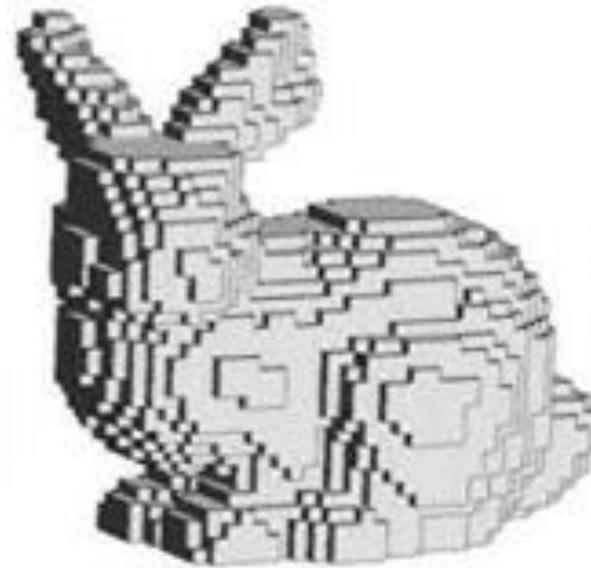


# 3D-SRCGANを用いた 低解像度三次元ボクセルモデルから サーフェイスモデルへの変換

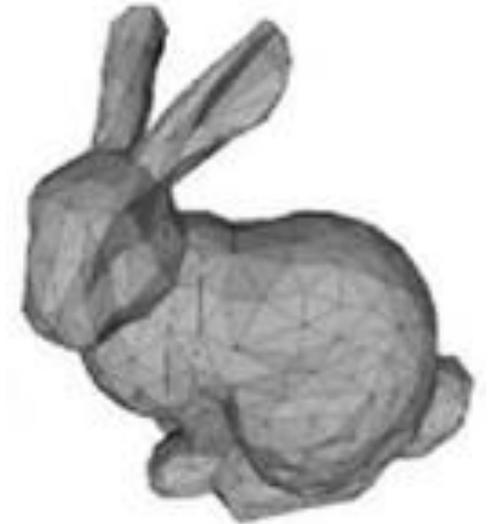
宮崎大学大学院 工学研究科 工学専攻  
機械・情報系コース 情報システム工学分野  
T2303736 ADAM MANATO MAEDA BIN AMIZAN  
指導教員 椋木 雅之 教授

# 研究背景

- ボクセル表現
  - 空間を均一な立方体グリッド（ボクセル）に分割し、物体の内外を0/1で表す
- サーフェイス表現
  - 物体の表面を三角形などの面で表現



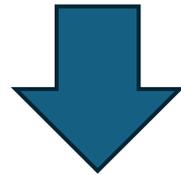
ボクセルモデル



サーフェイスモデル

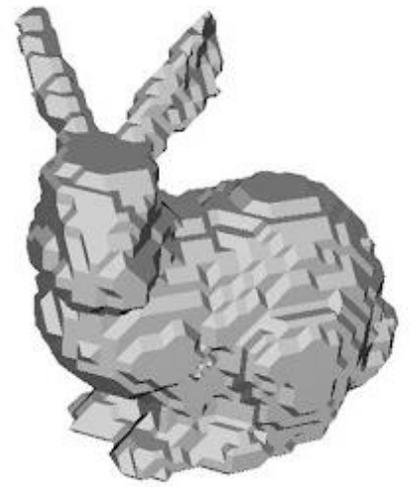
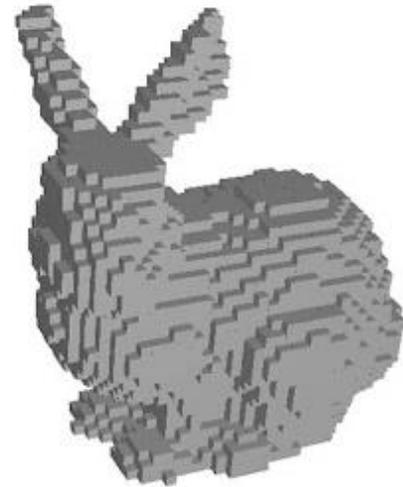
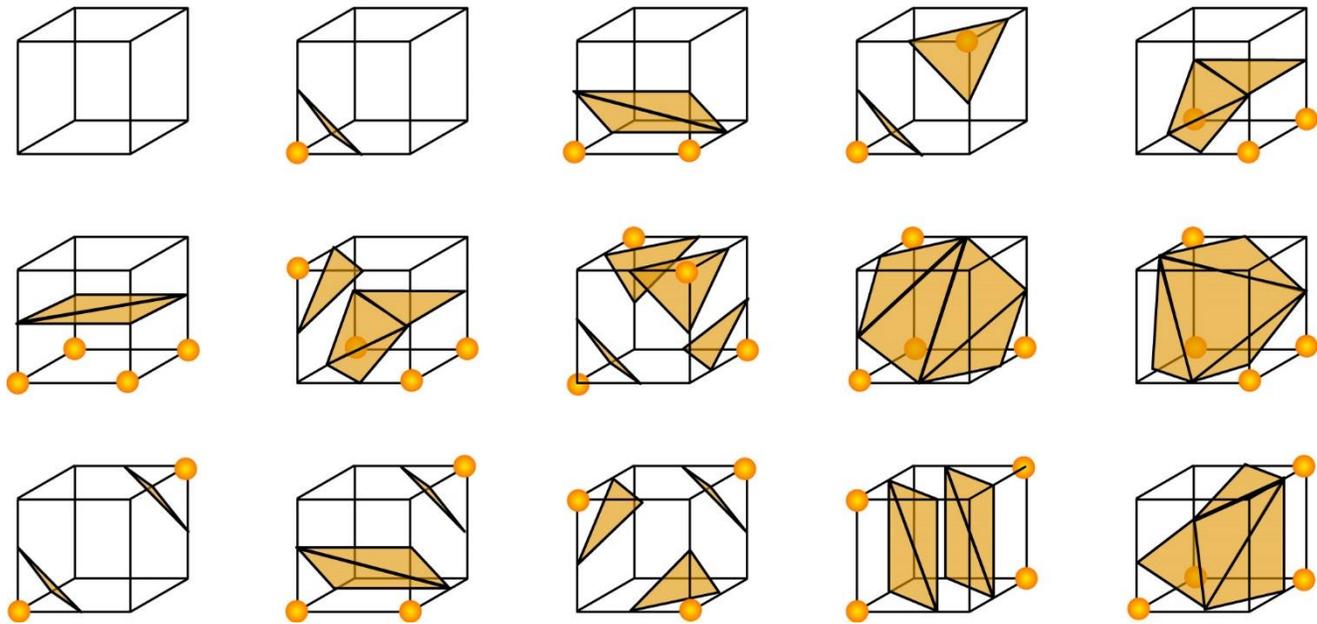
# 目的

- ボクセル表現
  - 利点：積み重ねることで、比較的容易に三次元モデルを作成
  - 欠点：必要なメモリ量が多く、処理時間が長くなる
- サーフェス表現
  - 利点：データ量を抑えやすく、高精細でリアルな外見を再現できる
  - 欠点：CADなどの専用ツールの操作や効率的なワークフローの習得が必要



