセルをグループ化

**ユーザーの求める品質の画像を高速に生成**

## ■ 可視性の優先順位を考慮した分割レンダリング・合成

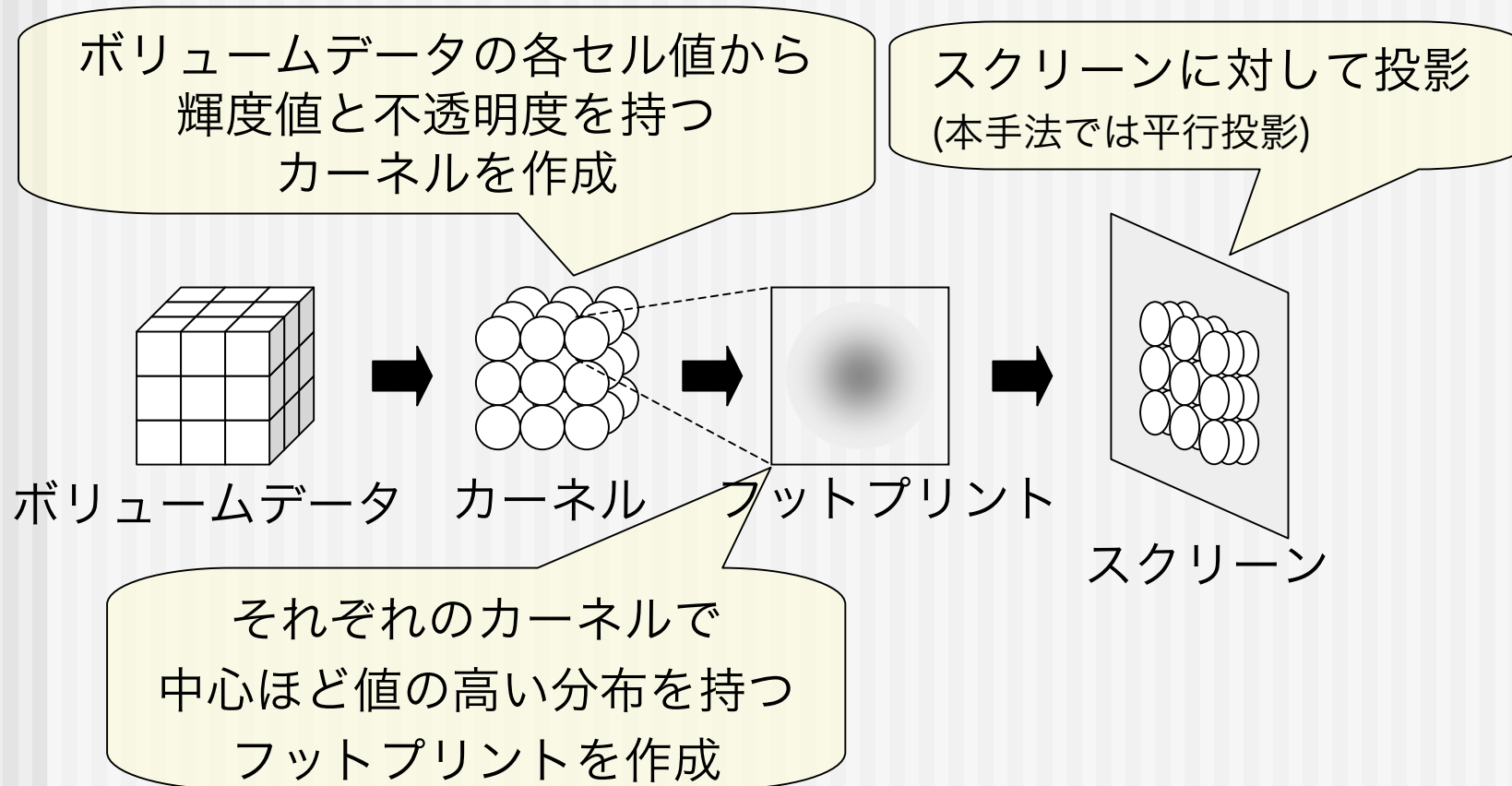
**オブスタクルフラグ**を使用し**可視性の優先順位**を決定

**可視性の優先順位**を考慮した順序で

ボリュームデータを分割・個別にレンダリング

不透明度をもとにそれぞれのレンダリング結果を合成

# スプラッティング法 [Westover '90]



全てのボリュームデータをカーネルとして投影  
⇒ **高精度な表示**

# 提案手法の要点

## ■ レンダリング手法の選択

レイキャスティング法 [Levoy '88]