粗い三次元ボクセルモデルからサーフェスモデルを生成

# 従来手法 (Marching Cubes法)

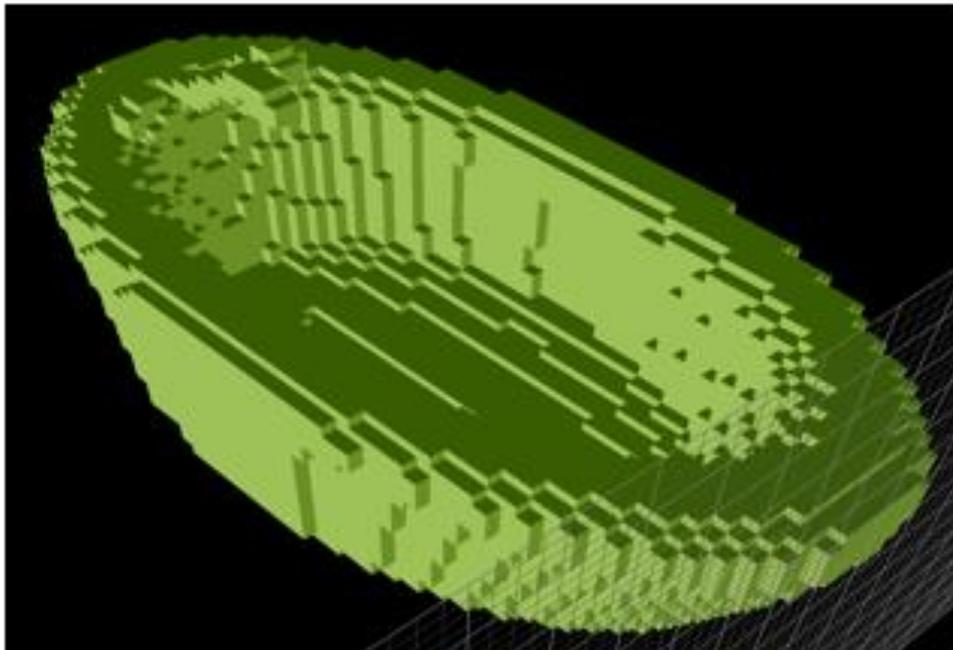


- ボクセルから面を生成するのに一般的に使われる
- ボクセルの接続関係を用いて面を生成

# Marching Cubes法の問題点

細かいボクセルモデルからの変換  
⇒サーフェイスマデルは細くなる

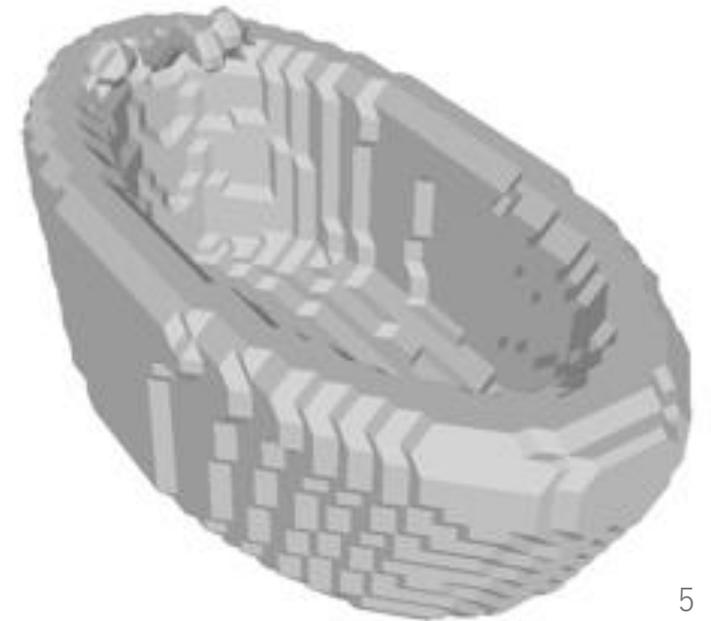
入力：解像度64の三次元ボクセルモデル



サーフェイス生成



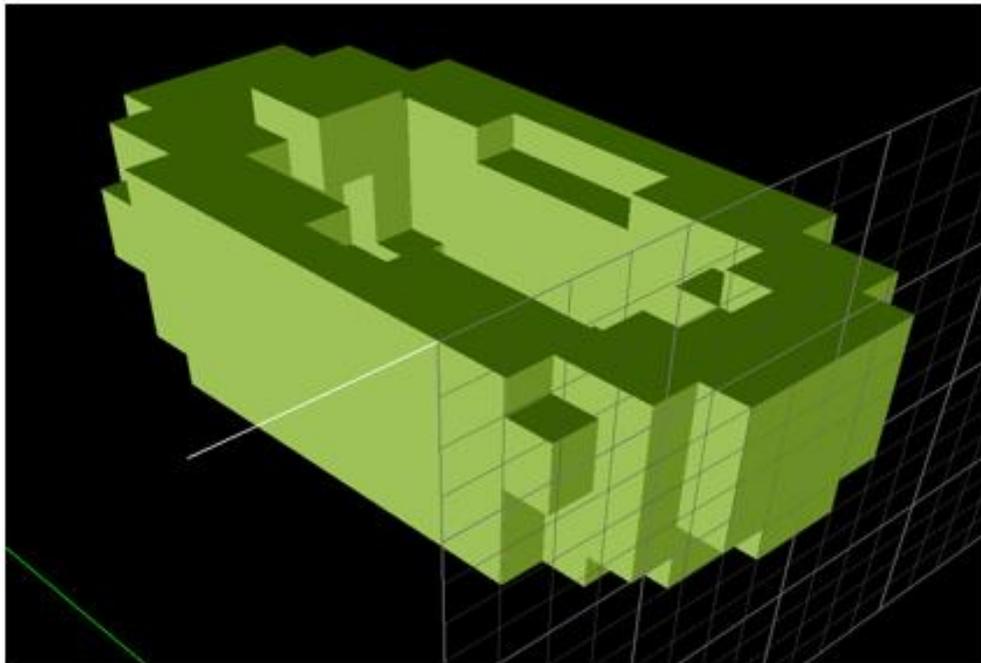
出力：三次元サーフェイスマデル



# Marching Cubes法の問題点

粗いボクセルモデルからの変換  
⇒サーフェイスモデルも粗くなる

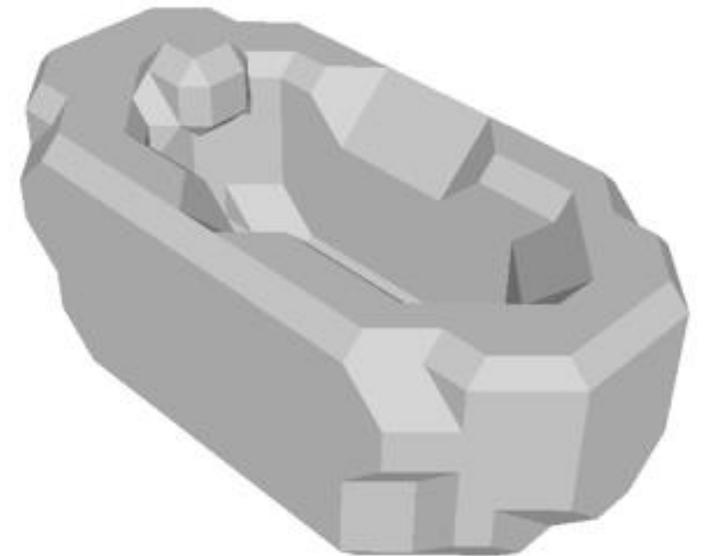
入力：解像度16の三次元ボクセルモデル



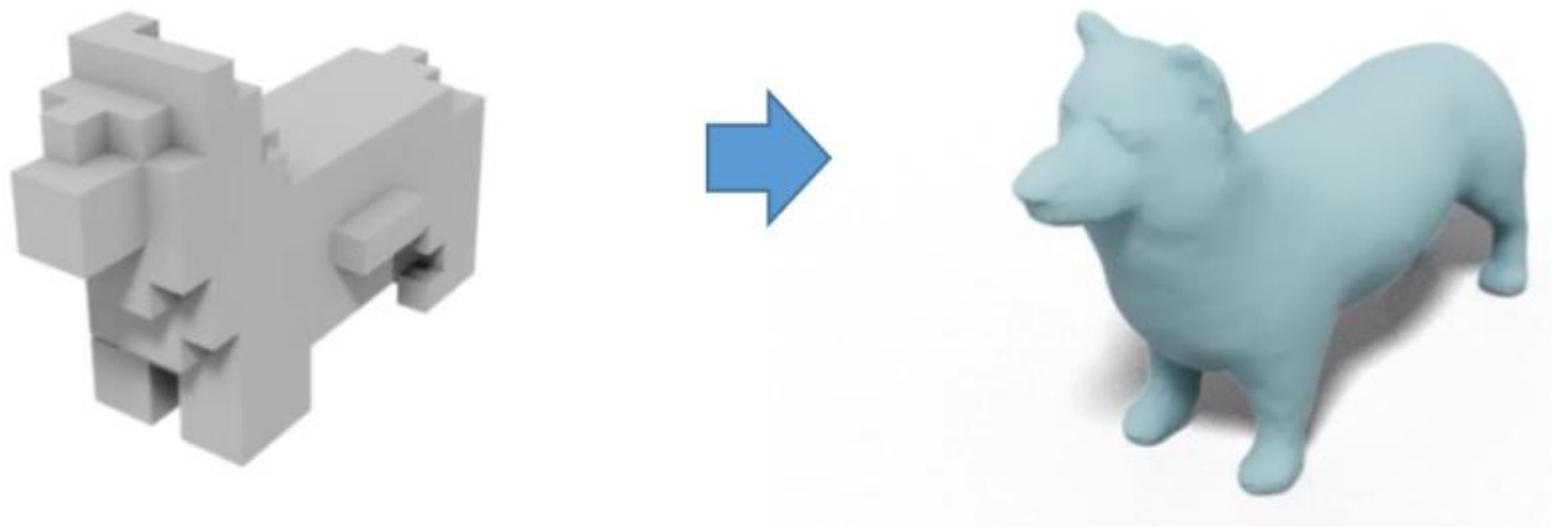
サーフェイス生成



出力：三次元サーフェイスモデル



# 従来手法 (Deep Marching Tetrahedra : DM Tet法)

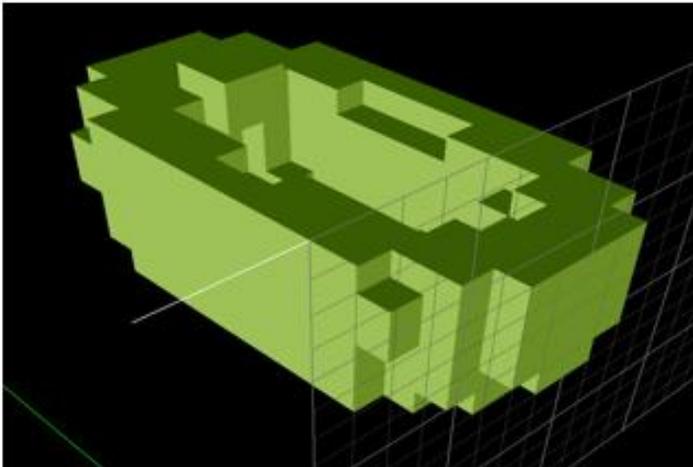


深層学習を用いて粗いボクセルから高精細なサーフェイスを生成

# DMTet法の問題点

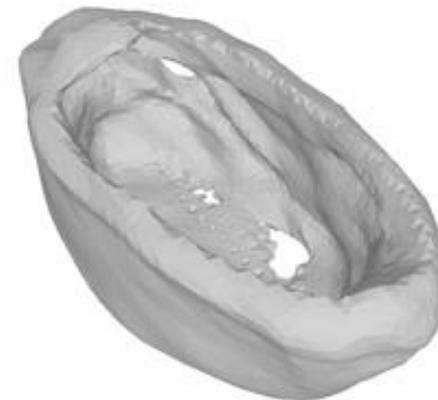
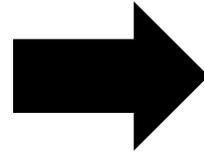
- サンプルコードが公開されているものの、完全な実装にはなっていない
- 生成されたサーフェイスに多数の穴が存在し、サーフェイスが滑らかではなく、凹凸が目立つ

入力：低解像度三次元ボクセルモデル



入力：三次元サーフェイスモデル

サーフェイス生成



# 提案手法

超解像を用いて低解像度三次元ボクセルモデルからサーフェイスモデルに変換

1. 3D-SRCGANによる三次元ボクセルモデルの超解像
2. Marching Cubes法によるサーフェイス生成
3. Laplacian Smoothingによるサーフェイスモデルの平滑化