・・・ 全てのセル値が画像に反映されるとは限らない

スプラッティング法 [Westover '90]

・・・ 全てのセル値が画像に反映され **高精度**

## ■ 品質係数の導入

**品質係数**に基づく閾値によりセルをグループ化

**ユーザーの求める品質の画像を高速に生成**

## ■ 可視性の優先順位を考慮した分割レンダリング・合成

**オブスタクルフラグ**を使用し **可視性の優先順位**を決定

**可視性の優先順位**を考慮した順序で

ボリュームデータを分割・個別にレンダリング

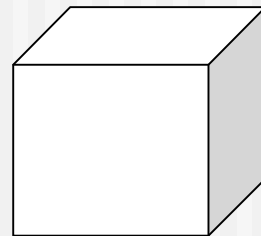
不透明度をもとにそれぞれのレンダリング結果を合成

# 品質係数

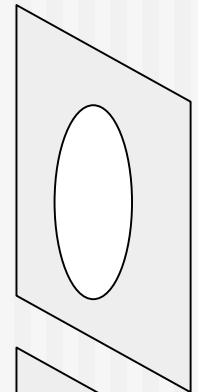
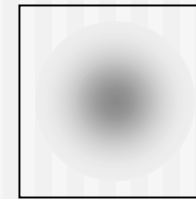
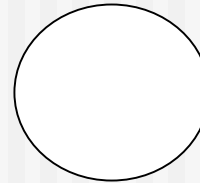
差分の最大値  $< t$

**グループ化可能**

グループ化[余傳 '03]

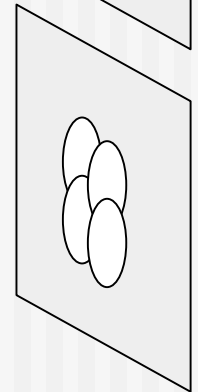
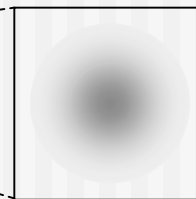
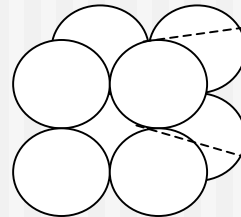
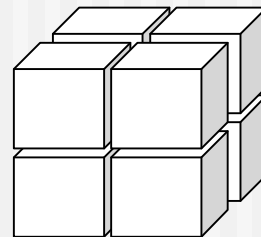


グループセル



差分の最大値  $\geq t$

**グループ化不可**



グループ化閾値  $t$  ... 品質係数、セルの最大値・最小値から算出

- ・ **ユーザにより** 1 (高品質) ~ 0 (低品質) で与えられる
- ・ 生成画像の精度は **品質係数に依存**



# 提案手法の要点

## ■ レンダリング手法の選択

レイキャスティング法 [Levoy '88]

・・・ 全てのセル値が画像に反映されるとは限らない

スプラッティング法 [Westover '90]