- 超解像処理の導入

低解像度のデータから高解像度のデータを生成・補完する技術

⇒ Marching Cubes法に細かいボクセルモデルを与えられる

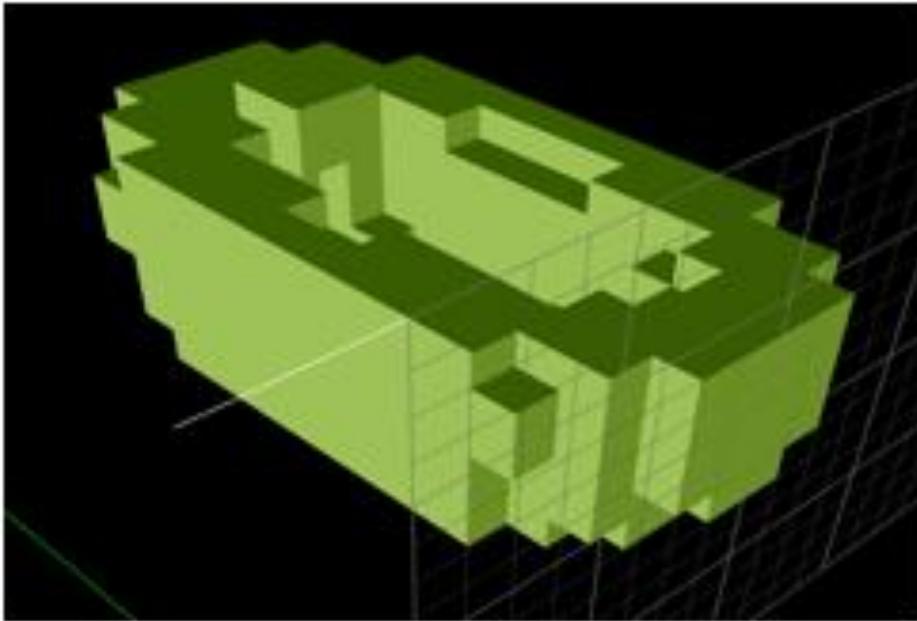
- 平滑化処理を実施

形状を滑らかにする手法

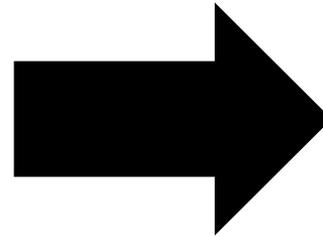
⇒ 生成結果の凹凸を軽減できる

# 1. 3D-SRCGANによる三次元ボクセルモデルの超解像

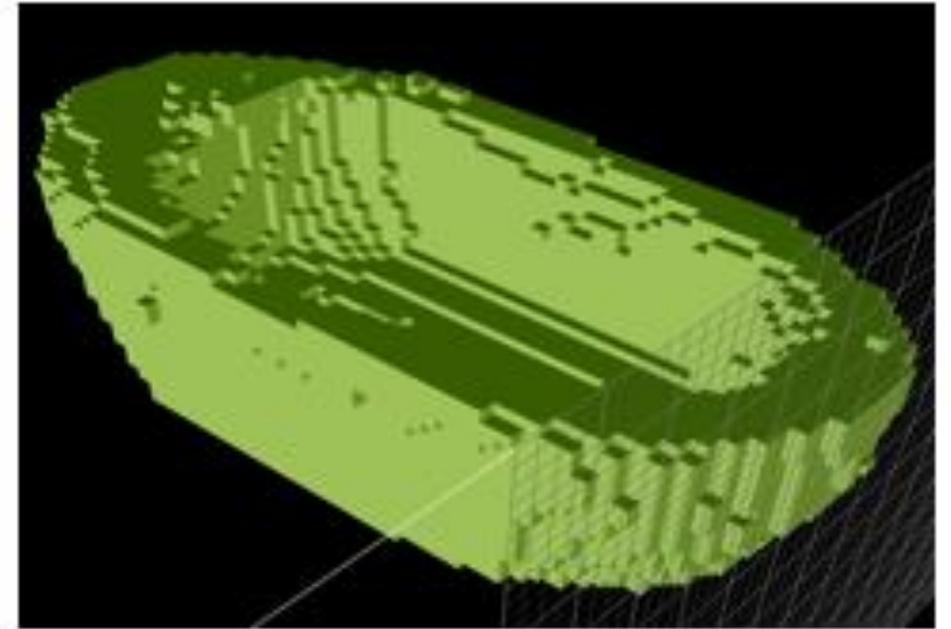
入力：低解像度三次元モデル



超解像

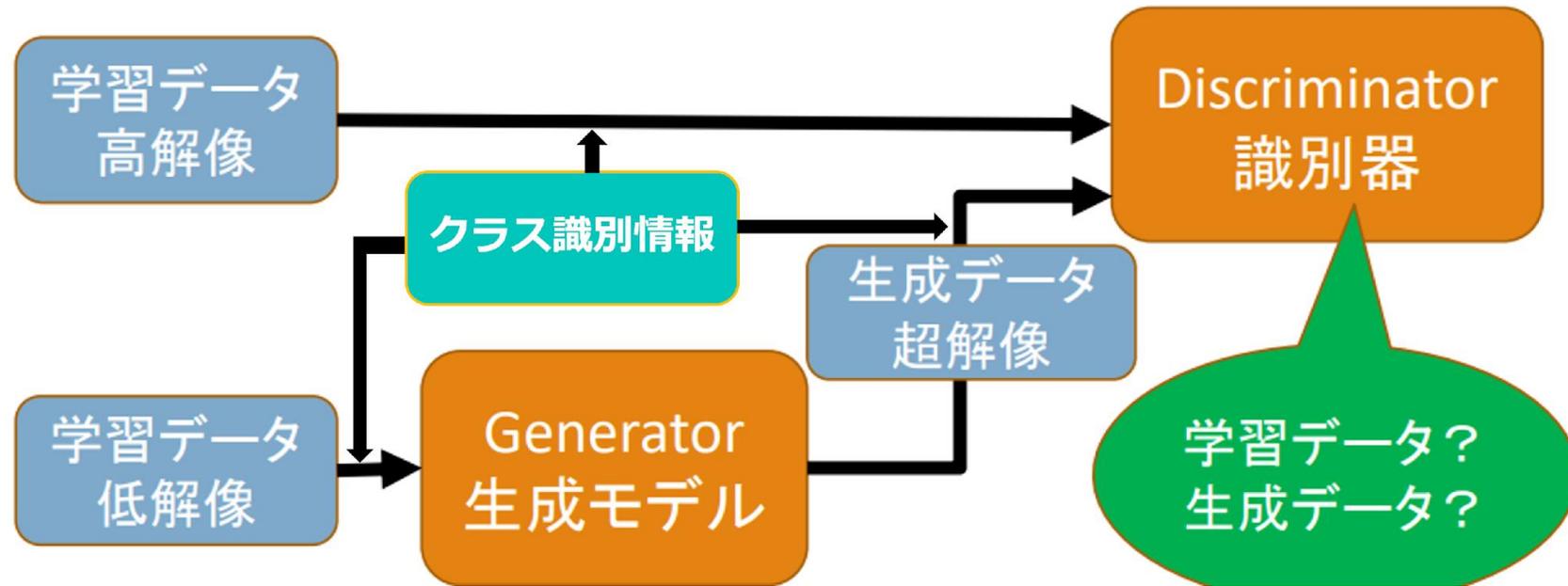


出力：超解像三次元モデル



粗い三次元モデル→細かな三次元モデル

# 3D-SRCGAN

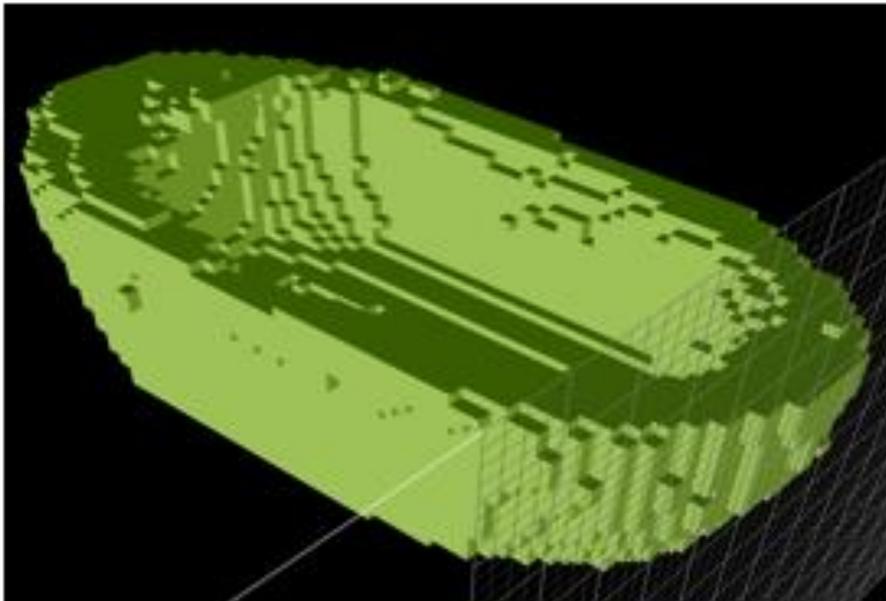


- 三次元ボクセルモデルをGANを用いて超解像
- GeneratorとDiscriminatorの入力にクラス識別情報を付加

## 2. Marching Cubes法によるサーフェイス生成

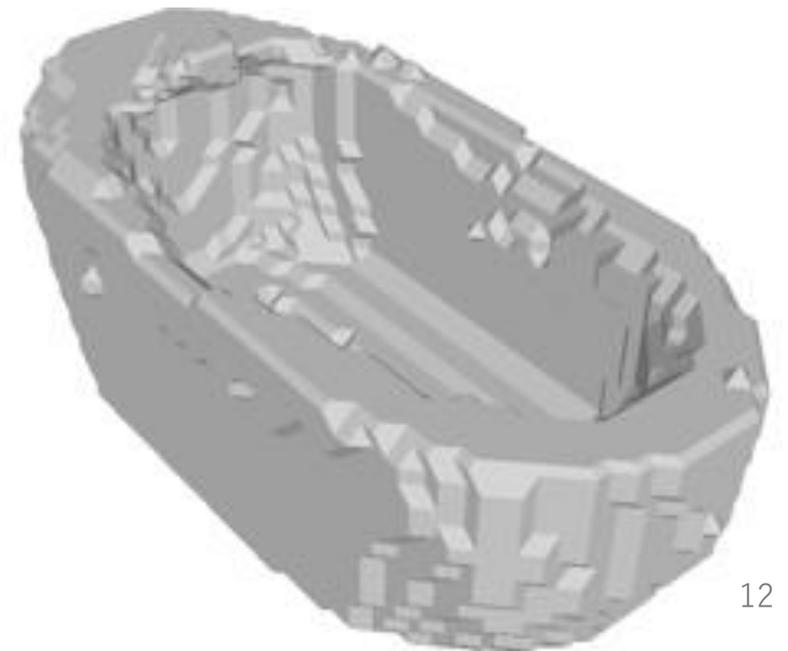
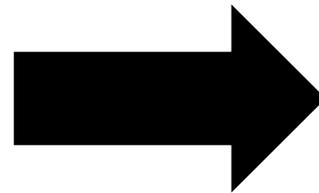
入力は $64^3$ の超解像三次元モデルで、出力はサーフェイス表現の三次元モデル

入力：超解像三次元モデル



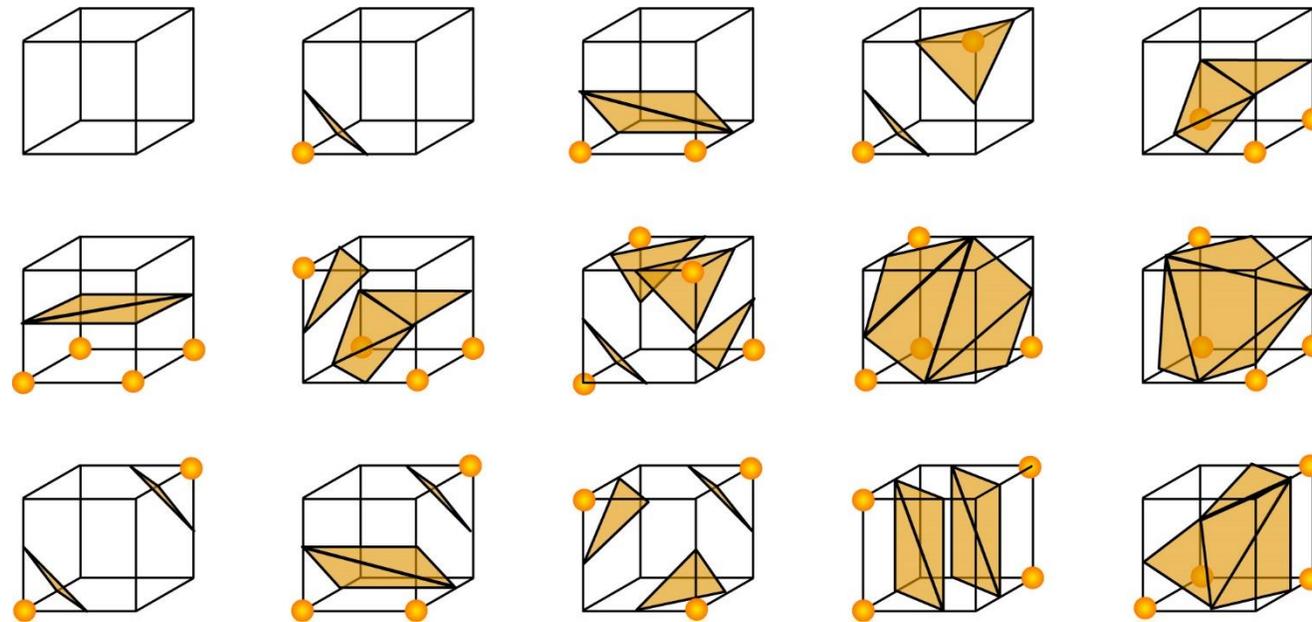
出力：三次元サーフェイスモデル

サーフェイス生成



# Marching Cubes法

- ボリウムデータ→等値面のサーフェイス表現を生成

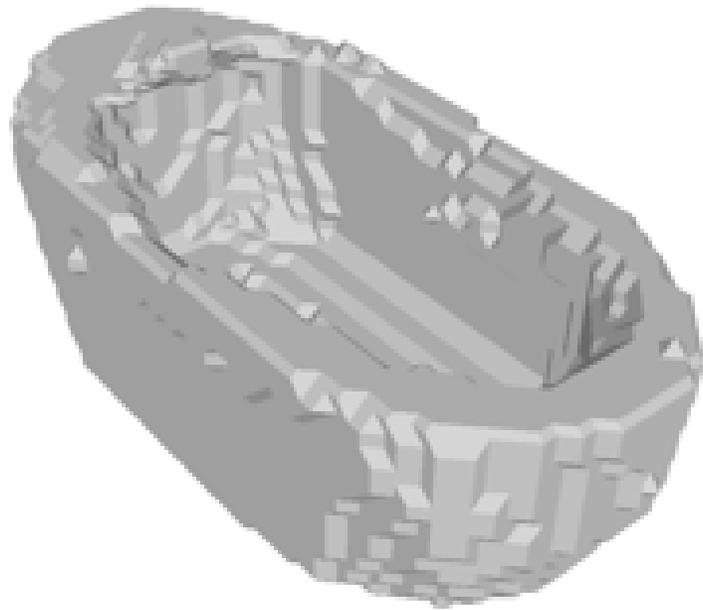


- ボクセル表現→等値面の値を0.5として、サーフェイス生成

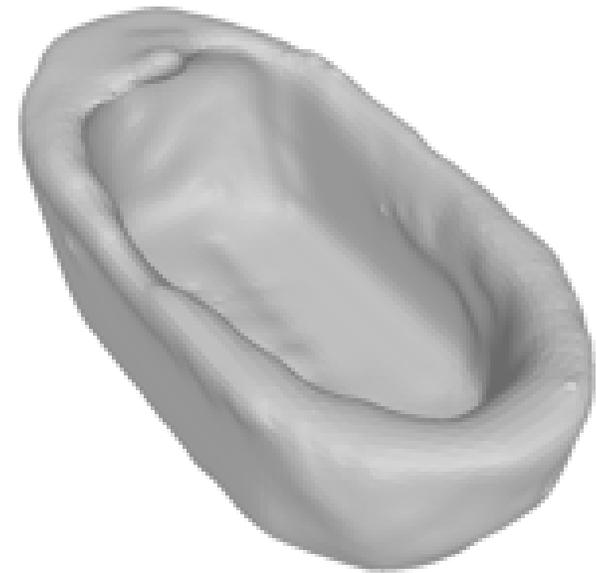
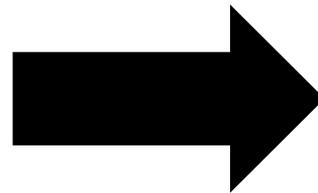
### 3. Laplacian Smoothingによる サーフェイスモデルの平滑化

入力：三次元サーフェイスモデル

出力：平滑化処理後の三次元サーフェイスモデル



平滑化



より滑らかなサーフェイスモデルを生成

# Laplacian Smoothing

- ポリゴンメッシュの形状を滑らかにするための手法
- 各頂点の位置を近傍の頂点の平均位置に基づいて更新する

$$\bar{x}_i = \frac{1}{N} \sum_{j=1}^N \bar{x}_j$$

$N$ はノード $i$ に隣接する頂点の数

$\bar{x}_j$ は $j$ 番目の隣接頂点の位置

$\bar{x}_i$ はノード $i$ の新しい位置

# 実験

提案手法によるサーフェイス生成の精度評価

比較手法

- 提案手法
- $16^3$ 手法：低解像度（解像度 $16^3$ ）の三次元ボクセルモデルに対してMarching Cubes法を直接適用
- DMTet手法：低解像度の三次元ボクセルモデルに対して追加実装したDMTet法を適用
- $64^3$ 手法：高解像度（解像度 $64^3$ ）の三次元ボクセルモデルに対してMarching Cubes法を適用

# DMTet法の追加実装

- 論文の記載
  - 損失関数：5要素
  - GANを使用
- サンプルコード
  - 損失関数：2要素
  - GANは未使用
- 追加実装
  - 損失関数：3要素
  - GANを導入

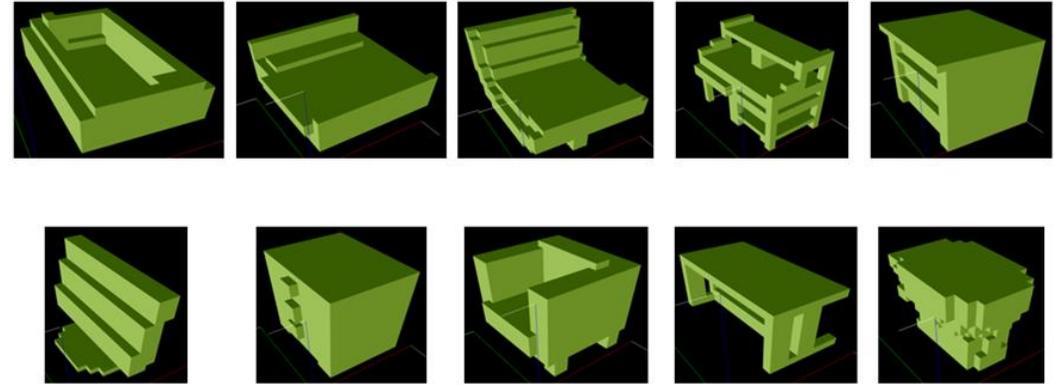
# 実験

提案手法によるサーフェイス生成の精度評価

比較手法

- 提案手法
- $16^3$ 手法：低解像度（解像度 $16^3$ ）の三次元ボクセルモデルに対してMarching Cubes法を直接適用
- DMTet手法：低解像度の三次元ボクセルモデルに対して追加実装したDMTet法を適用
- $64^3$ 手法：高解像度（解像度 $64^3$ ）の三次元ボクセルモデルに対してMarching Cubes法を適用

# 使用したデータセット



## ModelNet10

- 10種類の物体のサーフェイスモデル
- 学習データ、テストデータに分かれている

## ボクセルモデルの作成

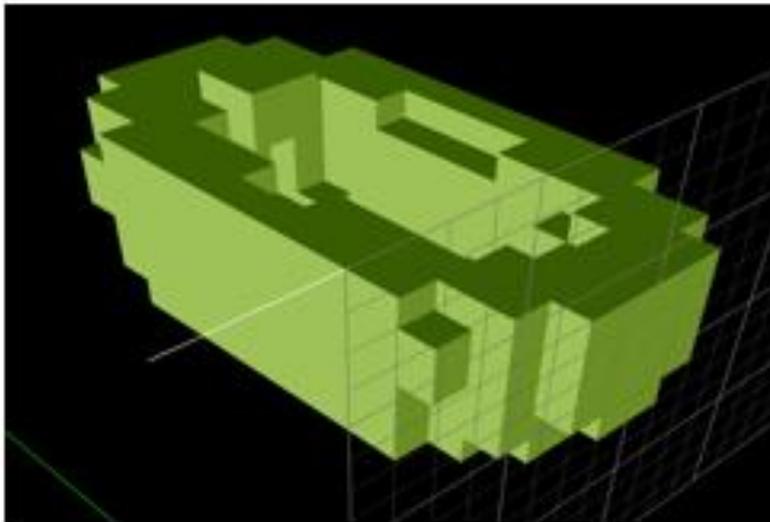
- $16^3$ と $64^3$ のボクセルモデルのペアに変換
- 学習データで学習
- テストデータに各手法を適用