・・・ 全てのセル値が画像に反映され **高精度**

## ■ 品質係数の導入

**品質係数**に基づく閾値によりセルをグループ化

**ユーザーの求める品質の画像を高速に生成**

## ■ 可視性の優先順位を考慮した分割レンダリング・合成

**オブスタクルフラグ**を使用し **可視性の優先順位**を決定

**可視性の優先順位**を考慮した順序で

ボリュームデータを分割・個別にレンダリング

不透明度をもとにそれぞれのレンダリング結果を合成

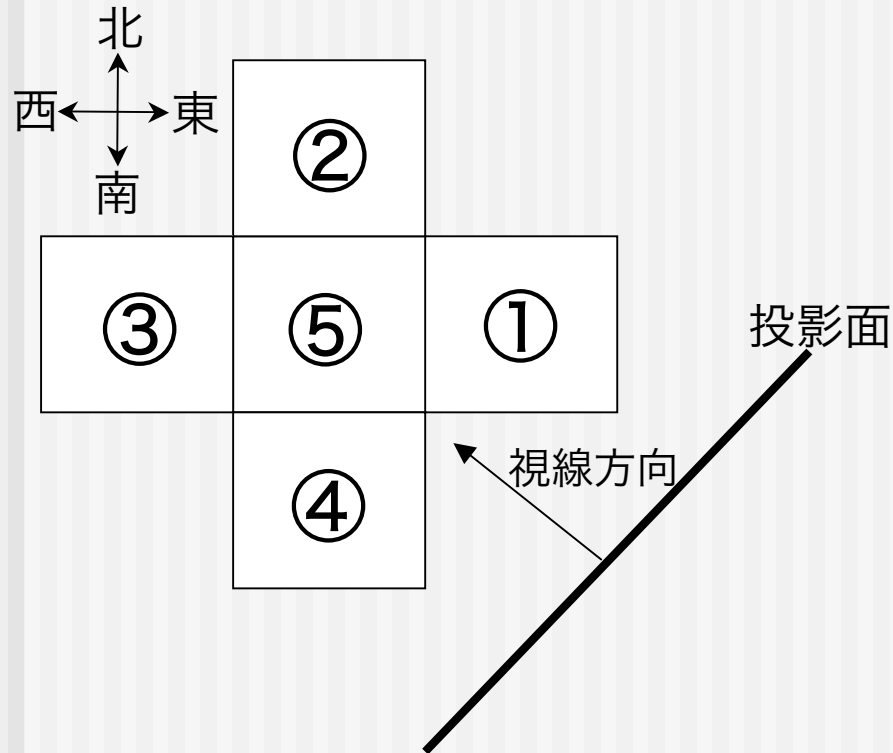
# 可視性の優先順位

- 投影面に近いセルほど可視性が高い
  - レンダリングには**順序**がある
  - 分割してレンダリングした結果を合成する場合にも**順序**がある
- タスクの割り振りと同時に合成を行う
  - タスクの結果が割り振った順に**戻ってこない可能性**
  - **合成可能**かどうか**確認**する必要性  
(合成できない場合は**メモリに格納**)

合成できる**可能性を高める**必要がある

**オブスタクルフラグ**を作成・**更新**することで  
効率的に割り振りと合成を行う

# オブスタクルフラグ



※説明の都合上、2次元で示す

⑤の合成の障害となるのは

①と④

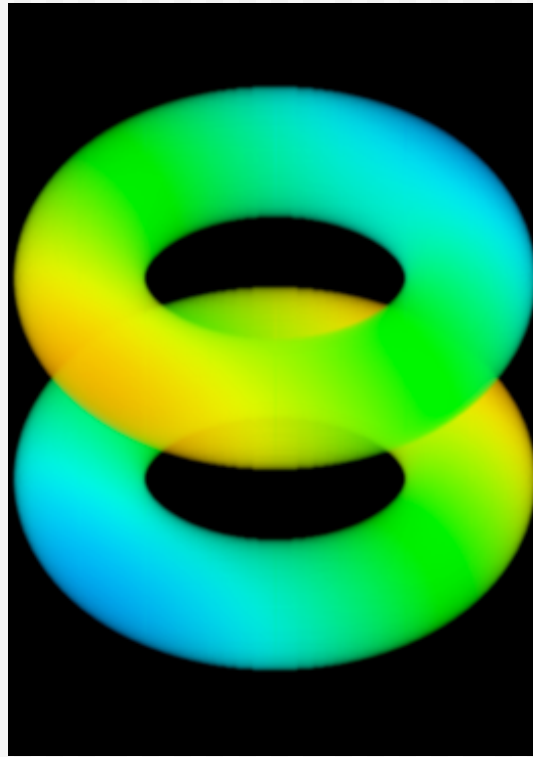
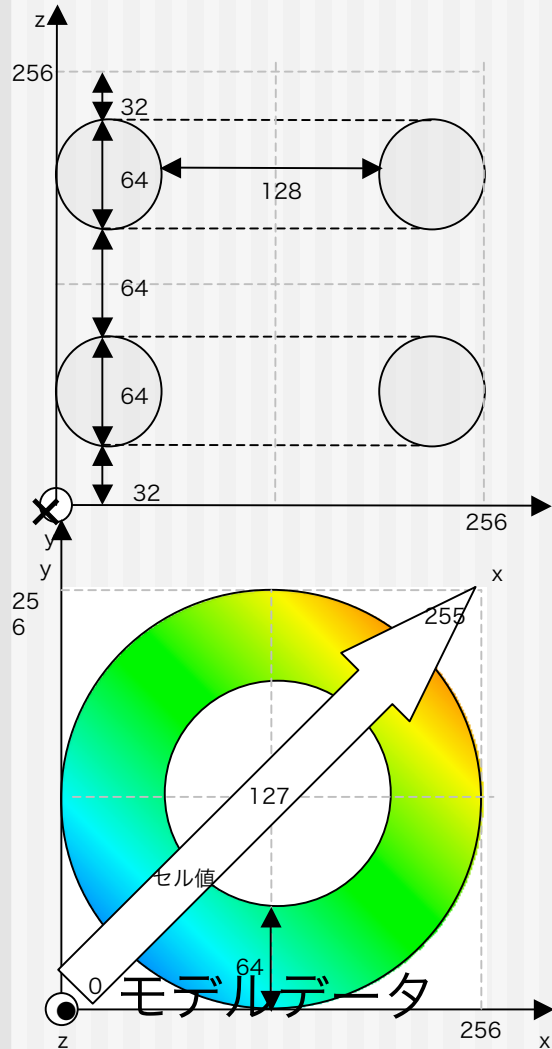
よって