# 実験設定

## 3D-SRCGANの学習

- ModelNet10の学習データ
- バッチサイズを30、Epoch数を50

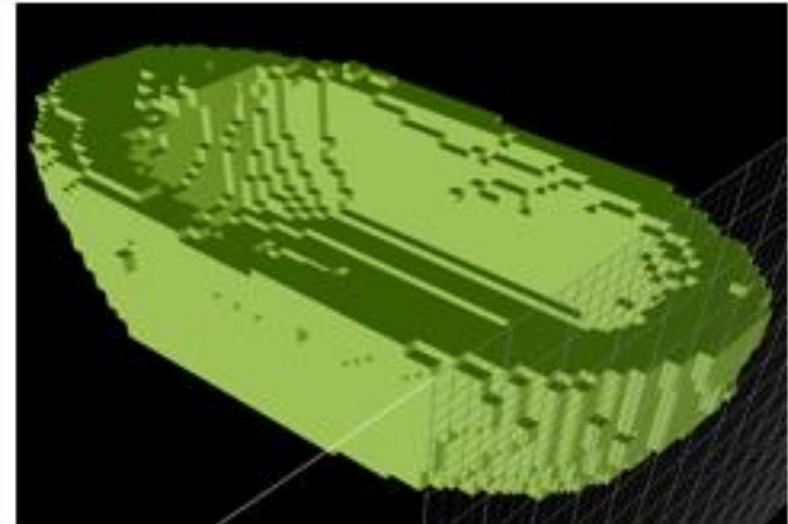
入力：低解像度三次元モデル



超解像



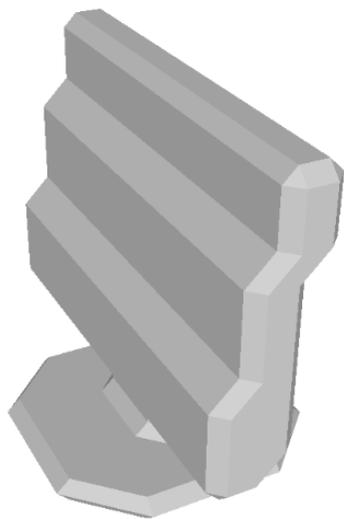
出力：超解像三次元モデル



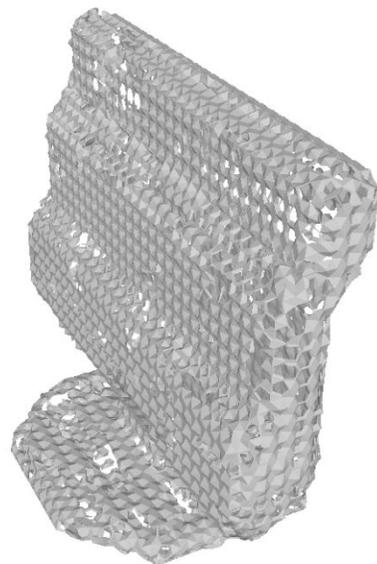
# 実験結果 (主観評価)