方向	北	東	南	西
フラグ	0	1	1	0

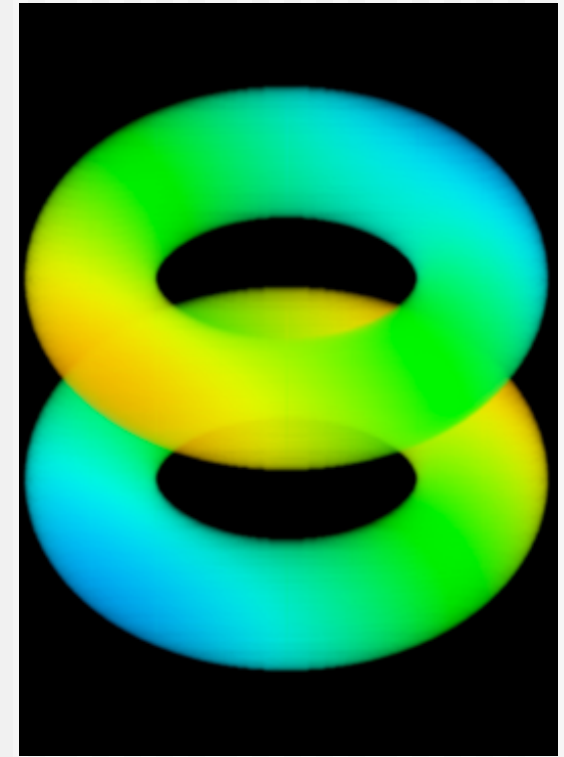
(障害があれば1、なければ0)

- 各分割ボリュームデータに**1つずつ用意**される
- 合成され障害が取り除かれると**更新**される
- 全てのフラグが**0**になったとき**合成可能**となる