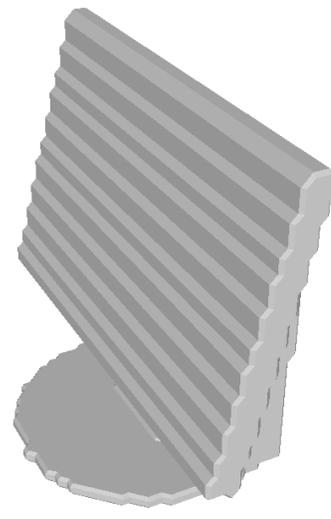
16<sup>3</sup>手法



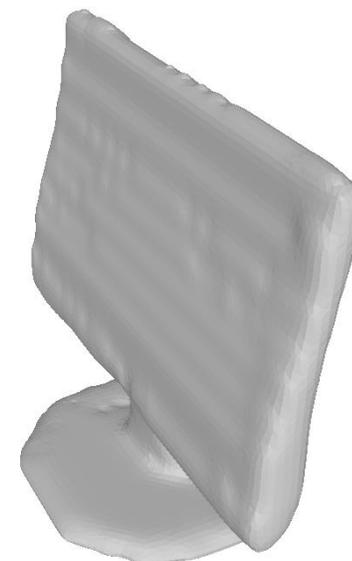
DMTet手法



64<sup>3</sup>手法



提案手法

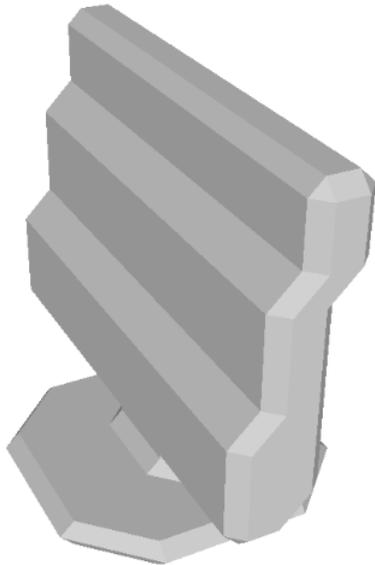


# 実験結果 (主観評価)

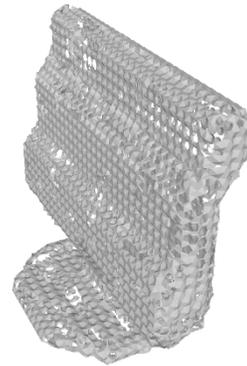
GT



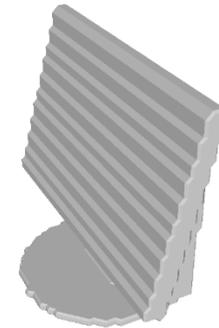
16<sup>3</sup>手法



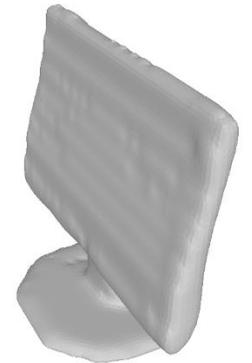
DMTet手法



64<sup>3</sup>手法



提案手法



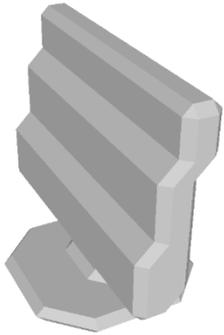
16<sup>3</sup>手法：ボクセルサイズが大きいため、凹凸が目立つ

# 実験結果 (主観評価)

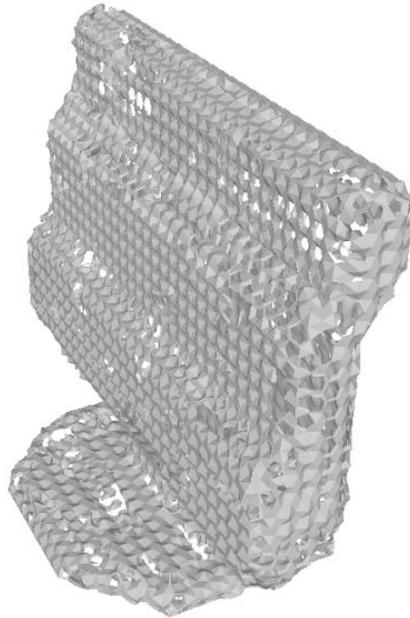
GT



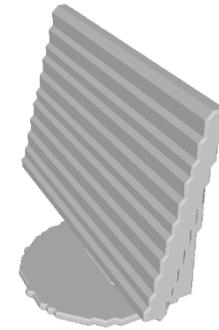
16<sup>3</sup>手法



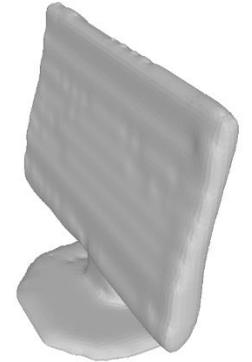
DMTet手法



64<sup>3</sup>手法

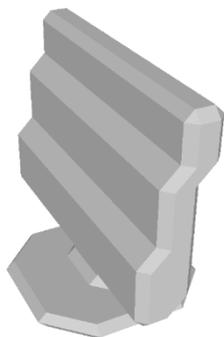


提案手法

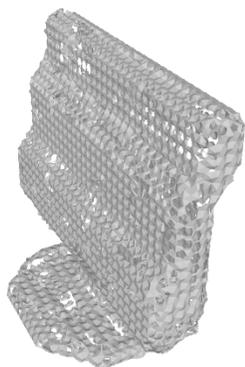
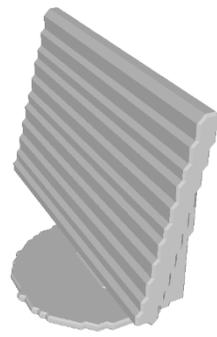


DMTet手法：穴が多数存在して、サーフェイスが滑らかではない

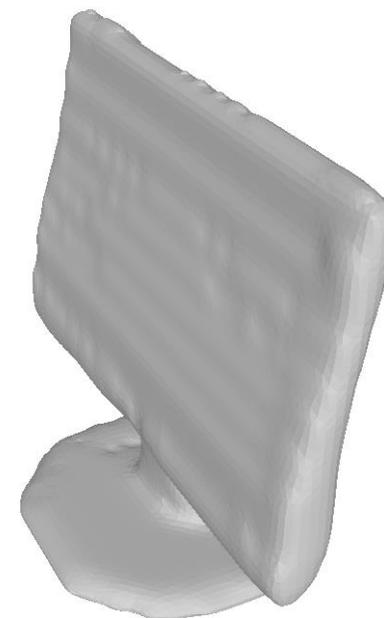
# 実験結果 (主観評価)

16<sup>3</sup>手法

DMTet手法

64<sup>3</sup>手法

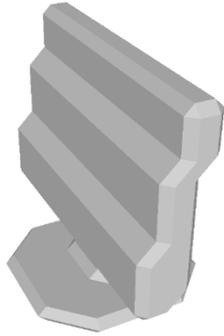
提案手法



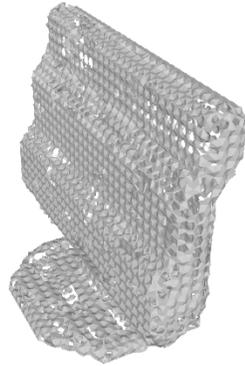
提案手法：平滑化処理により滑らかな面が生成されている

# 実験結果 (主観評価)

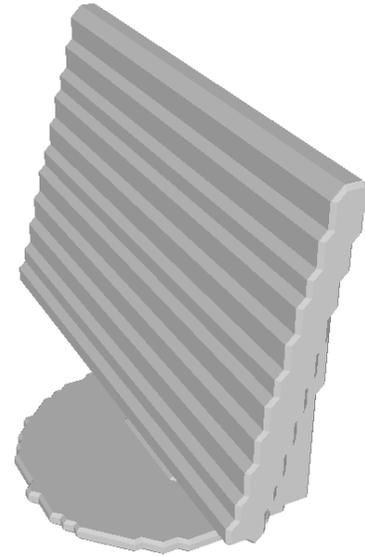
16<sup>3</sup>手法



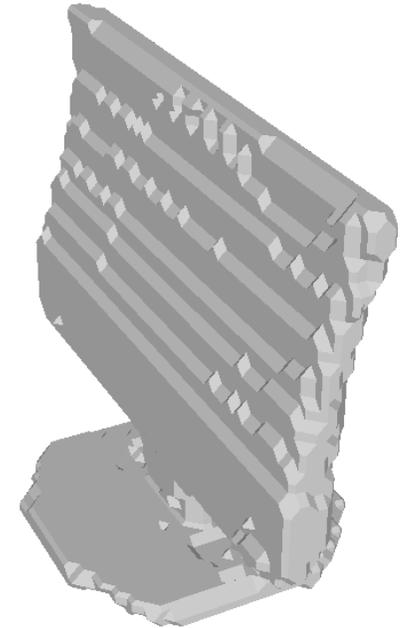
DMTet手法



64<sup>3</sup>手法



提案手法  
(平滑化处理前)



GT



提案手法 (平滑化处理前)、64<sup>3</sup>手法：ボクセルサイズが小さいため凹凸が比較的目立たない