# 結果 (品質係数のみ適用)



A

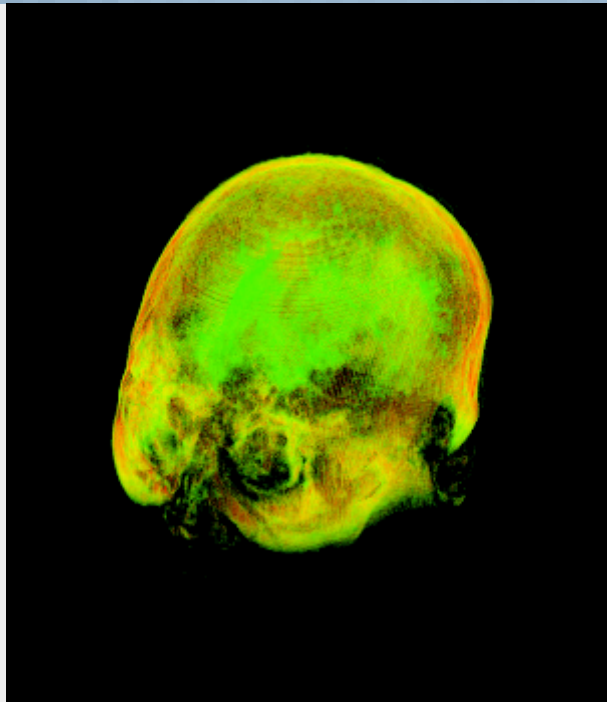


B

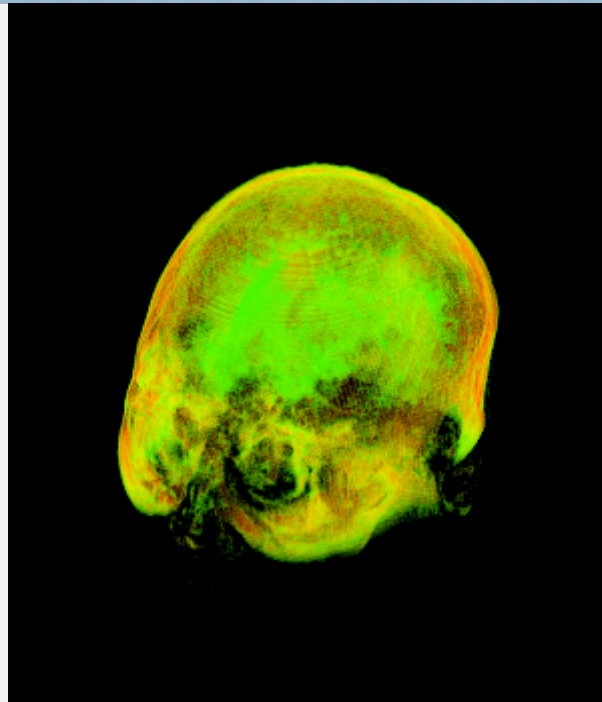
	A	B
品質係数	1.00(非グループ化)	0.99
不透明度	0.01	0.01
レンダリング時間	13.041sec	6.041sec
スクリーンサイズ	264×368	264×368

使用計算機 CPU : AMD Athlon64 2800+ Memory : 1.5GB

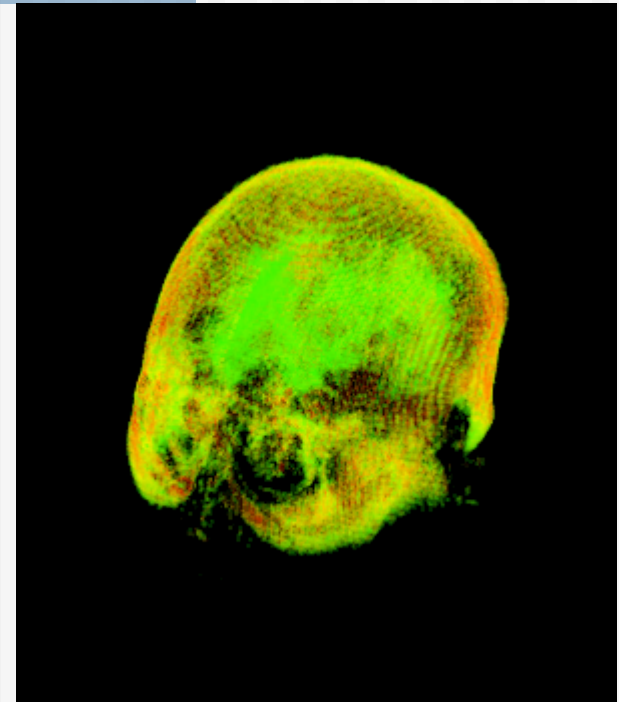
# MRIデータへの適応結果 (品質係数のみ適用)



A



B



C

	A	B	C
品質係数	1.00(非グループ化)	0.99	0.50
不透明度	0.1	0.1	0.1
レンダリング時間	8.047sec	7.938sec	3.172sec
スクリーンサイズ	300×350	300×350	300×350

使用計算機 CPU : Xeon3.6GHz×2 Memory : 3GB

# まとめ

---

グリッドコンピューティングを用いた  
ボリュームレンダリングの高速化のための

- システム概要、可視性の優先順位、品質係数について提案
- 品質係数により要求された品質でのレンダリングの高速化が行えることを確認

今後の課題

- グリッドを使用した実装
- タスク割り振りの最適化
- エージェント数に対する適当な分割数の選定