# 実験結果 (定量評価)

	16 <sup>3</sup> 手法	DMTet手法	提案手法	64 <sup>3</sup> 手法
L2 Chamfer Distance ↓	0.035952	0.035134	0.028861	0.027969
L1 Chamfer Distance ↓	0.193098	0.187012	0.158237	0.154968
LFD ↓	3407	4954	3059	1421
Point-to-Surface ↓	0.067025	0.065687	0.051669	0.050333

L1 Chamfer Distance : 生成されたサーフェイスと正解のサーフェイスの間の一致度を評価

L2 Chamfer Distance : Chamfer DistanceをL2距離で評価

LFD (Light Field Distance) : サーフェイスの形状の正確さや滑らかさを評価

Point-to-Surface : 生成されたサーフェイスが正解にどれだけ忠実に再現されているかを評価

# 実験結果（定量評価）

	16 <sup>3</sup> 手法	DMTet手法	提案手法	64 <sup>3</sup> 手法
L2 Chamfer Dist	0.025050	0.025124	0.022061	0.027069
L1 Dist				0.028
LFD				0.021
Poi ↓				0.033

提案手法は64<sup>3</sup>手法の結果には及ばなかったが、低解像度三ボクセルモデルを入力とする手法の中では、16<sup>3</sup>手法とDMTet手法より良い結果を示している

- 64<sup>3</sup>手法：最も良い
- 提案手法：2番目に良い
- DMTet手法：3項目で3番目に良い
- 16<sup>3</sup>手法：3項目で最も悪い

# 考察

- DMTet法によるサーフェイス生成は完全な実装にはなっていない
  - 主観評価、定量評価に影響してあまり良い精度ではなかった
- ボクセルサイズを小さくすることで、サーフェイス生成の精度を向上できることが分かった
- 提案手法でサーフェイス生成した方が $16^3$ 手法より良い結果が得られることが分かった

# まとめ

超解像を用いた低解像度三次元ボクセルモデルからサーフェイスモデルへの変換手法を提案した

1. 3D-SRCGANによる三次元ボクセルモデルの超解像
2. Marching Cubes法によるサーフェイス生成
3. Laplacian Smoothingによるサーフェイスモデルの平滑化  
⇒提案手法により高精細なサーフェイスモデルが生成できた

# 今後の課題

- DMTet法の完全実装
- 3D-SRCGANモデルのさらなる精度向上や、後処理の改良